以下程序有若干语法错误。请找出并改正之。

#include <stdio.h>

**struct** date

{

**int**  y;m;d;

}

**struct** stu

{   **char** n[10];

**struct** date b;

**int** a;

}s={“Wang”,{1980,11,11},30};

main()

{

**printf**(“%c,%d,%d\n”,s.n,s.d,s.a);

}

参考答案

#include <stdio.h>

**struct** date

{

**int**  y,m,d;

}                    ;

**struct** stu

{   **char** n[10];

**struct** date b;

**int** a;

}                    s={"Wang",{1980,11,11},30};

main()

{

**printf**("%s,%d,%d\n",s.n,s.b.d,s.a);

}

将一个字符串插入至另一个源字符串的某个位置：

将一个字符串2插入到源字符串1中 第一次出现某字符的位置，并打印出形成的新串。

如果 字符串1中找不到输入的字符， 则显示“Not found!”并结束程序。

注：源字符串长度及待插入字符串长度不超过50

|  |
| --- |
| #include <stdio.h>  #include <string.h>         //1  #include <stdlib.h>  /\*将一个字符串插入到另一个字符串中首次出现某字符的指定位置\*/  **int** main()  {  **int** i = 0, j = 0, n, len;  **char** str1[200] = {'\0'};  **char** str2[100];  **char** strtem[100];  **char** ch;  **printf**("Input source string 1:\n");  **gets**(str1);     //1  **printf**("Input inserted string 2:\n");  **gets**(str2);         //1  **printf**("Input a letter to locate the index:\n");  **scanf**(" %c", &ch);          //1      len = **strlen**(str1);           //  1  **while** (str1[i] != '\0')         //1      {  **if** (str1[i] == ch)              //1          {              n = i;  **break**;     //1          }          i++;      }  **if** (i != len)               //1      {  **for** (i = n; str1[i] != '\0'; j++, i++)          //1          {              strtem[j] = str1[i];            //1              str1[i] = '\0';             //1          }          strtem[j] = '\0';                   //1  **strcat**(str1, str2);             //1  **strcat**(str1, strtem);               //1  **printf**("The new string is:%s", str1);      }  **else**      {  **printf**("Not found!");      }    **return** 0;  } |

杨辉三角形

编程打印具有如下形式的杨辉三角形（它的特点是左右两边全是1，从第二行起，中间的每一个数是上一行里相邻两个数之和），其中输出数据的行数n从键盘输入，并且n<=10。

程序运行结果示例1：

Input n (n<=10):

5↙

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

程序运行结果示例2：

Input n (n<=10):

7↙

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

输入格式: "%d"

输出格式：

输入提示信息："input n (n<=10):\n"

输出数据格式："%4d"

数据换行： "\n"

#include<stdio.h>

#define ARR\_SIZE 11

void YHTriangle(int a[][ARR\_SIZE],int n);

void PrintYHTriangle(int a[][ARR\_SIZE],int n);

int main(void)

{

int a[ARR\_SIZE][ARR\_SIZE],n;

printf("input n(n<=10):\n");

scanf("%d",&n);//根据要求输入杨辉三角形的行数

YHTriangle(a,n);

PrintYHTriangle(a,n);

return 0;

}

//函数功能：计算n行杨辉三角形中各元素数值

void YHTriangle(int a[][ARR\_SIZE],int n)

{

int i,j;

for(i=1;i<=n;i++)

{

a[i][1]=1;

a[i][i]=1;

}

for(i=3;i<=n;i++)

{

for(j=2;j<=i-1;j++)

{

a[i][j]=a[i-1][j-1]+a[i-1][j];

}

}

}

//函数功能：输出n行杨辉三角形

void PrintYHTriangle(int a[][ARR\_SIZE],int n)

{

int i,j;

for(i=1;i<=n;i++)

{

for(j=1;j<=i;j++)

{

printf("%4d",a[i][j]);

}

printf("\n");

}

}

Q257.(10分)输入一行字符，用函数编程统计其中有多少单词。假设单词之间以空格分开。

\*\*输入格式要求：提示信息："Input a string:"

\*\*输出格式要求："Numbers of words = %d\n"

参考答案

#include <stdio.h>

int CountWords(char str[]);

int main()

{

    char  str[20];

    printf("Input a string:");

    gets(str);

    printf("Numbers of words = %d\n", CountWords(str));

    return 0;

}

int CountWords(char str[])

{

    int    i, num;

    num = (str[0] != ' ') ? 1 : 0;

    for (i = 1; str[i] != '\0'; i++)

    {

        if (str[i] != ' ' && str[i - 1] == ' ')

        {

            num++;

        }

    }

    return num;

}

Q1683.(10分)用二维数组作为函数参数，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int a[][N], int at[][M], int m, int n);

void InputMatrix(int a[][N], int m, int n);

void PrintMatrix(int at[][M], int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输出提示信息和格式："The transposed matrix is:\n

参考答案

#include <stdio.h>

#define M 10

#define N 10

void Transpose(int a[][N], int at[][M], int m, int n);

void InputMatrix(int a[][N], int m, int n);

void PrintMatrix(int at[][M], int n, int m);

int main()

{

int s[M][N], st[N][M], m, n;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(s, m, n);

Transpose(s, st, m, n);

printf("The transposed matrix is:\n");

PrintMatrix(st, n, m);

return 0;

}

/\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/

void Transpose(int a[][N], int at[][M], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

at[j][i] = a[i][j];

}

}

}

/\* 函数功能：输入m\*n矩阵a的值 \*/

void InputMatrix(int a[][N], int m, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i][j]);

}

}

}

/\* 函数功能：输出n\*m矩阵at的值 \*/

void PrintMatrix(int at[][M], int n, int m)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

printf("%d\t", at[i][j]);

}

printf("\n");

}

}

Q552.(10分)如果整数A的全部因子（包括1，不包括A本身）之和等于B；且整数B的全部因子（包括1，不包括B本身）之和等于A，则A和B称为亲密数。求10000以内的亲密数。

\*\*输出格式要求："(%4d,%4d)\n"

程序运行示例如下：

( 220, 284)

(1184,1210)

(2620,2924)

(5020,5564)

(6232,6368)

参考答案

#include <stdio.h>

int main()

{

int a, b, i, n;

for (a = 1; a < 10000; a++)

{

b = 0;

for (i = 1; i <= a / 2; i++)

{

if (a % i == 0)

b += i;

}

n = 0;

for (i = 1; i <= b / 2; i++)

{

if (b % i == 0)

n += i;

}

if (n == a && a < b)

printf("(%4d,%4d)\n", a, b);

}

return 0;

}

Q381.(10分)\*验证卡布列克运算。即：任意一个四位数，只要它们各个位上的数字是不全相同的，就有这样的规律：

（1）将组成该四位数的四个数字由大到小排列，形成由这四个数字构成的最大的四位数；

（2）将组成该四位数的四个数字由小到大排列，形成由这四个数字构成的最小的四位数（如果四个数字中含有0，则得到的数不足四位）；

（3）求两个数的差，得到一个新的四位数（高位零保留）。

重复以上过程，最后得到的结果总是6174。这个数被称为卡布列克常数。请编写一个递归函数，完成以上的卡布列克运算。

\*\*输入格式要求："%d" 提示信息："Enter number:"

\*\*输出格式要求：" [%d]:%d-%d=%d\n"

程序运行示例如下：

Enter number: 1234

[1]:4321-1234=3087

[2]:8730-378=8352

[3]:8532-2358=6174

参考答案

#include <stdio.h>

int count=0;

main()

{

int n;

printf("Enter number:");

scanf("%d",&n);

vr6174(n);

}

vr6174(int num)

{

int each[4],max,min;

if(num!=6174&&num)

{

parse\_sort(num,each);

max\_min(each,&max,&min);

num=max-min;

printf(" [%d]:%d-%d=%d\n",++count,max,min,num);

vr6174(num);

}

}

parse\_sort(int num,int \*each)

{

int i,\*j,\*k,temp;

for(i=0;i<=4;i++)

{

j=each+3-i;

\*j=num%10;

num/=10;

}

for(i=0;i<3;i++)

for(j=each,k=each+1;j<each+3-i;j++,k++)

if(\*j>\*k)

{ temp=\*j; \*j=\*k; \*k=temp;}

return;

}

max\_min(int \*each,int \*max,int \*min)

{

int \*i;

\*min=0;

for(i=each;i<each+4;i++)

\*min=\*min\*10+\*i;

\*max=0;

for(i=each+3;i>=each;i--)

\*max=\*max\*10+\*i;

return;

}

Q457.(10分)从键盘输入一个长度为N（比如10）的整型数组，而后将数组中小于零的元素移动到数组的前端，大于零的元素移动到数组的后端，等于零的元素留在数组中间。比如原始数组为：2 -5 -89 75 0 -89 0 93 48 0，经过处理后的数组为：-5 -89 -89 0 0 0 75 93 48 2。由于不要求数组有序，所以不允许用排序方法。

\*\*输入格式要求："%d" 提示信息："Please input %d interger number\n"

\*\*输出格式要求："%5d"

程序运行示例如下：

Please input 10 interger number

2 -5 -89 75 0 -89 0 93 48 0

-5 -89 -89 0 0 0 75 93 48 2

参考答案

#include <stdio.h>

#define N 10

int main()

{

int a[N];

int i, p1 = 0, p2 = 9, p, temp;

printf("Please input %d interger number\n", N);

for (i = 0; i <= N - 1; i++)

{

scanf("%d", &a[i]);

}

for (; p1 < p2 ; )

{

if (a[p1] == 0)

{

p = p1;

while (a[p1 + 1] == 0)

{

p1++;

}

if (p1 + 1 < p2)

{

temp = a[p1 + 1];

a[p1 + 1] = a[p];

a[p] = temp;

p1 = p;

continue;

}

else

{

break;

}

}

if (a[p1] > 0)

{

temp = a[p1];

a[p1] = a[p2];

a[p2] = temp;

p2--;

}

else

{

p1++;

}

if (a[p2] == 0)

{

p = p2;

while (a[p2 - 1] == 0)

{

p2--;

}

if (p1 < p2 - 1)

{

temp = a[p2 - 1];

a[p2 - 1] = a[p];

a[p] = temp;

p2 = p;

continue;

}

else

{

break;

}

}

if (a[p2] < 0)

{

temp = a[p2];

a[p2] = a[p1];

a[p1] = temp;

p1++;

}

else

{

p2--;

}

}

for (i = 0; i <= N - 1; i++)

{

printf("%5d", a[i]);

}

printf("\n");

}

Q305.(10分)用100元人民币兑换10元,5元和1元的纸币(每一种都要有)共50张,请用穷举法编程计算共有几种兑换方案,每种方案各兑换多少张纸币.

\*\*\*\*输入提示信息要求为：无

\*\*\*\*输出格式要求为：

"x = %d, y = %d, z = %d\n"

"count = %d\n"

参考答案

#include <stdio.h>

int main()

{

int x, y, z, count = 0;

for (x = 0; x <= 5; x++) //1

{

for (y = 0; y <= 10; y++) //1

{

z = 50 - x - y;//1

if (10 \* x + 5 \* y + z == 100) //2

{

count++; //1

printf("x = %d, y = %d, z = %d\n", x, y, z);

}

}

}

printf("count = %d\n", count);//1

return 0;

}

Q423.(10分)分数比较。比较两个分数的大小。

\*\*输入格式要求："%d%d,%d%d" 提示信息："Input two FENSHU :\n"

\*\*输出格式要求："%d/%d>%d/%d\n" "%d/%d<%d/%d\n" "%d/%d=%d/%d\n"

程序运行示例如下：

Input two FENSHU :

1 3,1 4

1/3>1/4

参考答案

#include <stdio.h>

main()

{int i,j,k,l,m,n;

printf("Input two FENSHU :\n");

scanf("%d%d,%d%d",&i,&j,&k,&l);

m=zxgb(j,l)/j\*i;

n=zxgb(j,l)/l\*k;

if(m>n)

printf("%d/%d>%d/%d\n",i,j,k,l);

else if(m==n)

printf("%d/%d=%d/%d\n",i,j,k,l);

else printf("%d/%d<%d/%d\n",i,j,k,l);

}

zxgb(a,b)

int a,b;

{ long int c;

int d;

if(a<b) c=a,a=b,b=c;

for(c=a\*b;b!=0;)

{d=b;b=a%b;a=d;}

return((int)c/a);

}

Q461.(10分)（附加题）编程将字符串s倒序输出，要求利用函数递归实现。

\*\*输入格式要求："%s" 提示信息："input your string:\n"

\*\*输出格式要求："%c"

程序运行的输入输出样例：

屏幕先输出提示信息：

input your string:

然后用户键盘输入：

abcdefg

最后屏幕输出：

Gfedcba

#include <stdio.h>

void reverse(char s[])

{

int len;

len = strlen(s);

if (len == 1)

{

printf("%c", s[0]);

}

else

{

reverse(s + 1);

printf("%c", s[0]);

}

}

int main()

{

char s[100];

printf("input your string:\n");

scanf("%s", s);

reverse(s);

}

Q780.(5分)改错题：用户从键盘任意输入一个数字表示月份值n，程序显示该月份对应的英文表示，若n不在1～12之间，则输出“Illegal month”。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include <stdio.h>

main()

{

int n;

char monthName[][20] = {"Illegal month", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"};

printf("Input month number:");

scanf("%d", n);

if ((n < 12) || (n > 1))

{

printf("month %d is %c\n", n, monthName[n][n]);

}

else

{

printf("%s\n", monthName[n]);

}

}

参考答案

#include <stdio.h>

main()

{

int n;

char monthName[][20] = {"Illegal month", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"};

printf("Input month number:");

scanf("%d", &n);

if ((n <= 12) && (n >= 1))

{

printf("month %d is %s\n", n, monthName[n]);

}

else

{

printf("%s\n", monthName[0]);

}

}

Q263.(10分)从键盘任意输入一个整型表示的月份值，用指针数组编程输出该月份的英文表示，若输入的月份值不在1～12之间，则输出“Illegal month”。

\*\*输入格式要求："%d" 提示信息："Input month number:"

\*\*输出格式要求："month %d is %s\n"

"Illegal month", "January", "February", "March", "April", "May", "June", "July", August", "September", "October", "November", "December"

程序运行示例1如下：

Input month number:5

month 5 is May

程序运行示例2如下：

Input month number:13

Illegal month

参考答案

#include <stdio.h>

int main()

{

int n;

static char \*monthName[] = {"Illegal month", "January", "February",

"March", "April", "May", "June", "July",

"August", "September", "October",

"November", "December"

};

printf("Input month number:");

scanf("%d", &n);

if ((n <= 12) && (n >= 1))

{

printf("month %d is %s\n", n, monthName[n]);/\* 输出相应月份 \*/

}

else

{

printf("%s\n", monthName[0]); /\* 输出错误提示信息 \*/

}

return 0;

}

Q438.(10分)哪个大夫哪天值班

医院有A、B、C、D、E、F、G七位大夫，在一个星期内（星期一至星期天）

每天要轮流值班一天。现在已知：

A大夫比C大夫晚一天值班；

D大夫比E大夫晚两天值班；

B大夫比G大夫早三天值班；

F大夫的值班日在B和C大夫的中间，且是星期四；

请确定每天究竟哪位大夫值班？

\*\*输出格式要求："Doctor %c is on duty %s.\n"

程序运行示例如下：

Doctor E is on duty MONDAY.

Doctor B is on duty TUESDAY.

Doctor D is on duty WEDNESDAY.

Doctor F is on duty THURDAY.

Doctor G is on duty FRIDAY.

Doctor C is on duty STAURDAY.

Doctor A is on duty SUNDAY.

参考答案

#include<stdio.h>

int a[8];

char \*day[]={

"","MONDAY","TUESDAY","WEDNESDAY","THURDAY",

"FRIDAY","STAURDAY","SUNDAY"};

main()

{

int i,t,j;

a[4]=6;

for(i=1;i<=3;i++)

{

a[i]=2;

if(!a[i+3])

a[i+3]=7;

else

{

a[i]=0;

continue;

}

for(t=1;t<=3;t++)

{

if(!a[t])

a[t]=5;

else

continue;

if(!a[t+2])

a[t+2]=4;

else

{

a[t]=0;

continue;

}

for(j=5;j<7;j++)

{

if(!a[j])

a[j]=3;

else

continue;

if(!a[j+1])

a[j+1]=1;

else

{

a[j]=0;

continue;

}

for(i=1;i<=7;i++)

printf("Doctor %c is on duty %s.\n",'A'+a[i]-1,day[i]);

exit(0);

}

}

}

}

Q1689.(10分)按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

输入某一年的第几天，计算并输出它是这一年的第几月第几日。

/\* 函数功能： 对给定的某一年的第几天，计算它是这一年的第几月第几日

函数入口参数：整型变量year，存储年

整型变量yearDay，存储这一年的第几天

函数出口参数：整型指针pMonth，指向存储这一年第几月的整型变量

整型指针pDay，指向存储第几日的整型变量

函数返回值： 无 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

输入提示信息："Please enter year, yearDay:"

输入格式："%d,%d"

输出提示信息和格式："month = %d, day = %d\n"

参考答案

#include <stdio.h>

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

int dayTab[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}, {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

int main()

{

int year, month, day, yearDay;

printf("Please enter year, yearDay:");

scanf("%d,%d", &year, &yearDay);

MonthDay(year, yearDay, &month, &day);

printf("month = %d, day = %d\n", month, day);

return 0;

}

/\* 函数功能：对给定的某一年的第几天，计算它是这一年的第几月第几日 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay)

{

int i, leap;

leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

for (i = 1; yearDay > dayTab[leap][i]; i++)

{

yearDay = yearDay - dayTab[leap][i];

}

\*pMonth = i; /\* 将计算出的月份值赋值给pMonth所指向的变量 \*/

\*pDay = yearDay; /\* 将计算出的日号赋值给pDay所指向的变量 \*/

}

Q1735.(10分)

编程输出如下上三角形式的九九乘法表。

\*\*输出格式要求："%4d"

参考答案

#include <stdio.h>

main()

{

int m, n, i;

for (n = 1; n < 10; n++) //被乘数n从1变化到9

{

for (i = 1; i <= 4 \* n - 4; i++)

{

printf(" "); //输出相应空格使数字右对齐

}

for (m = n; m < 10; m++) //乘数m从n变化到9

{

printf("%4d", m \* n); //输出第n行m列中的m\*n的值

}

printf("\n"); //输出换行符,准备打印下一行

}

}

Q588.(10分)编写一个程序，求满足以下条件的最大的n：1^2 + 2^2 + 3^2 + ... + n^2 <= 1000。

x^2表示x的平方。

\*\*输出格式要求："n=%d\n"

参考答案

#include <stdio.h>

int main()

{

int n, s;

s = n = 0;

while (s <= 1000)

{

++n;

s += n \* n;

}

printf("n=%d\n", n - 1);

return 0;

}

Q1595.(10分)马克思手稿中有一道趣味数学题：有30个人，其中有男人、女人和小孩，在一家饭馆里吃饭共花了50先令，每个男人各花3先令，每个女人各花2先令，每个小孩各花1先令，问男人、女人和小孩各有几人？

假设男人为x人，女人为y人，小孩为z人，则有以下方程组：

{x+y+z=303x+2y+z=50

main()

{

int x,y,z;

printf("Man \t Women \t Childern\n");

for (x=1; x<16; x++)

for (y=1; y<25; y++)

{

z = 30 – x - y;

if (3 \* x + 2 \* y + z = 50)

printf("%3f\t %5f\t %8f\n",x,y,z);

}

}

参考答案

#include <stdio.h>

main()

{

int x,y,z;

printf("Man \t Women \t Childern\n");

for (x=0; x<=16; x++)

for (y=0; y<=25; y++)

{

z = 30 - x - y;

if (3 \* x + 2 \* y + z == 50)

printf("%3d \t %5d \t %8d\n",x,y,z);

}

}

Q1314.(10分)有一篇文章，共有三行文字，每行有80个字符。要求分别统计出其中英文大写字母、小写字母、数字、空格、以及其它字符的个数。

程序的运行示例如下：

请输入第0行：

helloworld

请输入第1行：

12345gogogo

请输入第2行：

end bye bye.

helloworld

12345gogogo

end bye bye.

大写字母数：0

小写字母数：25

数字个数 ：5

空格个数 ：2

其它字符 ：1

参考答案

#include <stdio.h>

int main()

{

int i, j, uppn, lown, dign, span, othn;

char text[3][80];

uppn = lown = dign = span = othn = 0;

for (i = 0; i < 3; i++)

{

printf("\n请输入第%d行：\n", i);

gets(text[i]);

for (j = 0; j < 80 && text[i][j] != '\0'; j++)

{

if (text[i][j] >= 'A' && text[i][j] <= 'Z')

{

uppn += 1;

}

else if (text[i][j] >= 'a' && text[i][j] <= 'z')

{

lown += 1;

}

else if (text[i][j] >= '0' && text[i][j] <= '9')

{

dign += 1;

}

else if (text[i][j] == ' ')

{

span += 1;

}

else

{

othn += 1;

}

}

}

for (i = 0; i < 3; i++)

{

printf("%s\n", text[i]);

}

printf("大写字母数：%d\n", uppn);

printf("小写字母数：%d\n", lown);

printf("数字个数 ：%d\n", dign);

printf("空格个数 ：%d\n", span);

printf("其它字符 ：%d\n", othn);

return 0;

}

Q559.(10分)一球从100米高度自由落下，每次落地后反跳回原高度的一半；再落下，求它在第10次落地时，共经过多少米？第10次反弹多高？

\*\*输出格式要求："the total of road is %f\n" "the tenth is %f meter\n"

程序运行示例如下：

the total of road is 299.609375

the tenth is 0.097656 meter

参考答案

#include <stdio.h>

int main()

{

float sn = 100.0, hn = sn / 2;

int n;

for (n = 2; n <= 10; n++)

{

sn = sn + 2 \* hn;

hn = hn / 2;

}

printf("the total of road is %f\n", sn);

printf("the tenth is %f meter\n", hn);

return 0;

}

Q498.(20分)对输入的字符串进行长度验证，保证输入的字符串的长度在指定的范围内，如果不在指定的范围内，则一直提示用户输入，直到输入合法长度的字符串为止。

程序的示例运行如下：

请输入一个字符串(长度为[3..5]个字符)：a

请输入一个字符串(长度为[3..5]个字符)：ab

请输入一个字符串(长度为[3..5]个字符)：abcdef

请输入一个字符串(长度为[3..5]个字符)：abc

你输入的字符串为：abc

参考答案

#include <stdio.h>

void getString(char \*str, int min, int max)

{

int len;

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

while (len < min || len > max)

{

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

}

}

int main(int argc, char \*argv[])

{

char string[100];

getString(string, 3, 5);

printf("你输入的字符串为：%s\n", string);

return 0;

}

Q376.(10分)采用递归方法计算x的n次方。

\*\*输入格式要求："%d%d" 提示信息："x=? n=?"

\*\*输出格式要求："%d\*\*%d=%d\n"

程序运行示例如下：

x=? n=?2 8

2\*\*8=256

参考答案

#include <stdio.h>

main()

{

int x,n;

printf("x=? n=?");

scanf("%d%d",&x,&n);

printf("%d\*\*%d=%d\n",x,n,power(x,n));

}

power(int x,int n)

{

if(n==0) return 1;

else return x\*power(x,n-1);

}

Q567.(10分)请编写函数fuc（）,函数的功能是： 根据以下公式求出p的值，结果由函数值返回。m , n 是两个正整数，且 m>n. P= m!/(n!\*(n-m))!

\*\*输入格式要求："%d %d" 提示信息："请输入m,n的值( m>n )：\n"

\*\*输出格式要求："n项之和为：%lf\n"

程序运行示例如下：

请输入m,n的值( m>n )：

5 3

n项之和为：10.000000

参考答案

#include <stdio.h>

double fac(int m, int n)

{

int i;

long p = 1, q = 1, r = 1;

double d = 0.0;

for (i = 1; i <= m; i++)

{

p = p \* i;

}

for (i = 1; i <= n; i++)

{

q = q \* i;

}

for (i = 1; i <= m - n; i++)

{

r = r \* i;

}

d = (double)p / (r \* q);

return d;

}

main()

{

int n, m;

double d;

do

{

printf("请输入m,n的值( m>n )：\n");

scanf("%d %d", &m, &n);

}

while (m < n);

d = fac(m, n);

printf("n项之和为：%lf\n", d);

}

Q1159.(10分)

下面程序的功能是将数组元素倒置 ，找出其中的4处错误并改正之。

#include "stdio.h"

#define M 5;

main()

{ int a[M]={1,2,3,4,5};

int i,j,\*t;

i=0;j=M-1;

while(i)

{

t=\*(a+i);

\*(a+i)=\*(a+j);

\*(a+j)=t;

i++;j++

}

for(i=0;i<M;i++) printf("%d",(a+i));

}

参考答案

#include "stdio.h"

#define M 5

main()

{ int a[M]={1,2,3,4,5};

int i,j,t;

i=0;j=M-1;

while(i<=j)

{

t=\*(a+i);

\*(a+i)=\*(a+j);

\*(a+j)=t;

i++;j--;

}

for(i=0;i<M;i++) printf("%d",\*(a+i));

}

Q1601.(10分)按如下函数原型编写程序，输入10个整数，用函数编程将其中最大数与最小数位置互换，然后输出互换后的数组。

void MaxMinExchang(int a[], int n);

int FindMaxPos(int s[], int n);

int FindMinPos(int s[], int n);

输入提示信息："Input 10 numbers:"

输入格式："%d"

输出提示信息："Exchang results:"

输出格式："%4d"

参考答案

#include <stdio.h>

void MaxMinExchang(int a[], int n);

int FindMaxPos(int s[], int n);

int FindMinPos(int s[], int n);

int main()

{

int i, a[10];

printf("Input 10 numbers:");

for (i = 0; i < 10; i++)

{

scanf("%d", &a[i]);

}

MaxMinExchang(a, 10);

printf("Exchang results:");

for (i = 0; i < 10; i++)

{

printf("%4d", a[i]);

}

printf("\n");

return 0;

}

/\* 函数功能：将数组中的最大数与最小数位置互换 \*/

void MaxMinExchang(int a[], int n)

{

int maxPos, minPos, temp;

maxPos = FindMaxPos(a, n);

minPos = FindMinPos(a, n);

temp = a[maxPos];

a[maxPos] = a[minPos];

a[minPos] = temp;

}

/\* 函数功能：计算数组中的最大值在数组中的下标位置 \*/

int FindMaxPos(int s[], int n)

{

int maxPos = 0, max = s[0], i;

for (i = 1; i < n; i++)

{

if (s[i] > max)

{

max = s[i];

maxPos = i;

}

}

return maxPos;

}

/\* 函数功能：计算数组中的最小值在数组中的下标位置 \*/

int FindMinPos(int s[], int n)

{

int minPos = 0, min = s[0], i;

for (i = 1; i < n; i++)

{

if (s[i] < min)

{

min = s[i];

minPos = i;

}

}

return minPos;

}

Q1687.(10分)用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

参考答案

#include <stdio.h>

#include <string.h>

#define N 100

void Squeeze(char \*s, char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char \*s, char c)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; t++)

{

if (\*t != c)

{

\*s = \*t;

s++;

}

}

\*s = '\0'; /\* 在字符串t2的末尾添加字符串结束标志 \*/

}

Q1715.(10分)假设今年的工业产值为100万元，产值增长率为每年c%，请编程计算当c分别为6、8、10、12时工业产值分别过多少年可实现翻一番（即增加一倍）。

输入格式：无

输出格式："When grow rate is %d%%, the output can be doubled after %d years.\n"

参考答案

#include <stdio.h>

#define CURRENT 100

#define N 4

main()

{

int i;

int year; /\* 产值翻番所需年数 \*/

int growRate[N] = {6, 8, 10, 12}; /\* 工业产值的增长率 \*/

double output; /\* 工业总产值 \*/

for (i = 0; i < N; i++)

{

output = CURRENT; /\* 当年产值为100万元 \*/

for (year = 0; output < 2 \* CURRENT; year++)

{

output = output \* (1 + growRate[i] / 100.0);

}

printf("When grow rate is %d%%, the output can be doubled after

%d years.\n", growRate[i], year);

}

}

Q297.(15分)两个正整数的最大公约数是能够整除这两个整数的最大整数。

请按如下函数原型编写计算最大公约数的函数Gcd()。

int Gcd(int a, int b);

在主函数中调用该函数计算，并输出从键盘任意输入的两整数的最大公约数。

\*\*要求输入提示信息为: "Input a,b:\n"

\*\*要求输入格式为: "%d,%d"

\*\*要求输出格式为

如果a或b不是正整数，输出: "Input number should be positive!\n"

否则输出: "Greatest Common Divisor of %d and %d is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

int Gcd(int a, int b);

main()

{

int a, b, c;

printf("Input a,b:\n");

scanf("%d,%d", &a, &b); //1

c = Gcd(a, b); //1

if (c != -1) //1

{

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c); //1

}

else

{

printf("Input number should be positive!\n"); //1

}

}

int Gcd(int a, int b) //1

{

int i, t;

if (a <= 0 || b <= 0) //1

{

return -1; //1

}

t = a < b ? a : b; //2

for (i = t; i > 0; i--)//2

{

if (a % i == 0 && b % i == 0)//1

return i; //1

}

return 1; //1

}

Q320.(10分)请用循环语句，打印输出如下图所示的下三角形乘法九九表。

1

2 4

3 6 9

4 8 12 16

5 10 15 20 25

6 12 18 24 30 36

7 14 21 28 35 42 49

8 16 24 32 40 48 56 64

9 18 27 36 45 54 63 72 81

\*\*输入提示信息格式: 无

\*\*输入数据格式要求: 无

\*\*输出数据格式要求: "%4d"

参考答案

#include <stdio.h>

int main()

{

int i, j;

for (i = 1; i < 10; i++) //2

{

for (j = 1; j <= i; j++) //2

{

printf("%4d", i \* j); //1

}

printf("\n"); //1

}

return 0;

}

Q1281.(10分)

程序改错。下面程序的功能是从键盘输入一行字符，统计其中有多少单词。假设单词之间以空格分开。[提示：判断是否有新单词出现的方法是——当前被检验的字符不是空格，而前一被检验字符是空格，则表示有新单词出现。]

#include<stdio.h>

#define ARR\_SIZE 80;

main()

{

char str[ARR\_SIZE];

int i, num;

gets(str);

if(str[0] != ' ' && str[0] != '\0')

{

num = 1;

}

else

{

num = 0;

}

for(i = 1 ; str[i] != '\0'; i++)

{

if(str[i] != ' ' && str[i-1] = ' ')

{

num++;

}

}

printf("num=%d\n", num);

}

参考答案

#include<stdio.h>

#define ARR\_SIZE 80

main()

{

char str[ARR\_SIZE];

int i, num;

gets(str);

if(str[0] != ' ' && str[0] != '\0')

{

num = 1;

}

else

{

num = 0;

}

for(i = 1 ; str[i] != '\0'; i++)

{

if(str[i] != ' ' && str[i-1] == ' ')

{

num++;

}

}

printf("num=%d\n", num);

}

Q595.(10分)编程输入一个整数m（int型），输出该整数的位数n。

\*\*输入格式要求："%d"

\*\*输出格式要求："n=%d"

程序运行示例1如下：

345

n=3

程序运行示例2如下：

0

n=1

参考答案

#include <stdio.h>

main()

{

int m, n;

scanf("%d", &m);

n = 0;

do

{

m = m / 10;

n++;

}

while (m != 0);

printf("n=%d", n);

}

Q544.(10分)某班期末考试科目为数学（MT）、英语（EN）和物理（PH），有最多不超过40人参加考试。请编程计算：（1）每个学生的总分和平均分；（2）每门课程的总分和平均分。

\*\*输入格式要求："%d" "%ld"(学号) 提示信息："Input the total number of the students(n<40):" "Input student’s ID and score as: MT EN PH:\n"

\*\*输出格式要求："Counting Result:\n" "Student’s ID\t MT \t EN \t PH \t SUM \t AVER\n" "%12ld\t"(打印学号) "%4d\t"（打印每门课成绩）"%4d\t%5.1f\n"（打印总分和平均分） "SumofCourse \t" "%4d\t"(打印每门课的总分) "\nAverofCourse\t" "%4.1f\t"(每门课的平均分)

程序的运行示例如下：

Input the total number of the students(n<=40):4

Input student’s ID and score as: MT EN PH:

070310122 97 87 92

070310123 92 91 90

070310124 90 81 82

070310125 73 65 80

Counting Result:

Student’s ID MT EN PH SUM AVER

070310122 97 87 92 276 92.0

070310123 92 91 90 273 91.0

070310124 90 81 82 253 84.3

070310125 73 65 80 218 72.7

SumofCourse 352 324 344

AverofCourse 88.0 81.0 86.0

参考答案

#include <stdio.h>

#define STUD\_N 40 /\* 最多学生人数 \*/

#define COURSE\_N 3 /\* 考试科目数 \*/

int ReadScore(int score[][COURSE\_N], long num[]);

void AverforStud(int score[][COURSE\_N], int sum[], float aver[], int n);

void AverforCourse(int score[][COURSE\_N], int sum[], float aver[], int n);

void Print(int score[][COURSE\_N], long num[], int sumS[], float averS[],

int sumC[], float averC[], int n);

int main()

{

int score[STUD\_N][COURSE\_N], sumS[STUD\_N], sumC[STUD\_N], n;

long num[STUD\_N];

float averS[STUD\_N], averC[STUD\_N];

n = ReadScore(score, num); /\* 读入学生成绩 \*/

AverforStud(score, sumS, averS, n); /\* 计算每个学生的总分平均分 \*/

AverforCourse(score, sumC, averC, n); /\* 计算每门课程的总分平均分 \*/

Print(score, num, sumS, averS, sumC, averC, n); /\* 输出学生成绩 \*/

return 0;

}

/\* 函数功能：输入学生的学号及其三门课的成绩，当输入负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[][COURSE\_N], long num[])

{

int i, j, n;

printf("Input the total number of the students(n<40):");

scanf("%d", &n); /\* 输入参加考试的学生人数 \*/

printf("Input student’s ID and score as: MT EN PH:\n");

for (i = 0; i < n; i++) /\* 对所有学生进行循环 \*/

{

scanf("%ld", &num[i]); /\* 以长整型格式输入每个学生的学号 \*/

for (j = 0; j < COURSE\_N; j++) /\* 对所有课程进行循环 \*/

{

scanf("%d", &score[i][j]); /\* 输入每个学生的各门课成绩 \*/

}

}

return i; /\* 返回学生人数 \*/

}

/\* 函数功能： 计算每个学生的总分和平均分 \*/

void AverforStud(int score[][COURSE\_N], int sum[], float aver[], int n)

{

int i, j;

for (i = 0; i < n; i++)

{

sum[i] = 0;

for (j = 0; j < COURSE\_N; j++) /\* 对所有课程进行循环 \*/

{

sum[i] = sum[i] + score[i][j]; /\* 计算第i个学生的总分 \*/

}

aver[i] = (float)sum[i] / COURSE\_N;/\* 计算第i个学生的平均分 \*/

}

}

/\* 函数功能：计算每门课程的总分和平均分 \*/

void AverforCourse(int score[][COURSE\_N], int sum[], float aver[], int n)

{

int i, j;

for (j = 0; j < COURSE\_N; j++)

{

sum[j] = 0;

for (i = 0; i < n; i++) /\* 对所有学生进行循环 \*/

{

sum[j] = sum[j] + score[i][j]; /\* 计算第j门课程的总分 \*/

}

aver[j] = (float)sum[j] / n; /\* 计算第j门课程的平均分 \*/

}

}

/\*函数功能：打印每个学生的学号、各门课成绩、总分和平均分，以及每门课的总分和平均分\*/

void Print(int score[][COURSE\_N], long num[], int sumS[], float averS[],

int sumC[], float averC[], int n)

{

int i, j;

printf("Counting Result:\n");

printf("Student’s ID\t MT \t EN \t PH \t SUM \t AVER\n");

for (i = 0; i < n; i++)

{

printf("%12ld\t", num[i]); /\* 以长整型格式打印学生的学号 \*/

for (j = 0; j < COURSE\_N; j++)

{

printf("%4d\t", score[i][j]); /\* 打印学生的每门课成绩 \*/

}

printf("%4d\t%5.1f\n", sumS[i], averS[i]);/\*打印学生的总分平均分\*/

}

printf("SumofCourse \t");

for (j = 0; j < COURSE\_N; j++) /\* 打印每门课的总分 \*/

{

printf("%4d\t", sumC[j]);

}

printf("\nAverofCourse\t");

for (j = 0; j < COURSE\_N; j++) /\* 打印每门课的平均分 \*/

{

printf("%4.1f\t", averC[j]);

}

printf("\n");

}

Q270.(10分)编程用链表方法实现堆栈数据结构。堆栈（Strack）是指这样一段内存，它可以理解为一个筒结构，先放进筒中的数据被后放进筒中的数据“压住”，只有后放进筒中的数据都取出后，先放进去的数据才能被取出，称为“后进先出”。堆栈的长度可随意增加。堆栈结构可用链表实现。设计一个链表结构需包含两个成员：一个存放数据，一个为指向下一个节点的指针。当每次有一个新数据要放入堆栈时，称为“压入堆栈”，这时动态建立一个链表的节点，并连接到链表的结尾；当每次从堆栈中取出一个数据时，称为“弹出堆栈”，这意味着从链表的最后一个节点中取出该节点的数据成员，同时删除该节点，释放该节点所占的内存。

参考答案

#include <stdio.h>

#include <stdlib.h>

**typedef** **struct** stack

{

**int** data;

**struct** stack \*next;

}                     STACK;

STACK \*head, \*pr;

**int** nodeNum = 0;                            /\* 堆栈节点数寄存器 \*/

STACK \*CreateNode(**int** num);

STACK \*PushStack(**int** num);

**int** PopStack(**void**);

**int** main()

{

**int** pushNum[5] , popNum[5], i;

**for** (i = 0; i < 5; i++)

    {

**printf**("请输入要压栈的整数：");

**scanf**("%d",&pushNum[i]);

        PushStack(pushNum[i]);

**printf**("Push %dth Data:%d\n", i + 1, pushNum[i]);

    }

**for** (i = 0; i < 5; i++)

    {

        popNum[i] = PopStack();

**printf**("Pop %dth Data:%d\n", 5 - i, popNum[i]);

    }

**return** 0;

}

/\* 函数功能：生成一个新的节点，并为该节点赋初值，返回指向新的节点的指针 \*/

STACK \*CreateNode(**int** num)

{

    STACK \*p;

    p = (STACK \*)**malloc**(**sizeof**(STACK));

**if** (p == NULL)

    {

**printf**("No enough memory!\n");

**exit**(0);

    }

    p->next = NULL;              /\* 为新建的节点指针域赋空指针 \*/

    p->data = num;                   /\* 为新建的节点数据区赋值 \*/

**return** p;

}

/\*  函数功能：将整型变量num的值压入堆栈，返回指向链表新节点的指针 \*/

STACK \*PushStack(**int** num)

{

**if** (nodeNum == 0) /\* 若为首节点，则保留该节点地址在head中\*/

    {

        head = CreateNode(num);

        pr = head;

        nodeNum++;  /\* 堆栈节点数寄存器+1 \*/

    }

**else**             /\* 若不是首节点，则将新建节点连到链表的结尾处 \*/

    {

        pr->next = CreateNode(num);

        pr = pr->next;

        nodeNum++;  /\* 堆栈节点数寄存器+1 \*/

    }

**return** pr;

}

/\*  函数功能：将当前栈顶的数据弹出堆栈，返回从堆栈中弹出的数据 \*/

**int** PopStack(**void**)

{

    STACK \*p = head;

**int** result;

**for** (;;)

    {

**if** (p->next == NULL)     /\* 查找最后一个节点 \*/

        {

**break**;

        }

**else**

        {

            pr = p;             /\* 记录最后一个节点的前一个节点的地址 \*/

            p = p->next;

            nodeNum--;          /\* 堆栈节点数寄存器-1 \*/

        }

    }

    pr->next = NULL;          /\* 将末节点的前一个节点置成末节点 \*/

    result = p->data;

**free**(p);

**return** result;

}

Q1330.(50分)13个人围成一圈，从第1个人开始顺序报号1、2、3，凡报到3的人退出圈子。找出最后留在圈子里的人原来的序号。要求用结构体编程实现。

程序的运行示例如下：

出圈成员及顺序： 3 6 9 12 2 7 11 4 10 5 1 8

最后的成员是： 13

参考答案

#include <stdio.h>

#define N 13

struct person

{

int number;

int nextp;

} link[N + 1];

int main(void)

{

int i, count, h;

/\* 建立队列 \*/

for (i = 1; i <= N; i++)

{

if (i == N)

{

link[i].nextp = 1;

}

else

{

link[i].nextp = i + 1;

}

link[i].number = i;

}

printf("\n");

count = 0;

h = N;

printf("出圈成员及顺序：");

while (count < N - 1)

{

i = 0;

while (i != 3)

{

h = link[h].nextp;

if (link[h].number)

{

i++;

}

}

printf("%3d", link[h].number);

link[h].number = 0;

count++;

}

printf("\n最后的成员是：");

for (i = 1; i <= N; i++)

{

if (link[i].number)

{

printf("%3d", link[i].number);

}

}

return 0;

}

Q1332.(10分)创建并输出一个一维数组（含20个元素），数组元素的值分别是下标的3倍多2。

\*\*输出格式要求："%3d"

程序的运行示例如下：

2 5 8 11 14 17 20 23 26 29 32 35 38 41 44 47 50 53 56 59

参考答案

#include <stdio.h>

main()

{

int a[20], k;

for (k = 0; k < 20; k++)

{

a[k] = 3 \* k + 2;

printf("%3d", a[k]);

}

}

Q539.(10分)从键盘输入某班学生某门课的成绩（已知每班人数最多不超过40人，具体人数由键盘输入，成绩为整数），试编程计算其平均分。

\*\*输入格式要求："%d" 提示信息："Input n:" "Average score is %d\n" "Input score:"

\*\*输出格式要求："%d"

程序的运行示例如下：

Input n:3

Input score:80 100 60

Average score is 80

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, result;      for (i = 1; i <= 20; i++)      {          result = i \* i;          if (result > 100)              break;          else              printf("%d\*%d=%d\n", i, i, i \* i);      }  } |

Q1158.(10分)

|  |
| --- |
| 下面程序的功能是读取7个数（1—50）的整数值，每读取一个值，程序打印出该值个数的＊。找出其中的错误，并改正之。  #include "stdio.h"  main()  {   int i,a,n=1;      while(n<=7)      {          do          {              scanf("%d",a);          }while(a<1 && a>50);          for(i=1;i<=a;i++)              printf("\*");          printf("\n");          n++;      }  } |

参考答案

|  |
| --- |
| #include <stdio.h>  #define N 40  int Average(int score[], int n);     /\* Average()函数原型 \*/  void ReadScore(int score[], int n); /\* ReadScore()函数原型 \*/  int main()  {      int score[N], aver, n;      printf("Input n:");      scanf("%d", &n);      ReadScore(score, n);          /\* 数组名作为函数实参调用函数ReadScore() \*/      aver = Average(score, n);    /\* 数组名作为函数实参调用函数Average() \*/      printf("Average score is %d\n", aver);      return 0;  }  /\* 函数功能： 计算n个学生成绩的平均分 \*/  int Average(int score[], int n) /\* Average()函数定义 \*/  {      int i, sum = 0;      for (i = 0; i < n; i++)      {          sum += score[i];      }      return  sum / n;  }  /\* 函数功能：输入n个学生的某门课成绩 \*/  void ReadScore(int score[], int n)  /\* ReadScore()函数定义 \*/  {      int i;      printf("Input score:");      for (i = 0; i < n; i++)      {          scanf("%d", &score[i]);      }  } |

Q1683.(10分)用二维数组作为函数参数，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int a[][N], int at[][M], int m, int n);

void InputMatrix(int a[][N], int m, int n);

void PrintMatrix(int at[][M], int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输出提示信息和格式："The transposed matrix is:\n"

参考答案

#include <stdio.h>

#define M 10

#define N 10

void Transpose(int a[][N], int at[][M], int m, int n);

void InputMatrix(int a[][N], int m, int n);

void PrintMatrix(int at[][M], int n, int m);

int main()

{

int s[M][N], st[N][M], m, n;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(s, m, n);

Transpose(s, st, m, n);

printf("The transposed matrix is:\n");

PrintMatrix(st, n, m);

return 0;

}

/\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/

void Transpose(int a[][N], int at[][M], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

at[j][i] = a[i][j];

}

}

}

/\* 函数功能：输入m\*n矩阵a的值 \*/

void InputMatrix(int a[][N], int m, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i][j]);

}

}

}

/\* 函数功能：输出n\*m矩阵at的值 \*/

void PrintMatrix(int at[][M], int n, int m)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

printf("%d\t", at[i][j]);

}

printf("\n");

}

}

Q358.(10分)有一个3\*4的矩阵，求其中的最大元素的值。矩阵为：

{{1,3,5,7},{2,4,6,8},{15,17,34,12}};

\*\*输出格式要求："max value is %d\n"

程序运行示例如下：

max value is 34

参考答案

#include <stdio.h>

maxvalue(array)

int array[][4];

{

int i,j,k,max;

max=array[0][0];

for(i=0;i<3;i++)

for(j=0;j<4;j++)

if(array[i][j]>max)

max=array[i][j];

return max;

}

main()

{

static int a[3][4]={{1,3,5,7},{2,4,6,8},{15,17,34,12}};

printf("max value is %d\n",maxvalue(a));

}

Q434.(10分)除式还原（1）

给定下列除式，其中包含5和7，其它打X的位置上是任意数字，请加以还原。

X7X -------商

-------- -------被除数

除数------XX|XXXXX

X77

--------

X7X

X7X

-------

XX

XX

-------

0

\*\*输出格式要求："%ld/%d=%d\n"

程序运行示例如下：

51463/53=971

参考答案

#include <stdio.h>

#include <stdlib.h>

main()

{ long int i;

int j,k,l;

for(i=10000;i<=99999;i++)

if(i%1000-i%100==400)

for(j=10;j<=99;j++)

if(i%j==0&&(l=i/j)%100>=70&&i%100<80&&i%10!=0&&l>100&&l<=999)

if(j\*(l%10)<100&&j\*(l%10)>10)

if(j\*7%100>=70&&j\*7%100<80)

if(j\*(l/100)%100==77&&j\*(l/100)>100)

printf("%ld/%d=%d\n",i,j,l);

}

Q337.(10分)求1到10的阶乘。

\*\*输出格式要求：" %2d!=%ld" 每行输出5个数的阶乘。

程序运行示例如下：

1!=1 2!=2 3!=6 4!=24 5!=120

6!=720 7!=5040 8!=40320 9!=362880 10!=3628800

参考答案

#include <stdio.h>

main()

{

long int n=1;

int i;

for(i=1;i<=10;i++)

{

n=n\*i;

printf(i%5==0?" %2d!=%ld\n":" %2d!=%ld",i,n);

}

}

Q1160.(10分)

下面程序的功能是删除字符串中第一次出现的a字符。找出其中错误，并改正之。

#include <stdio.h>

#include <string.h>

void fun(char \*x,int n,\*t);

{ int i,k=0;

x[n]='a';x[n+1]='\0';

while(x[k]!=’a’) k++;

if(k==n){\*t=0;break;}

else

{ for(i=k;i<n;i++)

x[i]=x[i+1];

x[i]='\0';

}

}

main()

{

char x[20];

int len,t;

gets(x);

puts(x);

len=strlen(x);

fun(x,len,t);

if(t==0) printf("Not exist!\n");

else puts(x);

}

参考答案

#include <stdio.h>

#include <string.h>

void fun(char \*x,int n,int \*t)

{ int i,k=0;

x[n]='a';x[n+1]='\0';

while(x[k]!='a') k++;

if(k==n){\*t=0;}

else

{ for(i=k;i<n;i++)

x[i]=x[i+1];

x[i-1]='\0';

}

}

main()

{

char x[20];

int len,t;

gets(x);

puts(x);

len=strlen(x);

fun(x,len,&t);

if(t==0) printf("Not exist!\n");

else puts(x);

}

Q275.(20分)输入某班学生某门课的成绩（最多不超过40人），

当输入为负值时，表示输入结束，

用函数编程统计成绩不低于平均分的学生人数。

要求：

(1)按如下函数原型进行编程：

int GetAboveAver(int score[], int n);

(2)在主函数中：

输入学生成绩，

然后调用函数GetAboveAver计算成绩不低于平均分的学生人数，

最后输出该人数。

要求:

(1)学生成绩和平均分均定义为int类型

(2)\*\*无输入提示信息

\*\*输入格式为："%d"

\*\*输出格式为："Students of above average is %d\n"

参考答案

#include <stdio.h>

#define N 40

int GetAboveAver(int score[], int n);

void main()

{

int score[N], m, n;

n = -1; //1

do{

n++;

scanf("%d", &score[n]);//1

}while (score[n] >= 0); //e2

m = GetAboveAver(score, n); //2

printf("Students of above average is %d\n", m);

}

int GetAboveAver(int score[], int n)

{

int i, sum, aver,count;

count = 0;//1

sum = 0;//1

for (i=0; i<n; i++)//1

{

sum += score[i];//2

}

aver=sum/n;//1

for (i=0; i<n; i++)//1

{

if (score[i] >= aver)//2

count++;//1

}

return count;//1

}

Q432.(10分)求不超过1000的回文素数。

\*\*输入格式要求：提示信息："Following are palindrome primes not greater than 1000:\n"

\*\*输出格式要求："%d\t"

程序运行示例如下：

Following are palindrome primes not greater than 1000:

11 101 131 151 181 191 313 353 373 383 727 757 787 797 919 929

参考答案

#include <stdio.h>

#include <stdlib.h>

int main()

{

int i,j,t,k,s;

printf("Following are palindrome primes not greater than 1000:\n");

for (i=0; i<=9;i++)

for(j=0;j<=9;j++)

for(k=0;k<=9;k++)

{ s=i\*100+j\*10+k;

t=k\*100+j\*10+i;

if(i==0&&j==0)

t/=100;

else if(i==0)

t/=10;

if(s>10&&s==t&&a(s))

printf("%d\t",s);

}

}

a(n)

int n;

{int i;

for (i=2;i<=(n-1)/2;i++)

if(n%i==0)

return(0);

return(1);

}

Q455.(10分)编写一个程序打印如下对称图形（行数由键盘输入1-9范围的值），例如下面是输入数字4时的情形：

4444444

33333

222

1

222

33333

4444444

参考答案

#include <stdio.h>

int main()

{

int i, j, k;

int n; /\*定义变量n，表示行数\*/

printf("Enter n(1-9):\n");

scanf("%d", &n);

for (i = 0; i < 2 \* n - 1; i++)

{

k = i;

if (i >= n)

{

k = 2 \* n - 2 - i; /\*当i >=n时，令k=2\*n-2-i，打印相应对称图形\*/

}

for (j = 0; j < k; j++)

{

printf(" "); /\*打印空格\*/

}

for (j = 0; j < 2 \* (n - k) - 1; j++)

{

printf("%d", n - k); /\*打印数字\*/

}

printf("\n");

}

}

Q206.(15分)编程计算2×3阶矩阵A和3×2阶矩阵B之积C。

要求：

(1)从键盘分别输入矩阵A和B，

输出乘积矩阵C

(2)

\*\*输入提示信息为：

输入矩阵A之前提示："Input 2\*3 matrix a:\n"

输入矩阵B之前提示："Input 3\*2 matrix b:\n"

\*\*输入矩阵中每个值的格式为："%d"

\*\*输出格式为：

输出矩阵C之前提示："Results:\n"

输出矩阵C中每个值的格式："%6d"

输出矩阵C各行结束时换行

例如：

Input 2\*3 matrix a:

1 2 3

0 1 2

Input 3\*2 matrix b:

1 -2

2 3

1 3

Results:

8 13

4 9

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

#define ROW 2

#define COL 3

main()

{

int a[ROW][COL], b[COL][ROW], c[ROW][ROW], i, j,k;

printf("Input 2\*3 matrix a:\n");

for (i=0; i<ROW ;i++)//1

{

for (j=0; j<COL; j++)//1

{

scanf("%d", &a[i][j]);//1

}

}

printf("Input 3\*2 matrix b:\n");

for (i=0; i<COL; i++)//1

{

for (j=0; j<ROW; j++)//1

{

scanf("%d", &b[i][j]);//1

}

}

for (i=0; i<ROW; i++)//1

{

for (j=0; j<ROW; j++)//1

{

c[i][j] = 0 ;//1

for (k=0; k<COL; k++)//1

{

c[i][j] = c[i][j]+a[i][k]\*b[k][j] ;//2

}

}

}

printf("Results:\n");

for (i=0; i<ROW; i++)//1

{

for (j=0; j<ROW; j++) //1

{

printf("%6d", c[i][j]);//1

}

printf("\n") ;//1

}

}

Q374.(10分)产生动态数组，编写程序，输入数组大小后，通过动态分配内存函数malloc产生数组。

\*\*输入格式要求："%d" 提示信息："Enter array size:"

\*\*输出格式要求："array[%d]=%d\n"

程序运行示例如下：

Enter array size:8

array[0]=0

array[1]=10

array[2]=20

array[3]=30

array[4]=40

array[5]=50

array[6]=60

array[7]=70

参考答案

#include <stdio.h>

#include <stdlib.h>

main()

{

int \*p,n,i;

printf("Enter array size:");

scanf("%d",&n);

p=(int \*)malloc(n\*sizeof(int));

for(i=0;i<n;i++)

\*(p+i)=i\*10;

for(i=0;i<n;i++)

printf("array[%d]=%d\n",i,\*(p+i));

}

Q1699.(10分)根据最大公约数的如下3条性质，采用迭代法编写计算最大公约数的函数Gcd()，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大公约数。

性质1 如果a>b，则a和b与a-b和b的最大公约数相同，即Gcd(a, b) = Gcd(a-b, b)

性质2 如果b>a，则a和b与a和b-a的最大公约数相同，即Gcd(a, b) = Gcd(a, b-a)

性质3 如果a=b，则a和b的最大公约数与a值和b值相同，即Gcd(a, b) = a = b

要求如下：

（1）从键盘任意输入的两整数

主函数调用Gcd()函数，并输出两整数的最大公约数。

（2）Gcd函数原型为：

int Gcd(int a, int b);

如果输入的数不是正整数，则返回-1，否则，返回两个数的最大公约数。

（3）\*\*输入提示信息格式要求："Input a,b:\n"

输入两个整数时用,号分隔

\*\*输出提示信息要求：

如果输入的数不是正整数，则输出"Input error!\n"，否则按如下格式输出"Gcd=%d\n"

参考答案

#include <stdio.h>

int Gcd(int a, int b);

int main()

{

int a, b, c;

printf("Input a,b:");

scanf("%d,%d", &a, &b);

c = Gcd(a, b);

if (c != -1)

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c);

else

printf("Input number should be positive!\n");

return 0;

}

int Gcd(int a, int b)

{

if (a <= 0 || b <= 0) return -1;

while (a != b)

{

if (a > b)

a = a - b;

else if (b > a)

b = b - a;

}

return a;

}

Q357.(10分)某班期终考试科目为高等数学MT，英语EN和物理PH，有5人参加考试。为评定奖学金，要求统计并输出一个表格，表格内容包括学号、各科分数、总分和平均分，并标出三门课均在90分以上者（该栏标志输出“Y”，否则输出“N”），表格形式如下：

NO MT EN PH SUM V >90

------------------------------------------------------

1 88 87 83 258 86 N

2 69 90 88 247 82 N

3 77 70 79 226 75 N

4 96 99 98 293 97 Y

5 77 88 90 255 85 N

\*\*输入格式要求："%d" 提示信息："Enter No. and score as: MT EN PH\n"

\*\*输出格式要求：

"NO\tMT\tEN\tPH\tSUM\tV\t>90\n"

"------------------------------------------------------\n"

"%d\t%d\t%d\t%d\t%d\t%d\t%c\n"

参考答案

#include <stdio.h>

main()

{

int i, j, a[5][4], s = 0;

char c;

for (i = 0; i <= 4; i++)

{

printf("Enter No. and score as: MT EN PH\n");

for (j = 0; j <= 3; j++)

scanf("%d", &a[i][j]);

}

printf("NO\tMT\tEN\tPH\tSUM\tV\t>90\n");

printf("------------------------------------------------------\n");

for (i = 0; i <= 4; i++)

{

printf("%d\t", a[i][0]);

for (s = 0, j = 1; j <= 3; j++)

{

s += a[i][j];

printf("%d\t", a[i][j]);

}

if (a[i][1] >= 90 && a[i][2] >= 90 && a[i][3] >= 90)

c = 'Y';

else c = 'N';

printf("%d\t%d\t%c\n", s, s / 3, c);

}

}

Q240.(10分)有5个人围坐在一起，问第5个人多大年纪，他说比第4个人大2岁；问第4个人，他说比第3个人大2岁；问第3个人，他说比第2个人大2岁；问第2个人，他说比第1个人大2岁。第1个人说自己10岁，请利用递归法编程计算并输出第5个人的年龄。

\*\*输出格式要求："The 5th person's age is %d\n"

参考答案

#include <stdio.h>

unsigned int ComputeAge(unsigned int n);

main()

{

unsigned int n = 5;

printf("The 5th person's age is %d\n", ComputeAge(n));

}

/\* 函数功能：用递归算法计算年龄 \*/

unsigned int ComputeAge(unsigned int n)

{

unsigned int age;

if (n == 1)

{

age = 10;

}

else

{

age = ComputeAge(n - 1) + 2;

}

return age;

}

Q530.(10分)用函数编写计算整数n的阶乘n！。

\*\*输入格式要求："%d" 提示信息："Input m:"

\*\*输出格式要求："%d! = %ld\n"

程序运行示例如下：

Input m:5

5! = 120

参考答案

#include <stdio.h>

long Fact(int n); /\* 函数原型声明 \*/

int main()

{

int m;

long ret;

printf("Input m:");

scanf("%d", &m);

ret = Fact(m); /\* 调用函数Fact()，并将函数的返回值存入ret \*/

printf("%d! = %ld\n", m, ret);

return 0;

}

/\* 函数功能：用迭代法计算n! \*/

long Fact(int n) /\* 函数定义 \*/

{

int i;

long result = 1;

for (i = 2; i <= n; i++)

{

result \*= i;

}

return result;

}

Q109.(15分)按如下函数原型，采用欧几里德算法编写计算两个整数的最大公约数的函数Gcd()。欧几里德算法，也称辗转相除法。其基本思想是：对正整数a和b，连续进行求余运算，直到余数为0为止，此时非0的除数就是最大公约数。

/\* 函数功能：计算a和b的最大公约数，输入负数时返回-1 \*/

int Gcd(int a, int b)

要求如下：

（1）从键盘任意输入的两整数

主函数调用Gcd()函数，并输出两整数的最大公约数。

（2）Gcd函数原型为：

int Gcd(int a, int b);

如果输入的数不是正整数，则返回-1，

否则，返回两个数的最大公约数。

（3）\*\*输入提示信息格式要求："Input a,b:\n"

输入两个整数时用,号分隔

\*\*输出提示信息要求：

如果输入的数不是正整数，则输出"Input error!\n"

否则按如下格式输出"Gcd=%d\n"

参考答案

#include <stdio.h>

int Gcd(int a, int b);

main()

{

int a, b, c;

printf("Input a,b:\n");

scanf("%d,%d", &a, &b);//1

c = Gcd(a,b);//2

if (c != -1)//1

{

printf("Gcd=%d\n", c);

}

else//1

{

printf("Input error!\n");

}

}

int Gcd(int a, int b)//1

{

int r;

if (a <= 0 || b <= 0)//2

{

return -1;//1

}

do{

r = a % b;//1

a = b;//1

b = r;//1

}while (r != 0);//e2

return a; //1

}

Q224.(10分)某人在国外留学，不熟悉当地天气预报中的华氏温度值，请编程按每隔10°输出-40°到110°之间的华氏温度到摄氏温度的对照表，以方便他对照查找。已知华氏和摄氏温度的转换公式为C=5/9\*(F-32)，其中，C表示摄氏温度，F表示华氏温度。

\*\*输出格式要求："%4d\t%6.1f\n"

程序运行示例如下：

-40 -40.0

-30 -34.4

-20 -28.9

-10 -23.3

0 -17.8

10 -12.2

20 -6.7

30 -1.1

40 4.4

50 10.0

60 15.6

70 21.1

80 26.7

90 32.2

100 37.8

110 43.3

参考答案

#include <stdio.h>

main()

{

int fahr;

float celsius;

for (fahr = -40; fahr <= 110; fahr = fahr + 10)

{

celsius = 5.0 / 9 \* (fahr - 32);

printf("%4d\t%6.1f\n", fahr, celsius);

}

}

Q1693.(10分)计算1\*2\*3 + 3\*4\*5+…+ 99\*100\*101的值

输入提示信息和格式：无

输出提示信息和格式："sum = %ld\n"

参考答案

#include <stdio.h>

main()

{

long i ;

long term, sum = 0;

for (i = 1; i <= 99; i = i + 2)

{

term = i \* (i + 1) \* (i + 2);

sum = sum + term;

}

printf("sum = %ld\n", sum);

}

Q1695.(10分)爱因斯坦曾出过这样一道数学题：有一条长阶梯，若每步跨2阶，最后剩下1阶；若每步跨3阶，最后剩下2阶；若每步跨5阶，最后剩下4阶；若每步跨6阶，则最后剩下5阶；只有每步跨7阶，最后才正好1阶不剩。参考例6.15程序，编写计算这条阶梯共有多少阶的程序如下所示，其中存在一些语法和逻辑错误，请找出并改正之，然后上机运行程序并写出程序的运行结果。

#include <stdio.h>

main()

{

int x = 1, find = 0;

while (!find);

{

if (x%2=1 && x%3=2 && x%5=4 && x%6=5 && x%7=0)

{

printf("x = %d\n", x);

find = 1;

x++;

}

}

}

参考答案

#include <stdio.h>

main()

{

int x = 1, find = 0;

while (!find)

{

if (x % 2 == 1 && x % 3 == 2 && x % 5 == 4 && x % 6 == 5 && x % 7 == 0)

{

printf("x = %d\n", x);

find = 1;

}

x++;

}

}

Q1304.(10分)有一分数序列：2/1，3/2，5/3，8/5，13/8，21/13，……求出这个数列的前20项之和。

\*\*输出格式要求：总和=%9.6f\n"

参考答案

#include <stdio.h>

int main()

{

int n, t, number = 20;

float a = 2, b = 1, s = 0;

for (n = 1; n <= number; n++)

{

s = s + a / b;

t = a;

a = a + b;

b = t;

}

printf("总和=%9.6f\n", s);

return 0;

}

Q200.(50分)按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。

其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息要求："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%d\t"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数使用void main,不用写return。

参考答案

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

void main()

{

int s[N][N], n;

printf("Input n:\n");

scanf("%d", &n);

InputMatrix(s, n);//1

Transpose(s, n);//1

printf("The transposed matrix is:\n");

PrintMatrix(s, n);//1

}

void Transpose(int a[][N], int n)

{

int i, j, temp;

for (i = 0; i < n; i++) //1

{

for (j = i; j < n; j++)//1

{

temp = a[i][j];//1

a[i][j] = a[j][i];//1

a[j][i] = temp;//1

}

}

}

void InputMatrix(int a[][N], int n) {

int i, j;

printf("Input %d\*%d matrix:\n", n, n);

for (i = 0; i < n; i++) //1

{

for (j = 0; j < n; j++)//1

{

scanf("%d", &a[i][j]);//1

}

}

}

void PrintMatrix(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j < n; j++) //1

{

printf("%d\t", a[i][j]);//1

}

printf("\n");//1

}

}

Q1625.(10分)按如下函数原型，采用梯形法编程实现，在积分区间[a,b]内计算函数y1=∫10(1+x2)dx和y2=∫30x1+x2dx的定积分。其中，指向函数的指针变量f用于接收被积函数的入口地址。

Integral(float (\*f)(float), float a, float b);

\*\*输出格式要求："y1=%f\ny2=%f\n"

参考答案

#include <stdio.h>

float Fun1(float x);

float Fun2(float x);

float Integral(float (\*f)(float), float a, float b);

int main()

{

float y1, y2;

y1 = Integral(Fun1, 0.0, 1.0);

y2 = Integral(Fun2, 0.0, 3.0);

printf("y1=%f\ny2=%f\n", y1, y2);

return 0;

}

/\* 函数功能：计算函数1+x\*x的函数值 \*/

float Fun1(float x)

{

return 1 + x \* x;

}

/\* 函数功能：计算函数x/(1+x\*x)的函数值 \*/

float Fun2(float x)

{

return x / (1 + x \* x);

}

/\* 函数功能：用梯形法计算函数的定积分 \*/

float Integral(float (\*f)(float), float a, float b)

{

float s, h;

int n = 100, i;

s = ((\*f)(a) + (\*f)(b)) / 2;

h = (b - a) / n;

for (i = 1; i < n; i++)

{

s += (\*f)(a + i \* h);

}

return s \* h;

}

Q416.(10分)该存多少钱？假设银行一年整存零取的月息为0.63%。现在某人手中有一笔钱，他打算在今后的五年中每年的年底取出1000元，到第五年时刚好取完，请算出他存钱时应存入多少。

\*\*输出格式要求："He must save %.2f at first.\n"

参考答案

#include <stdio.h>

main()

{int i;float total=0.;

for(i=0;i<5;i++)

total=(total+1000.)/(1+.0063\*12);

printf("He must save %.2f at first.\n",total);

}

Q518.(10分)编程从键盘输入n，然后计算并输出1+2+3+…+n的值。用for语句编程实现。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："sum = %d\n"

程序运行示例如下：

Input n:100

sum = 5050

参考答案

#include <stdio.h>

main()

{

int i, n, sum;

printf("Input n:");

scanf("%d", &n);

sum = 0; /\* 累加和变量初始化为0 \*/

for (i = 1; i <= n; i++)

{

sum = sum + i; /\* 做累加运算 \*/

}

printf("sum = %d\n", sum);

}

Q525.(10分)编程从键盘输入n值（10≥n≥3），然后计算并输出1! + 2! + 3! + … + n!。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："1!+2!+…+%d! = %ld\n"

程序运行示例如下：

Input n:10

1!+2!+…+10! = 4037913

参考答案

#include <stdio.h>

main()

{

int i, j, n;

long term, sum = 0; /\* 累加求和变量sum初始化为0 \*/

printf("Input n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

term = 1; /\* 累乘求积变量term赋初值为1 \*/

for (j = 1; j <= i; j++)

{

term = term \* j; /\* 累乘求积 \*/

}

sum = sum + term; /\* 累加求和 \*/

}

printf("1!+2!+…+%d! = %ld\n", n, sum);

}

Q170.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

参考答案

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q184.(15分)有一天，一位百万富翁遇到一个陌生人，陌生人找他谈一个换钱的计划，

陌生人对百万富翁说：“我每天给你10万元，而你第一天只需给我1分钱，

第二天我仍给你10万元，你给我2分钱，第三天我仍给你10万元，你给我4分钱……。

你每天给我的钱是前一天的两倍，直到满一个月（30天）为止”，

百万富翁很高兴，欣然接受了这个契约。

请编程计算在这一个月中陌生人总计给百万富翁多少钱，

百万富翁总计给陌生人多少钱。

输入格式要求：无任何输入数据或提示信息。

输出格式要求：

"Millionaire gives the stranger: %8ld Yuan\nThe stranger gives millionaire: %8ld Yuan\n"

参考答案

#include <stdio.h>

main() {

int j;

long sumToStranger = 0;//1

long sumToRichman = 0;//1

long term = 1;//1

for (j = 1; j <= 30; j++) //2

{

sumToRichman = sumToRichman + 100000;//2

sumToStranger = sumToStranger + term;//2

term = term \* 2;//2

}

sumToStranger = sumToStranger / 100;//2

printf("Millionaire gives the stranger: %8ld Yuan\nThe stranger gives millionaire: %8ld Yuan\n", sumToStranger, sumToRichman);//2

}

Q592.(10分)编写一个程序，将一个字符串s2插入到字符串s1中，其起始插入位置为n。

\*\*输入格式要求："%d" 提示信息："main string:" "sub string:" "site of begining:(<=%d)"

\*\*输出格式要求："After instert:%s\n"

程序运行示例如下：

main string:hello,world!

sub string:c

site of begining:(<=12)6

After instert:hello,c world!

参考答案

#include <stdio.h>

#include <string.h>

#define N 100

int main()

{

int n, i, j, k, len;

char s1[N], s2[N], s3[2 \* N];

printf("main string:");

gets(s1);

printf("sub string:");

gets(s2);

len = strlen(s1);

do

{

printf("site of begining:(<=%d)", len);

scanf("%d", &n);

}

while (n > len);

for (i = 0; i < n; i++)

{

s3[i] = s1[i];

}

for (j = 0; s2[j] != '\0'; j++)

{

s3[i + j] = s2[j];

}

for (k = n; s1[k] != '\0'; k++)

{

s3[j + k] = s1[k];

}

s3[j + k] = '\0';

printf("After instert:%s\n", s3);

return 0;

}

Q1721.(10分)用switch语句编程设计一个简单的计算器程序，要求根据用户从键盘输入的表达式：

操作数1 运算符op 操作数2

计算表达式的值，指定的算术运算符为加（+）、减（-）、乘（\*）、除（/）。

要求如下：

（1）要求程序能进行浮点数的算术运算

（2）要求输入的算术表达式中的操作数和运算符之间可以加入任意多个空白符

（3）要求连续做多次算术运算，每次运算结束后，程序都给出提示：

Do you want to continue（Y/N or y/n）?

用户输入Y或y时，程序继续进行其他算术运算；否则程序退出运行状态。

参考答案

#include <stdio.h>

#include <math.h>

main()

{

float data1, data2; //定义两个操作符

char op; //定义运算符

char reply; //用户输入的回答

do

{

printf("Please enter the expression:\n");

scanf("%f %c%f", &data1, &op, &data2); //%c前有一空格

switch (op) //根据输入的运算符确定要执行的运算

{

case '+':

printf("%f + %f = %f\n", data1, data2, data1 + data2);

break;

case '-':

printf("%f - %f = %f \n", data1, data2, data1 - data2);

break;

case '\*':

printf("%f \* %f = %f \n", data1, data2, data1 \* data2);

break;

case '/':

if (fabs(data2) <= 1e-7)

{

printf("Division by zero!\n");

}

else

{

printf("%f / %f = %f \n", data1, data2, data1 / data2);

}

break;

default:

printf("Unknown operator!\n");

}

printf("Do you want to continue(Y/N or y/n)?");

scanf(" %c", &reply); //%c前有一空格

}

while (reply == 'Y' || reply == 'y');

printf("Program is over!\n");

}

Q119.(10分)韩信点兵。韩信有一队兵，他想知道有多少人，便让士兵排队报数。

按从1至5报数，最末一个士兵报的数为1；

按从1至6报数，最末一个士兵报的数为5；

按从1至7报数，最末一个士兵报的数为4；

最后再按从1至11报数，最末一个士兵报的数为10。

你知道韩信至少有多少兵吗？

编写程序计算并输出结果。

\*\*输出格式要求为"x = %d\n"。

参考答案

#include <stdio.h>

main()

{

int x = 1; //1

int find = 0;

while (!find) //1

{

if (x % 5 == 1 && x % 6 == 5 && x % 7 == 4 && x % 11 == 10) //2

{

printf("x = %d\n", x);

find = 1; //1

}

x++; //1

}

}

Q203.(15分)已知不等式：1! + 2! + … + m! < n，

请编程对用户指定的n值计算并输出满足该不等式的m的整数解。

\*\*输入提示信息格式: "Please enter n:\n"

\*\*输入数据格式要求:"%lu"

\*\*输出格式要求:"m <= %lu\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

main()

{

unsigned long i, n, term = 1, sum = 0;//1

printf("Please enter n:\n");

scanf("%lu", &n);//1

for (i=1; ;i++)//4

{

term = term \* i;//2

sum = sum + term;//2

if (sum >= n) //2

break;//2

}

printf("m <= %lu\n", i-1);//1

}

Q1707.(10分)编程计算三色球问题。若一个口袋中放有12个球，其中有3个红色的，3个白色的，6个黑色的，从中任取8个球，问共有多少种不同的颜色搭配？

输入格式：无

输出格式："i=%d, j=%d, k=%d\n"

参考答案

#include <stdio.h>

main()

{

int i, j, k;

for (i = 0; i <= 3; i++)

{

for (j = 0; j <= 3; j++)

{

for (k = 0; k <= 6; k++)

{

if (i + j + k == 8)

{

printf("i=%d, j=%d, k=%d\n", i, j, k);

}

}

}

}

}

Q1680.(10分)输入m个学生（最多为30人）n门课程（最多为5门）的成绩，然后计算并打印每个学生各门课的总分和平均分。其中，m和n的值由用户从键盘输入。

标签：第11章

通过上机运行程序并观察运行结果，分析下面程序错误的原因并改正之。

#include <stdio.h>

#define STUD 30 /\* 最多可能的学生人数 \*/

#define COURSE 5 /\* 最多可能的考试科目数 \*/

void Total(int \*pScore, int sum[], float aver[], int m, int n);

void Print(int \*pScore, int sum[], float aver[], int m, int n);

int main()

{

int i, j, m, n, score[STUD][COURSE], sum[STUD];

float aver[STUD];

printf("How many students?");

scanf("%d", &m);

printf("How many courses?");

scanf("%d", &n);

printf("Input scores:\n");

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &score[i][j]);

}

}

Total(\*score, sum, aver, m, n);

Print(\*score, sum, aver, m, n);

return 0;

}

void Total(int \*pScore, int sum[], float aver[], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

sum[i] = 0;

for (j = 0; j < n; j++)

{

sum[i] = sum[i] + pScore[i \* n + j];

}

aver[i] = (float) sum[i] / n;

}

}

void Print(int \*pScore, int sum[], float aver[], int m, int n)

{

int i, j;

printf("Result:\n");

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

printf("%4d\t", pScore[i \* n + j]);

}

printf("%5d\t%6.1f\n", sum[i], aver[i]);

}

}

参考答案

#include <stdio.h>

#define STUD 30 /\* 最多可能的学生人数 \*/

#define COURSE 5 /\* 最多可能的考试科目数 \*/

void Input(int \*pScore, int m, int n);

void Total(int \*pScore, int sum[], float aver[], int m, int n);

void Print(int \*pScore, int sum[], float aver[], int m, int n);

int main()

{

int m, n, score[STUD][COURSE], sum[STUD];

float aver[STUD];

printf("How many students?");

scanf("%d", &m);

printf("How many courses?");

scanf("%d", &n);

Input(\*score, m, n);

Total(\*score, sum, aver, m, n);

Print(\*score, sum, aver, m, n);

return 0;

}

/\*函数功能：连续输入m\*n个整数存入起始地址为pScore的连续内存单元中 \*/

void Input(int \*pScore, int m, int n)

{

int i, j;

printf("Input scores:\n");

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &pScore[i \* n + j]);

}

}

}

/\* 函数功能：计算起始地址为pScore的连续内存中存储的m\*n个数组元素值的平均值 \*/

void Total(int \*pScore, int sum[], float aver[], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

sum[i] = 0;

for (j = 0; j < n; j++)

{

sum[i] = sum[i] + pScore[i \* n + j];

}

aver[i] = (float) sum[i] / n;

}

}

/\* 函数功能：打印起始地址为pScore的连续内存单元中存储的m\*n个数组元素值 \*/

void Print(int \*pScore, int sum[], float aver[], int m, int n)

{

int i, j;

printf("Result:\n");

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

printf("%4d\t", pScore[i \* n + j]);

}

printf("%5d\t%6.1f\n", sum[i], aver[i]);

}

}

Q474.(10分)假设对折一张厚度为0.1mm的纸，请问要对折多少次才能使纸的厚度从地球到达月球（假设地球到月球的距离为30万km）？

\*\*输出格式要求："%f"

参考答案

#include <stdio.h>

#include <math.h>

int main()

{

printf("%f", (log10(3.0) + 12) / log10(2.0));

return 0;

}

Q1323.(10分)写一函数，输入一个十六进制数，输出相应的十进制数。

程序的运行示例如下：

输入十六进制数：

3a

十进制数58

继续吗？

y

输入十六进制数：

2f

十进制数47

继续吗？

n

参考答案

#include <stdio.h>

#define MAX 1000

int main()

{

int c, i, flag, flag1;

char t[MAX];

i = 0;

flag = 0;

flag1 = 1;

printf("\n输入十六进制数：");

while ((c = getchar()) != '\0' && i < MAX && flag1)

{

if (c >= '0' && c <= '9' || c >= 'a' && c <= 'f' || c >= 'A' && c <= 'F')

{

flag = 1;

t[i++] = c;

}

else if (flag)

{

t[i] = '\0';

printf("\n十进制数%d\n", htoi(t));

printf("继续吗？");

c = getchar();

if (c == 'N' || c == 'n')

{

flag1 = 0;

}

else

{

flag = 0;

i = 0;

printf("\n输入十六进制数：");

}

}

}

return 0;

}

htoi(char s[])

{

int i, n;

n = 0;

for (i = 0; s[i] != '\0'; i++)

{

if (s[i] >= '0' && s[i] <= '9')

{

n = n \* 16 + s[i] - '0';

}

if (s[i] >= 'a' && s[i] <= 'f')

{

n = n \* 16 + s[i] - 'a' + 10;

}

if (s[i] >= 'A' && s[i] <= 'F')

{

n = n \* 16 + s[i] - 'A' + 10;

}

}

return n;

}

Q591.(10分)编写一个程序，判断输入的一行字符是否构成一个合法的C标识符。（假设输入的字符串非关键字且最大长度为32）

\*\*输入格式要求：getchar()

\*\*输出格式要求："is\n" "is not\n"

程序运行示例1如下：

$223

is not

程序运行示例2如下：

hello world

is not

程序运行示例3如下：

hello

is

参考答案

#include <stdio.h>

int main()

{

char c;

int first = 1, flag1 = 0, flag2 = 1, n = 0;

while ((c = getchar()) != '\n')

{

n++;

if (first == 1)

{

if ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z') || (c == '\_'))

flag1 = 1;

first = 0;

}

else if (!((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z') || (c >= '0' && c <= '9') || (c == '\_')))

flag2 = 0;

}

if (flag1 == 1 && flag2 == 1 && n <= 31)

printf("is\n");

else

printf("is not\n");

return 0;

}

Q419.(10分)波瓦松的分酒趣题。法国著名数学家波瓦松青年时代研究过一个有趣的数学问题：某人有12品脱的啤酒一瓶，想从中倒出6品脱，但他没有6品脱的容器，仅有一个8品脱和一个5品脱的容器，怎样倒才能将啤酒分为两个6品脱呢？

\*\*输入格式要求："%d%d%d%d" 提示信息："Input Full a,Empty b,c,Get i:"

\*\*输出格式要求：" a%d b%d c%d\n%4d%4d%4d\n" "%4d%4d%4d\n"

程序运行示例如下：

Input Full a,Empty b,c,Get i: 12 8 5 6

a12 b8 c5

12 0 0

4 8 0

4 3 5

9 3 0

9 0 3

1 8 3

1 6 5

6 6 0

a12 b5 c8

12 0 0

7 5 0

7 0 5

2 5 5

2 2 8

10 2 0

10 0 2

5 5 2

5 0 7

0 5 7

0 4 8

8 4 0

8 0 4

3 5 4

3 1 8

11 1 0

11 0 1

6 5 1

6 0 6

参考答案

#include <stdio.h>

int i;

main()

{ int a,y,z;

printf("Input Full a,Empty b,c,Get i:");

scanf("%d%d%d%d",&a,&y,&z,&i);

getti(a,y,z);

getti(a,z,y);

}

getti(a,y,z)

int a,y,z;

{int b=0,c=0;

printf(" a%d b%d c%d\n%4d%4d%4d\n",a,y,z,a,b,c);

while(a!=i||b!=i&&c!=i)

{if(!b)

{a-=y;b=y;}

else if(c==z)

{a+=z;c=0;}

else if(b>z-c)

{b-=(z-c);c=z;}

else {c+=b;b=0;}

printf("%4d%4d%4d\n",a,b,c);

}

}

Q120.(25分)打印所有的水仙花数。

所谓“水仙花数”是指一个三位数，其各位数字立方和等于该数本身。

例如153是“水仙花数”，因为153 = 1+125+27。

\*\*输出格式要求为"%6d"。

参考答案

#include <stdio.h>

int main()

{

int i,j,k,n;

for(n=100;n<1000;n++) //1

{

i=n/100; //1

j=(n-i\*100)/10; //1

k=n-i\*100-j\*10; //1

if(i\*i\*i+j\*j\*j+k\*k\*k==n) //1

printf("%6d",n);

}

printf("\n");

return 0;

}

Q1567.(25分)水仙花是指各位数字的立方和等于该数本身的三位数。例如153是水仙花数，因为153=1\*1\*1+5\*5\*5+3\*3\*3.请编程计算并输出所有的水仙花数。

要求输出格式为"%6d"，且所有的水仙花数字在同一行。

参考答案

#include <stdio.h>

int main()

{

int i, j, k, n;

for (n = 100; n < 1000; n++)

{

i = n / 100;

j = (n - i \* 100) / 10;

k = n % 10;

if (n == i \* i \* i + j \* j \* j + k \* k \* k)

{

printf("%6d", n);

}

}

printf("\n");

return 0;

}

Q1742.(10分)

寻找最佳存款方案。已知银行整存整取不同期限存款的月息利率分别为

假设银行对定期存款过期部分不支付利息，现在某人有2000元钱，要存20年，问怎样存才能使20年后得到的本利之和最多？

\*\*输出格式要求："8 year: %d\n" "5 year: %d\n" "3 year: %d\n" "2 year: %d\n" "1 year: %d\n" "Total: %.2f\n"

程序运行示例如下：

8 year: 0

5 year: 0

3 year: 0

2 year: 0

1 year: 20

Total: 3121.02

参考答案

#include <stdio.h>

#include <math.h>

int main(void)

{

int i8, i5, i3, i2, i1, n8, n5, n3, n2, n1;

double max = 0, total;

for (i8 = 0; i8 < 3; i8++)

for (i5 = 0; i5 <= (20 - 8 \* i8) / 5; i5++)

for (i3 = 0; i3 <= (20 - 8 \* i8 - 5 \* i5) / 3; i3++)

for (i2 = 0; i2 <= (20 - 8 \* i8 - 5 \* i5 - 3 \* i3) / 2; i2++)

{

i1 = 20 - 8 \* i8 - 5 \* i5 - 3 \* i3 - 2 \* i2;

total = 2000 \* pow(1 + 0.0225, i1)

\* pow(1 + 0.0243, i2)

\* pow(1 + 0.0270, i3)

\* pow(1 + 0.0288, i5)

\* pow(1 + 0.0300, i8);

if (total > max)

{

max = total;

n1 = i1;

n2 = i2;

n3 = i3;

n5 = i5;

n8 = i8;

}

}

printf("8 year: %d\n", n8);

printf("5 year: %d\n", n5);

printf("3 year: %d\n", n3);

printf("2 year: %d\n", n2);

printf("1 year: %d\n", n1);

printf("Total: %.2f\n", max);

return 0;

}

Q354.(10分)反向输出有5个元素的int数组中的各元素的值。数组为{10,20,30,40,50}。

\*\*输出格式要求："\ta[%d]=%d,"

程序运行示例如下：

a[4]=50, a[3]=40, a[2]=30, a[1]=20, a[0]=10,

参考答案

#include <stdio.h>

main()

{

static int a[5]={10,20,30,40,50};

int i=5;

while(--i>=0)

printf("\ta[%d]=%d,",i,a[i]);

}

Q118.(5分)在海军节开幕式上，有A、B、C三艘军舰要同时开始鸣放礼炮各21响。

已知A舰每隔5秒放1次，B舰每隔6秒放1次，C舰每隔7秒放1次。

假设各炮手对时间的掌握非常准确，

请编程计算观众总共可以听到几次礼炮声。

下面给出的程序有错误，请找到错误并改正之。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

\*\*输入输出格式要求：不要改变以下代码的输入输出格式。

#include <stdio.h>

main()

{

int n, t;

for (t=0; t<20\*7; t++);

{

if (t%5 = 0 && t < 20\*5);

{

n++;

}

if (t%6 = 0 && t < 20\*6);

{

n++;

}

if (t%7 = 0);

{

n++;

}

}

printf("n=%d\n", n);

}

参考答案

#include <stdio.h>

main()

{

int n = 0, t;

for (t=0; t<=20\*7; t++)

{

if (t%5 == 0 && t <= 20\*5)

{

n++;

continue;

}

if (t%6 == 0 && t <= 20\*6)

{

n++;

continue;

}

if (t%7 == 0)

{

n++;

}

}

printf("n=%d\n", n);

}

Q244.(10分)从键盘输入某班学生某门课的成绩和学号（最多不超过40人），当输入为负值时，表示输入结束，用函数编程通过返回数组中最大元素的下标，查找并输出成绩的最高分及其所在的学生学号。

\*\*输入格式要求："%ld%d" 提示信息："Input student’s ID and score:" "input error!\n"

\*\*输出格式要求："Total students are %d\n" "The highest is:%ld, %d\n"

程序运行示例如下：

Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

The highest is:70310124, 88

参考答案

#include <stdio.h>

#define N 40

int ReadScore(int score[], long num[]);

int FindMax(int score[], int n);

int main()

{

int score[N], maxNum, n;

long num[N];

n = ReadScore(score, num); /\* 输入成绩，返回学生人数 \*/

printf("Total students are %d\n", n);

maxNum = FindMax(score, n); /\* 计算并返回最高分所在数组的下标 \*/

printf("The highest is:%ld, %d\n", num[maxNum], score[maxNum]);

return 0;

}

/\* 函数功能：输入学生某门课的成绩，当输入负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[], long num[])

{

int i = -1;

do

{

i++;

printf("Input student’s ID and score:");

scanf("%ld%d", &num[i], &score[i]);

}

while (score[i] >= 0 && num[i] >= 0);

return i;

}

/\* 函数功能：计算并返回最高分所在数组的下标 \*/

int FindMax(int score[], int n)

{

int max, i, maxNum = 0;

max = score[0];

for (i = 1; i < n; i++)

{

if (score[i] > max)

{

max = score[i];

maxNum = i;

}

}

return maxNum;

}

Q340.(10分)从键盘输入任意的字符，按下列规则进行分类计数。

第一类：‘0’， ‘1’， ‘2’， ‘3’， ‘4’， ‘5’， ‘6’， ‘7’， ‘8’， ‘9’

第二类：‘+’， ‘-’， ‘\*’， ‘/’， ‘%’， ‘=’

第三类：其它字符。

\*\*输出格式要求："class1=%d, class2=%d, class3=%d\n"

程序运行示例如下：

ghdf^%^#$^&(+-//+\_8\*(\

class1=1, class2=7, class3=14

参考答案

#include <stdio.h>

main()

{

int class1,class2,class3;

char ch;

class1=class2=class3=0;

do

{

ch=getchar();

switch(ch)

{

case'0':case'1':case'2':case'3':case'4':

case'5':case'6':case'7':case'8':case'9':

class1++;break;

case'+':case'-':case'\*':case'/':case'%':case'=':

class2++;break;

default:class3++;break;

}

}while(ch!='\\');

printf("class1=%d, class2=%d, class3=%d\n",class1,class2,class3);

}

Q349.(10分)求组合数。编程计算：

C(m,n)=m!n!(m−n)!

\*\*输入格式要求："%d%d" 提示信息："Input m and n="

\*\*输出格式要求："The combination is %ld\n"

程序运行示例如下：

Input m and n=10 5

The combination is 252

参考答案

#include <stdio.h>

main()

{

int m,n;

long cmn,facto();

printf("Input m and n=");

scanf("%d%d",&m,&n);

cmn=facto(m)/(facto(n)\*facto(m-n));

printf("The combination is %ld\n",cmn);

}

long facto(x)

int x;

{

long y;

for(y=1;x>0;--x)

y\*=x;

return y;

}

Q531.(10分)编写计算组合数的程序。要求输入数据要有容错功能。

\*\*输入格式要求："%d,%d" 提示信息："Input m,k (m>=k>0):"

\*\*输出格式要求："p = %.0f\n"

程序运行示例1如下：

Input m,k (m>=k>0):3,2

p = 3

程序运行示例2如下：

Input m,k (m>=k>0):2,3

Input m,k (m>=k>0):3,3

p = 1

Input m,k (m>=k>0):-2,-4

Input m,k (m>=k>0):4,2

p = 6

参考答案

#include <stdio.h>

unsigned long Fact(unsigned int n);

int main()

{

int m, k;

double p;

do

{

printf("Input m,k (m>=k>0):");

scanf("%d,%d", &m, &k);

}

while (m < k || m < 0 || k < 0);

p = (double)Fact(m) / (Fact(k) \* Fact(m - k));

printf("p = %.0f\n", p);

return 0;

}

/\* 函数功能：用迭代法计算无符号整型变量n的阶乘，当n>=0时返回n!；否则返回-1 \*/

unsigned long Fact(unsigned int n)

{

unsigned int i;

unsigned long result = 1;

for (i = 2; i <= n; i++)

result \*= i;

return result;

}

Q532.(10分)用递归方法计算整数n的阶乘n！。

\*\*输入格式要求："%d" 提示信息："Input n:" "n<0, data error!\n"

\*\*输出格式要求："%d! = %ld\n"

程序运行示例如下：

Input n:5

5! = 120

参考答案

#include <stdio.h>

long Fact(int n);

int main()

{

int n;

long result;

printf("Input n:");

scanf("%d", &n);

result = Fact(n); /\* 调用递归函数Fact()计算n! \*/

if (result == -1) /\* 处理非法数据 \*/

printf("n<0, data error!\n");

else /\* 输出n!值 \*/

printf("%d! = %ld\n", n, result);

return 0;

}

/\* 函数功能：用递归法计算长整型变量n的阶乘，当n>=0时返回n!，否则返回-1 \*/

long Fact(int n)

{

long result = 1;

if (n < 0) /\* 处理非法数据 \*/

return -1;

else if (n == 0 || n == 1) /\* 基线情况，即递归终止条件 \*/

return 1;

else /\* 一般情况 \*/

return (n \* Fact(n - 1)); /\* 递归调用，利用(n-1)!计算n! \*/

}

Q302.(15分)编程打印如下形式的杨辉三角形。打印的杨辉三角形的行数n（不超过15行）。要求由用户在主函数中通过键盘输入。请按照如下给定的函数原型进行编程。函数CaculateYH用于生成杨辉三角形，函数PrintYH用于打印杨辉三角形。给定的函数原型如下:

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

\*\*输入提示信息\*\*："Input n(n<15):\n"

\*\*输入数据格式\*\*："%d"

\*\*输出提示信息\*\*：无

\*\*输出数据格式\*\*："%5d"

杨辉三角形形式如下：

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

参考答案

#include<stdio.h>

#define N 15

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

int main()

{

int a[N][N] = {0}, n;//1

printf("Input n(n<15):\n");

scanf("%d", &n);//1

CaculateYH(a, n);//1

PrintYH(a, n); //1

return 0;

}

void CaculateYH(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

a[i][0] = 1;//1

a[i][i] = 1;//1

}

for (i = 2; i < n; i++)//1

{

for (j = 1; j <= i - 1; j++) //1

{

a[i][j] = a[i - 1][j - 1] + a[i - 1][j];//1

}

}

}

void PrintYH(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j <= i; j++) //1

{

printf("%5d", a[i][j]);//1

}

printf("\n");//1

}

}

Q237.(10分)利用求阶乘函数Fact()，编程计算并输出从1到n之间所有数的阶乘值。

\*\*输入格式要求："%u" 提示信息："Input n(n>0):"

\*\*输出格式要求："%d! = %lu\n"

程序运行示例如下：

Input n(n>0):10

1! = 1

2! = 2

3! = 6

4! = 24

5! = 120

6! = 720

7! = 5040

8! = 40320

9! = 362880

10! = 3628800

参考答案

#include <stdio.h>

unsigned long Fact(unsigned int n);

int main()

{

unsigned int i, n;

printf("Input n(n>0):");

scanf("%u", &n);

for (i = 1; i <= n; i++)

{

printf("%d! = %lu\n", i, Fact(i));

}

return 0;

}

/\* 函数功能：用迭代法计算无符号整型变量n的阶乘 \*/

unsigned long Fact(unsigned int n)

{

unsigned int i;

unsigned long result = 1;

for (i = 2; i <= n; i++)

result \*= i;

return result;

}

Q1719.(10分)从键盘任意输入一个整数n，编程计算并输出1~n之间的所有素数之和

输入提示信息："Input n:"

输入格式："%d"

输出格式："sum = %d\n"

参考答案

#include <stdio.h>

#include <math.h>

int IsPrime(int x);

int main()

{

int m, n, sum = 0;

printf("Input n:");

scanf("%d", &n);

for (m = 1; m <= n; m++)

{

if (IsPrime(m)) /\* 素数判定 \*/

{

sum += m;

}

}

printf("sum = %d\n", sum);

return 0;

}

/\* 函数功能：判断x是否是素数，若函数返回0，则表示不是素数，若返回1，则代表是素数 \*/

int IsPrime(int x)

{

int i, flag = 1;

int squareRoot = (int)sqrt(x);

if (x <= 1) flag = 0; /\* 负数、0和1都不是素数 \*/

for (i = 2; i <= squareRoot && flag; i++)

{

if (x % i == 0) flag = 0; /\* 若能被整除，则不是素数 \*/

}

return flag;

}

Q256.(10分)

按如下函数原型，采用梯形法编程实现，在积分区间[a,b]内计算函数y1=∫10(1+x2)dx和y2=∫30x1+x2dx的定积分。其中，指向函数的指针变量f用于接收被积函数的入口地址。 Integral(float (\*f)(float), float a, float b);

\*\*输出格式要求："y1=%f\ny2=%f\n"

参考答案

#include <stdio.h>

float Fun1(float x);

float Fun2(float x);

float Integral(float (\*f)(float), float a, float b);

int main()

{

float y1, y2;

y1 = Integral(Fun2, 0.0, 3.0);

y2 = Integral(Fun1, 0.0, 1.0);

printf("y1=%f\ny2=%f\n", y1, y2);

return 0;

}

/\* 函数功能：计算函数1+x\*x的函数值 \*/

float Fun1(float x)

{

return 1 + x \* x;

}

/\* 函数功能：计算函数x/(1+x\*x)的函数值 \*/

float Fun2(float x)

{

return x / (1 + x \* x);

}

/\* 函数功能：用梯形法计算函数的定积分 \*/

float Integral(float (\*f)(float), float a, float b)

{

float s, h;

int n = 100, i;

s = ((\*f)(a) + (\*f)(b)) / 2;

h = (b - a) / n;

for (i = 1; i < n; i++)

{

s += (\*f)(a + i \* h);

}

return s \* h;

}

Q579.(10分)输入某年某月某日，判断这一天是这一年的第几天？程序分析：以3月5日为例，应该先把前两个月的加起来，然后再加上5天即本年的第几天，特殊情况，闰年且输入月份大于3时需考虑多加一天。

\*\*输入格式要求："%d,%d,%d" 提示信息："\nplease input year,month,day\n"

\*\*输出格式要求："It is the %dth day." "data error"

程序运行示例如下：

please input year,month,day

2011,11,22

It is the 327th day.

参考答案

int main()

{

int day, month, year, sum, leap;

printf("\nplease input year,month,day\n");

scanf("%d,%d,%d", &year, &month, &day);

switch (month)/\*先计算某月以前月份的总天数\*/

{

case 1:

sum = 0;

break;

case 2:

sum = 31;

break;

case 3:

sum = 59;

break;

case 4:

sum = 90;

break;

case 5:

sum = 120;

break;

case 6:

sum = 151;

break;

case 7:

sum = 181;

break;

case 8:

sum = 212;

break;

case 9:

sum = 243;

break;

case 10:

sum = 273;

break;

case 11:

sum = 304;

break;

case 12:

sum = 334;

break;

default:

printf("data error");

break;

}

sum = sum + day;

/\*再加上某天的天数\*/

if (year % 400 == 0 || (year % 4 == 0 && year % 100 != 0))

/\*判断是不是闰年\*/

leap = 1;

else

leap = 0;

if (leap == 1 && month > 2)/\*如果是闰年且月份大于2,总天数应该加一天\*/ sum++;

printf("It is the %dth day.", sum);

}

Q1513.(20分)从键盘输入一个小写英文字母,将其转换成大写英文字母后,输出大写英文字母及其对应的十进制ASCII码值.

\*\*输入格式要求：无任何信息提示；

\*\*输出格式要求："%c,%d\n"

参考答案

#include<stdio.h>

int main()

{

char ch;

ch = getchar();

ch = ch - 32;

printf("%c,%d\n", ch, ch);

return 0;

}

Q619.(10分)某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下： 每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

\*\*输入格式要求："%d"

\*\*输出格式要求："%d"

程序运行示例如下：

输入:3421

输出:8976

参考答案

int main()

{

int i, aa[4], t;

for (i = 0; i <= 3; i++)

{

scanf("%d", &aa[i]);

}

for (i = 0; i <= 3; i++)

{

aa[i] += 5;

aa[i] %= 10;

}

for (i = 0; i <= 3 / 2; i++)

{

t = aa[i];

aa[i] = aa[3 - i];

aa[3 - i] = t;

}

for (i = 3; i >= 0; i--)

printf("%d", aa[i]);

}

Q581.(10分)打印出如下图案（菱形）

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

参考答案

int main()

{

int i, j, k;

for (i = 0; i <= 3; i++)

{

for (j = 0; j <= 2 - i; j++)

printf(" ");

for (k = 0; k <= 2 \* i; k++)

printf("\*");

printf("\n");

}

for (i = 0; i <= 2; i++)

{

for (j = 0; j <= i; j++)

printf(" ");

for (k = 0; k <= 4 - 2 \* i; k++)

printf("\*");

printf("\n");

}

}

Q248.(10分)兔子生崽问题。假设一对小兔的成熟期是一个月，即一个月可长成成兔，那么如果每对成兔每个月都可以生一对小兔，一对新生的小兔从第二个月起就开始生兔子，试问从一对兔子开始繁殖，一年以后可有多少对兔子？请编程求解该问题。

\*\*输出格式要求："\nTotal = %d\n" "%4d"

程序运行示例如下：

1 2 3 5 8 13 21 34 55 89 144 233

Total = 233

参考答案

#include <stdio.h>

#define N 12

void Fibonacci(int f[], int n);

main()

{

int f[N], i;

Fibonacci(f, N);

printf("\nTotal = %d\n", f[N - 1]);

}

/\* 函数功能：计算并打印Fibonacci数列的前n项 \*/

void Fibonacci(int f[], int n)

{

int i;

f[0] = 1;

f[1] = 2;

for (i = 2; i < n; i++)

{

f[i] = f[i - 1] + f[i - 2];

}

for (i = 0; i < N; i++)

{

printf("%4d", f[i]);

}

}

Q386.(15分)编程计算：3+33+333+...+33...3(n个3)的值，n值由键盘输入。

\*\*\*输入提示信息\*\*\*："Input n:\n"

\*\*\*输入数据格式\*\*\*："%d"

\*\*\*输出数据格式\*\*\*："sum=%ld\n"

注：不允许使用goto语句

参考答案

#include<stdio.h>

main()

{

int i, n;

long term, sum = 0; //1

printf("Input n:\n");

scanf("%d", &n); //1

for (i = 1, term = 3; i <= n; i++) //4

{

sum = sum + term; //2

term = term \* 10 + 3; //2

}

printf("sum=%ld\n", sum); //1

}

Q421.(10分)黑白子交换。有三个白子和三个黑子如下图布置：(-表示一个空格)

○ ○ ○ - ● ● ●

游戏的目的是用多少的步数将上图中白子和黑子的位置进行交换：

● ● ● - ○ ○ ○

游戏的规则是：一次只能移动一个棋子；棋子可以向空格中移动，也可以跳过一个对方的格子进入空格，但不能向后跳，也不能跳过两个子。请用计算机实现上述游戏。

程序运行示例如下：

No. 0:-----------

|\*|\*|\*| |@|@|@|

-----------

No. 1:-----------

|\*|\*| |\*|@|@|@|

-----------

No. 2:-----------

|\*|\*|@|\*| |@|@|

-----------

No. 3:-----------

|\*|\*|@|\*|@| |@|

-----------

No. 4:-----------

|\*|\*|@| |@|\*|@|

-----------

No. 5:-----------

|\*| |@|\*|@|\*|@|

-----------

No. 6:-----------

| |\*|@|\*|@|\*|@|

-----------

No. 7:-----------

|@|\*| |\*|@|\*|@|

-----------

No. 8:-----------

|@|\*|@|\*| |\*|@|

-----------

No. 9:-----------

|@|\*|@|\*|@|\*| |

-----------

No.10:-----------

|@|\*|@|\*|@| |\*|

-----------

No.11:-----------

|@|\*|@| |@|\*|\*|

-----------

No.12:-----------

|@| |@|\*|@|\*|\*|

-----------

No.13:-----------

|@|@| |\*|@|\*|\*|

-----------

No.14:-----------

|@|@|@|\*| |\*|\*|

-----------

No.15:-----------

|@|@|@| |\*|\*|\*|

-----------

参考答案

int number;

print(a)

int a[];

{

int i;

printf("No.%2d:-----------\n",number++);

printf(" ");

for(i=0;i<=6;i++)

printf("|%c",a[i]==1?'\*':(a[i]==2?'@':' '));

printf("|\n -----------\n");

}

main()

{

int t[7]={1,1,1,0,2,2,2};

int i,flag;

print(t);

while(t[0]+t[1]+t[2]!=6||t[4]+t[5]+t[6]!=3)

{

flag=1;

for(i=0;flag&&i<5;i++)

if(t[i]==1&&t[i+1]==2&&t[i+2]==0)

{change(&t[i],&t[i+2]);print(t);flag=0;}

for(i=0;flag&&i<5;i++)

if(t[i]==0&&t[i+1]==1&&t[i+2]==2)

{change(&t[i],&t[i+2]);print(t);flag=0;}

for(i=0;flag&&i<6;i++)

if(t[i]==1&&t[i+1]==0&&(i==0||t[i-1]!=t[i+2]))

{change(&t[i],&t[i+1]);print(t);flag=0;}

for(i=0;flag&&i<6;i++)

if(t[i]==0&&t[i+1]==2&&(i==5||t[i-1]!=t[i+2]))

{change(&t[i],&t[i+1]);print(t);flag=0;}

}

}

change(n,m)

int \*n,\*m;

{

int term;

term=\*n;\*n=\*m;\*m=term;

}

Q246.(10分)输入n×n阶矩阵，用函数编程计算并输出其两条对角线上的各元素之和。

\*\*输入格式要求："%d" 提示信息："Input n:" "Input %d\*%d matrix:\n"

\*\*输出格式要求："sum = %d\n" "%6d"

程序运行示例如下：

Input n:5

Input 5\*5 matrix:

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7

4 5 6 7 8

5 6 7 8 9

sum = 45

参考答案

#include <stdio.h>

#define N 10

void InputMatrix(int a[N][N], int n);

int AddDiagonal(int a[N][N], int n);

int main()

{

int a[N][N], n, sum;

printf("Input n:");

scanf("%d", &n);

InputMatrix(a, n);

sum = AddDiagonal(a, n);

printf("sum = %d\n", sum);

return 0;

}

/\* 函数功能： 输入n×n矩阵的元素值，存于数组a中 \*/

void InputMatrix(int a[N][N], int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", n, n);

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i][j]);

}

}

}

/\* 函数功能： 计算n×n矩阵中两条对角线上的元素之和 \*/

int AddDiagonal(int a[N][N], int n)

{

int i, j, sum = 0;

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

if (i == j || i + j == n - 1) sum = sum + a[i][j];

}

}

return sum;

}

Q199.(50分)从键盘输入n个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。其中，n由用户从键盘输入。已知n值不超过10。

要求按如下函数原型编写程序

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

\*\*输入提示信息要求：

"Input n(n<=10):\n"

\*\*要求输入格式为："%d"

\*\*输出提示信息：

"Input %d numbers:\n"

"Exchange results:"

\*\*要求输出格式为："%5d"

输出数组中所有数据后换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

int main()

{

int a[10], n;

printf("Input n(n<=10):\n");

scanf("%d", &n);

printf("Input %d numbers:\n", n);

ReadData(a, n);//1

MaxMinExchang(a, n);//1

printf("Exchange results:");

PrintData(a, n);//1

return 0;

}

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++) //1

{

scanf("%d", &a[i]);

}

}

void PrintData(int a[], int n) {

int i;

for (i = 0; i < n; i++)//1

{

printf("%5d", a[i]);//1

}

printf("\n");//1

}

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;//1

int i, temp;

for (i = 1; i < n; i++) //1

{

if (a[i] > maxValue) //1

{

maxValue = a[i];//1

maxPos = i;//1

}

if (a[i] < minValue) //1

{

minValue = a[i];

minPos = i;

}

}

temp = a[maxPos];//1

a[maxPos] = a[minPos];//1

a[minPos] = temp;//1

}

Q299.(15分)用二维字符数组方法编程实现从键盘任意输入10个字符串，将这10个字符串按字典顺序排序后打印输出，字符串最大长度80。

\*\*输入提示信息格式: 无

\*\*输入数据格式要求: 要求读入的字符串可以包含有空格

\*\*输出数据格式要求: 一行输出一个字符串

注:不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include<stdio.h>

#include<string.h>

main()

{

char a[10][80],term[80]; //1

int i,j;

for(i=0;i<10;i++) //1

{

gets(a[i]); //1

}

for(i=0;i<10-1;i++) //2

{

for(j=i+1;j<10;j++) //2

{

if(strcmp(a[i],a[j])>0) //1

{

strcpy(term,a[i]); //1

strcpy(a[i],a[j]); //1

strcpy(a[j],term); //1

}

}

}

for(i=0;i<10;i++) //1

{

puts(a[i]); //1

}

}

Q134.(5分)从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str)==0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

参考答案

#include "stdio.h"

#include "string.h"

main()

{

int i=0, findFlag=1;

char x[13];

char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str[i])==0)

{

findFlag = 0;

}

i++;

}

if ( !findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

Q347.(10分)比较两个整数的大小。

\*\*输入格式要求："%d%d" 提示信息："Enter integer X and Y:"

\*\*输出格式要求："X>Y\n" "X<Y\n" "X=Y\n"

程序运行示例如下：

Enter integer X and Y:5 6

X<Y

参考答案

#include <stdio.h>

main()

{

int x,y;

printf("Enter integer X and Y:");

scanf("%d%d",&x,&y);

if(x!=y)

if(x>y) printf("X>Y\n");

else printf("X<Y\n");

else

printf("X=Y\n");

}

Q494.(10分)用数组实现7阶魔方矩阵。所谓的N阶魔方矩阵是指把1～N\*N的自然数按一定方法排列成N\*N的矩阵，使得：任意行、任意列以及两个对角线上的数之和都相等（N为奇数）。例如下面的5阶魔方矩阵，任意行、任意列以及两个对角线上的数之和都为65。

5阶魔方如下：

17 24 1 8 15

23 5 7 14 16

4 6 13 20 22

10 12 19 21 3

11 18 25 2 9

奇数阶魔方矩阵的算法如下：

第1步：将1放入第一行的正中处。

第2步：按如下的方法依次将第i个数（i从2到N\*N）放到合适的位置上。

 如果第i-1个数的右上位置没有放数，则将第i个数放到前一个数的右上位置。

 如果第i-1个数的右上位置已经有数，则将第i个数放到第i-1个数的下一行，列数相同的位置。

\*\*输出格式要求："\n%d阶魔方矩阵如下：\n" "%4d"

程序运行结果如下：

7阶魔方矩阵如下：

================================

30 39 48 1 10 19 28

38 47 7 9 18 27 29

46 6 8 17 26 35 37

5 14 16 25 34 36 45

13 15 24 33 42 44 4

21 23 32 41 43 3 12

22 31 40 49 2 11 20

参考答案

#include <stdio.h>

#define N 7

int main()

{

int matrix[N][N];

int row, col;

int i, m, n;

//0. 初始化数组元素全为0

for (row = 0; row < N; row++)

{

for (col = 0; col < N; col++)

{

matrix[row][col] = 0;

}

}

//1. 定位1的初始位置

row = 0;

col = (N - 1) / 2;

matrix[row][col] = 1;

//2. 将2..N\*N每个数插入到相应位置

for (i = 2; i <= N \* N; i++)

{

int r, c;

r = row;

c = col;

row = (row + N - 1) % N; //行列坐标的计算，算法难点

col = (col + N + 1) % N;

if (0 == matrix[row][col]) //该处有数则冲突，处理到前一个数的下一列

matrix[row][col] = i; //无冲突则插入该数。

else

{

r = (r + 1) % N;

matrix[r][c] = i;

row = r;

col = c;

}

}

//3. 输出魔方矩阵

printf("\n%d阶魔方矩阵如下：\n", N);

for (m = 0; m <= N; m++)

{

printf("====");

}

printf("\n");

for (m = 0; m < N; m++)

{

for (n = 0; n < N; n++)

printf("%4d", matrix[m][n]);

printf("\n");

}

for (m = 0; m <= N; m++)

{

printf("====");

}

printf("\n");

return 0;

}

Q1609.(10分)按如下函数原型编写程序，输入n×n阶矩阵，用函数编程计算并输出其两条对角线上的各元素之和。

void InputMatrix(int a[N][N], int n);

int AddDiagonal(int a[N][N], int n);

输入提示信息：

"Input n:"

"Input %d\*%d matrix:\n"

输入格式："%d"

输出提示信息和输出格式："sum = %d\n"

参考答案

#include <stdio.h>

#define N 10

void InputMatrix(int a[N][N], int n);

int AddDiagonal(int a[N][N], int n);

int main()

{

int a[N][N], n, sum;

printf("Input n:");

scanf("%d", &n);

InputMatrix(a, n);

sum = AddDiagonal(a, n);

printf("sum = %d\n", sum);

return 0;

}

/\* 函数功能： 输入n×n矩阵的元素值，存于数组a中 \*/

void InputMatrix(int a[N][N], int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", n, n);

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i][j]);

}

}

}

/\* 函数功能： 计算n×n矩阵中两条对角线上的元素之和 \*/

int AddDiagonal(int a[N][N], int n)

{

int i, j, sum = 0;

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

if (i == j || i + j == n - 1)

sum = sum + a[i][j];

}

}

return sum;

}

Q293.(5分)改错题：设计一个函数MinCommonMultiple,其功能是计算两个正整数的最小公倍数。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include <stdio.h>

int MinCommonMultiple(int a,int b)

main()

{

int a,b,x;

printf("Input a,b:");

scanf("%d,%d",a,b);

x = MinCommonMultiple(a,b);

printf("MinCommonMultiple = %d\n", x);

}

int MinCommonMultiple(int a,int b);

{

int i;

for(i=1; i<a\*b; i++)

{

if(i%a == 0 || i%b == 0)

return i;

}

}

参考答案

int MinCommonMultiple(int a,int b);

main()

{

int a,b,x;

printf("Input a,b:");

scanf("%d,%d",&a,&b);

x = MinCommonMultiple(a,b);

printf("MinCommonMultiple = %d\n", x);

}

int MinCommonMultiple(int a,int b)

{

int i;

for(i=1; i<a\*b; i++)

{

if(i%a == 0 && i%b == 0)

return i;

}

}

Q245.(10分)假设有40个学生被邀请来给餐厅的饮食和服务质量打分，分数划分为1～10这10个等级（1表示最低分，10表示最高分），编程统计并按如下格式输出餐饮服务质量调查结果。

Grade Count Histogram

1 5 \*\*\*\*\*

2 10 \*\*\*\*\*\*\*\*\*\*

3 7 \*\*\*\*\*\*\*

...

\*\*输入格式要求："%d" 提示信息："Input the feedbacks of 40 students:\n" "input error!\n"

\*\*输出格式要求："Feedback\tCount\tHistogram\n" "%8d\t%5d\t"

程序运行示例如下：

Input the feedbacks of 40 students:

10 9 10 8 7 6 5 10 9 8

8 9 7 6 10 9 8 8 7 7

6 6 8 8 9 9 10 8 7 7

9 8 7 9 7 6 5 9 8 7

Feedback Count Histogram

1 0

2 0

3 0

4 0

5 2 \*\*

6 5 \*\*\*\*\*

7 9 \*\*\*\*\*\*\*\*\*

8 10 \*\*\*\*\*\*\*\*\*\*

9 9 \*\*\*\*\*\*\*\*\*

10 5 \*\*\*\*\*

参考答案

#include <stdio.h>

#define M 40

#define N 11

int main()

{

int i, j, grade, feedback[M], count[N] = {0};

printf("Input the feedbacks of 40 students:\n");

for (i = 0; i < M; i++)

{

scanf("%d", &feedback[i]);

}

for (i = 0; i < M; i++)

{

switch (feedback[i])

{

case 1:

count[1]++;

break;

case 2:

count[2]++;

break;

case 3:

count[3]++;

break;

case 4:

count[4]++;

break;

case 5:

count[5]++;

break;

case 6:

count[6]++;

break;

case 7:

count[7]++;

break;

case 8:

count[8]++;

break;

case 9:

count[9]++;

break;

case 10:

count[10]++;

break;

default:

printf("input error!\n");

break;

}

}

printf("Feedback\tCount\tHistogram\n");

for (grade = 1; grade <= N - 1; grade++)

{

printf("%8d\t%5d\t", grade, count[grade]);

for (j = 0; j < count[grade]; j++)

{

printf("%c", '\*');

}

printf("\n");

}

return 0;

}

Q535.(10分)用递归方法编程计算输出Fibonacci数列，同时打印出计算Fibonacci数列每一项时所需的递归调用次数。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："Fib(%d)=%d, count=%d\n"

程序运行示例如下：

Input n:10

Fib(1)=1, count=1

Fib(2)=1, count=3

Fib(3)=2, count=5

Fib(4)=3, count=9

Fib(5)=5, count=15

Fib(6)=8, count=25

Fib(7)=13, count=41

Fib(8)=21, count=67

Fib(9)=34, count=109

Fib(10)=55, count=177

参考答案

#include <stdio.h>

long Fib(int a);

int count; /\*全局变量count用于累计递归函数被调用的次数，自动初始化为0\*/

int main()

{

int n, i, x;

printf("Input n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

count = 0; /\* 计算下一项Fibonacci数列时将计数器count清零 \*/

x = Fib(i);

printf("Fib(%d)=%d, count=%d\n", i, x, count);

}

return 0;

}

/\* 函数功能：用递归法计算Fibonacci数列中的第n项的值 \*/

long Fib(int n)

{

long f;

count++; /\* 累计递归函数被调用的次数，记录于全局变量count中 \*/

if (n == 0) f = 0;

else if (n == 1) f = 1;

else f = Fib(n - 1) + Fib(n - 2);

return f;

}

Q1317.(10分)写一个函数，输入一行字符，将此字符串中最长的单词输出。

程序的运行示例如下：

输入一行文本：I am a student.

最长的单词是：student

参考答案

#include <stdio.h>

int alphabetic(char c)

{

if ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z'))

{

return 1;

}

else

{

return 0;

}

}

/\* 寻找最长单词 \*/

int longest(char string[])

{

int len = 0, i, length = 0, flag = 1, place, point;

for (i = 0; i <= strlen(string); i++)

{

if (alphabetic(string[i]))

{

if (flag)

{

point = i;

flag = 0;

}

else

{

len++;

}

}

else

{

flag = 1;

if (len > length)

{

length = len;

place = point;

len = 0;

}

}

}

return place;

}

int main()

{

int i;

char line[100];

printf("输入一行文本：\n");

gets(line);

printf("\n最长的单词是：");

for (i = longest(line); alphabetic(line[i]); i++)

{

printf("%c", line[i]);

}

printf("\n");

return 0;

}

Q1309.(10分)用二分法求方程2x^3-4x^2+3x-6=0在(-10, 10)之间的根。

\*\*输入格式要求："%f,%f" 提示信息："请输入x1，x2的值："

\*\*输出格式要求："方程的根=%6.2f\n"

程序的运行示例如下：

请输入x1，x2的值：-10,10

方程的根= 2.00

参考答案

#include <stdio.h>

#include <math.h>

int main()

{

float x0, x1, x2, fx0, fx1, fx2;

do

{

printf("请输入x1，x2的值：");

scanf("%f,%f", &x1, &x2);

fx1 = x1 \* ((2 \* x1 - 4) \* x1 + 3) - 6;

fx2 = x2 \* ((2 \* x2 - 4) \* x2 + 3) - 6;

}

while (fx1 \* fx2 > 0);

do

{

x0 = (x1 + x2) / 2;

fx0 = x0 \* ((2 \* x0 - 4) \* x0 + 3) - 6;

if (fx0 \* fx1 < 0)

{

x2 = x0;

fx2 = fx0;

}

else

{

x1 = x0;

fx1 = fx0;

}

}

while (fabs(fx0) >= 1e-5);

printf("方程的根=%6.2f\n", x0);

return 0;

}

Q233.(10分)一辆卡车违反了交通规则，撞人后逃逸。现场有三人目击该事件，但都没有记住车号，只记住车号的一些特征。甲说：车号的前两位数字是相同的；乙说：车号的后两位数字是相同的，但与前两位不同；丙是位数学家，他说：4位的车号正好是一个整数的平方。请根据以上线索编程协助警方找出车号，以便尽快破案，抓住交通肇事犯。

\*\*输出格式要求："k=%d, m=%d\n"

参考答案

#include <stdio.h>

main()

{

int i, j, k, m;

for (i = 0; i <= 9; i++)

{

for (j = 0; j <= 9; j++)

{

if (i != j)

{

k = i \* 1000 + i \* 100 + j \* 10 + j;

for (m = 31; m \* m <= k; m++)

{

if (m \* m == k)

{

printf("k=%d, m=%d\n", k, m);

}

}

}

}

}

}

Q548.(15分)从键盘任意输入一个字符串，计算其实际字符个数并打印输出，即不使用字符串处理函数strlen(),编程实现strlen()的功能。

\*\*输入格式要求："%s" 提示信息："Please enter a string:"

\*\*输出格式要求："The length of the string is: %u\n"

程序的运行示例如下：

Please enter a string:Hello China

The length of the string is: 11

参考答案

#include <stdio.h>

unsigned int MyStrlen(char str[]);

/\* 函数功能：用字符型数组作函数参数，计算字符串的长度 \*/

unsigned int MyStrlen(char str[])

{

int i ;

unsigned int len = 0; /\* 计数器置0 \*/

for (i = 0; str[i] != '\0'; i++)

{

len++; /\* 利用循环统计不包括’\0’在内的字符个数 \*/

}

return (len); /\* 返回实际字符个数 \*/

}

int main()

{

char a[80];

unsigned int len;

printf("Please enter a string:");

gets(a);

len = MyStrlen(a); /\* 计算字符串实际字符个数 \*/

printf("The length of the string is: %u\n", len);

return 0;

}

Q1383.(10分)以下程序的功能是从键盘输入10个整型数据，放入数组a 中，求其最大值、最小值及其所在元素的下标位置并输出。找出其中错误并改正之。

#include <stdio.h>

main()

{

int a[10], n, max, min, maxPos, minPos;

for (n=0, n<10, n++)

{

scanf("%d", a[n]);

}

max = min = a;

maxPos = minPos = 0;

for (n=1; n<10; n++)

{

if (a[n] > max)

{

max = a[n];

maxPos = n;

}

else if (a[n] < min)

{

min = a[n];

minPos = n;

}

}

printf("max=%d, pos=%d\n",max, maxPos);

printf("min=%d, pos=%d\n",min, minPos);

}

参考答案

#include <stdio.h>

main()

{ int a[10], n, max, min, maxPos, minPos;

for (n=0; n<10; n++)

{

scanf("%d", &a[n]);

}

max = min = a[0]; //应初始化为数组中的某一元素

maxPos = minPos = 0;

for (n=1; n<10; n++)

{

if (a[n] > max)

{

max = a[n];

maxPos = n;

}

else if (a[n] < min)

{

min = a[n];

minPos = n;

}

}

printf("max=%d, pos=%d\n",max, maxPos);

printf("min=%d, pos=%d\n",min, minPos);

}

Q486.(20分)求1+1/2+1/3+...+1/100的值。

\*\*输出格式要求："sum=%f\n"

参考答案

#include <stdio.h>

int main()

{

int k;

float sum;

k = 1;

sum = 0;

while (k <= 100)

{

sum = sum + 1.0 / k;

k++;

}

printf("sum=%f\n", sum);

return 0;

}

Q529.(10分)利用pi/2 = (2/1)\*(2/3)\*(4/3)\*(4/5)\*(6/5)\*(6/7)\*...前100项之积，编程计算pi的值。

\*\*输出格式要求："pi = %f\n"

参考答案

#include <stdio.h>

main()

{

float term, result = 1;

int n;

for (n = 2; n <= 100; n = n + 2)

{

term = (double)(n \* n) / (( n - 1) \* ( n + 1));

result = result \* term;

}

printf("pi = %f\n", 2 \* result);

}

Q225.(10分)假设银行一年整存零取的月息为1.875%，现在某人手头有一笔钱，他打算在今后5年中，每年年底取出1000元作为孩子来年的教育金，到第5年孩子毕业时刚好取完这笔钱，请采用逆推法编程计算第1年年初时他应存入银行多少钱。

\*\*输出格式要求："He must save %.2f at the first year.\n"

程序运行示例如下：

He must save 2833.29 at the first year.

参考答案

#include <stdio.h>

#define RATE 0.01875

#define MONTHS 12

#define CAPITAL 1000

#define YEARS 5

main()

{

int i;

double deposit = 0;

for (i = 0; i < YEARS; i++)

{

deposit = (deposit + CAPITAL) / (1 + RATE \* MONTHS);

}

printf("He must save %.2f at the first year.\n", deposit);

}

Q234.(10分)在海军节开幕式上，有A、B、C三艘军舰要同时开始鸣放礼炮各21响。已知A舰每隔5秒放1次，B舰每隔6秒放1次，C舰每隔7秒放1次。假设各炮手对时间的掌握非常准确，请编程计算观众总共可以听到几次礼炮声。

\*\*输出格式要求："n = %d\n"

参考答案

#include <stdio.h>

main()

{

int n = 0, t;

for (t = 0; t <= 20 \* 7; t++)

{

if (t % 5 == 0 && t <= 20 \* 5) /\* 控制A舰每隔5秒放1次 \*/

{

n++;

continue; /\* 继续下一次循环 \*/

}

if (t % 6 == 0 && t <= 20 \* 6) /\* 控制B舰每隔6秒放1次 \*/

{

n++;

continue; /\* 继续下一次循环 \*/

}

if (t % 7 == 0) /\* 控制C舰每隔7秒放1次 \*/

{

n++;

}

}

printf("n = %d\n", n);

}

Q1698.(25分)根据最大公约数的如下3条性质，采用递归法编写计算最大公约数的函数Gcd()，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大公约数。

性质1 如果a>b，则a和b与a-b和b的最大公约数相同，即Gcd(a, b) = Gcd(a-b, b)

性质2 如果b>a，则a和b与a和b-a的最大公约数相同，即Gcd(a, b) = Gcd(a, b-a)

性质3 如果a=b，则a和b的最大公约数与a值和b值相同，即Gcd(a, b) = a = b

要求如下：

（1）从键盘任意输入的两整数

主函数调用Gcd()函数，并输出两整数的最大公约数。

（2）Gcd函数原型为：

int Gcd(int a, int b);

如果输入的数不是正整数，则返回-1，否则，返回两个数的最大公约数。

（3）\*\*输入提示信息格式要求："Input a,b:\n"

输入两个整数时用,号分隔

\*\*输出提示信息要求：

如果输入的数不是正整数，则输出"Input error!\n"，否则按如下格式输出"Gcd=%d\n"

参考答案

#include <stdio.h>

int Gcd(int a, int b);

int main()

{

int a, b, c;

printf("Input a,b:");

scanf("%d,%d", &a, &b);

c = Gcd(a, b);

if (c != -1)

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c);

else

printf("Input number should be positive!\n");

return 0;

}

/\* 函数功能：递归方法计算a和b的最大公约数，输入负数时返回-1 \*/

int Gcd(int a, int b)

{

if (a <= 0 || b <= 0) return -1;

if (a == b)

return a;

else if (a > b)

return Gcd(a - b, b);

else

return Gcd(a, b - a);

}

Q238.(10分)利用求阶乘函数Fact()，编程计算并输出1! + 2! + …… + n!的值。

\*\*输入格式要求："%u" 提示信息："Input n(n>0):"

\*\*输出格式要求："sum = %lu\n

参考答案

#include <stdio.h>

unsigned long Fact(unsigned int n);

int main()

{

unsigned int i, n;

unsigned long sum = 0;

printf("Input n(n>0):");

scanf("%u", &n);

for (i = 1; i <= n; i++)

{

sum = sum + Fact(i);

}

printf("sum = %lu\n", sum);

return 0;

}

/\* 函数功能：用迭代法计算无符号整型变量n的阶乘 \*/

unsigned long Fact(unsigned int n)

{

unsigned int i;

unsigned long result = 1;

for (i = 2; i <= n; i++)

result \*= i;

return result;

}

Q1697.(10分)按如下函数原型，采用穷举法编写计算最大公约数的函数Gcd()，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大公约数。穷举法的基本思想是：先找到a和b中的较小者t，然后从t开始逐次减1尝试每种可能，即检验t到1之间的所有整数，第一个满足公约数条件的t，就是a和b的最大公约数。

/\* 函数功能：计算a和b的最大公约数，输入负数时返回-1 \*/

int Gcd(int a, int b)

要求如下：

（1）从键盘任意输入的两整数

主函数调用Gcd()函数，并输出两整数的最大公约数。

（2）Gcd函数原型为：

int Gcd(int a, int b);

如果输入的数不是正整数，则返回-1，否则，返回两个数的最大公约数。

（3）\*\*输入提示信息格式要求："Input a,b:\n"

输入两个整数时用,号分隔

\*\*输出提示信息要求：

如果输入的数不是正整数，则输出"Input error!\n"，否则按如下格式输出"Gcd=%d\n"

参考答案

#include <stdio.h>

int Gcd(int a, int b);

int main()

{

int a, b, c;

printf("Input a,b:");

scanf("%d,%d", &a, &b);

c = Gcd(a, b);

if (c != -1)

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c);

else

printf("Input number should be positive!\n");

return 0;

}

/\* 函数功能：计算a和b的最大公约数，输入负数时返回-1 \*/

int Gcd(int a, int b)

{

int i, t;

if (a <= 0 || b <= 0) return -1;

t = a < b ? a : b;

for (i = t; i > 0; i--)

{

if (a % i == 0 && b % i == 0) return i;

}

return 1;

}

Q239.(10分)五个水手在岛上发现一堆椰子，先由第1个水手把椰子分为等量的5堆，还剩下1个给了猴子，自己藏起1堆。然后，第2个水手把剩下的4堆混合后重新分为等量的5堆，还剩下1个给了猴子，自己藏起1堆。以后第3、4个水手依次按此方法处理。最后，第5个水手把剩下的椰子分为等量的5堆后，同样剩下1个给了猴子。请用迭代法编程计算并输出原来这堆椰子至少有多少个。

\*\*输出格式要求："y = %d\n"

参考答案

#include <stdio.h>

int Coconut(int n);

int main()

{

printf("y = %d\n", Coconut(5));

return 0;

}

int Coconut(int n)

{

int i = 1;

float x = 1, y;

y = n \* x + 1;

do

{

y = y \* n / (n - 1) + 1;

i++;

if (y != (int)y)

{

x = x + 1;

y = n \* x + 1;

i = 1;

}

}

while (i < n);

return (int)y;

}

Q226.(10分)已知今年的工业产值为100万元，产值增长率为从键盘输入，请编程计算工业产值过多少年可实现翻一番（即增加一倍）。

参考答案：用符号常量CURRENT表示今年的工业产值为100万元，用变量growRate表示产值增长率，用变量year表示产值翻番所需的年数，则计算年产值增长额的计算公式为：

output = output \* (1 + growRate)

利用迭代法循环计算，直到output >= 2\*CURRENT时为止。当output >= 2\*CURRENT时，表示已实现产值翻番。此时，循环被执行的次数year即为产值翻番所需的年数。

\*\*输入格式要求："%lf" 提示信息："Input grow rate:"

\*\*输出格式要求："When grow rate is %.0f%%, the output can be doubled after %d years.\n"

程序运行示例如下：

Input grow rate:0.08

When grow rate is 8%, the output can be doubled after 10 years.

参考答案

#include <stdio.h>

#define CURRENT 100

main()

{

int year; /\* 产值翻番所需年数 \*/

double growRate; /\* 工业产值的增长率 \*/

double output; /\* 工业总产值 \*/

printf("Input grow rate:");

scanf("%lf", &growRate);

output = CURRENT; /\* 当年产值为100万元 \*/

for (year = 0; output < 2 \* CURRENT; year++)

{

output = output \* (1 + growRate);

}

printf("When grow rate is %.0f%%, the output can be doubled after

%d years.\n", growRate \* 100, year);

}

Q231.(10分)马克思手稿中有这样一道趣味数学题：男人、女人和小孩总计30个人，在一家饭店里吃饭，共花了50先令，每个男人各花3先令，每个女人各花2先令，每个小孩各花1先令，请用穷举法编程计算男人、女人和小孩各有几人。

\*\*输出格式要求："Man\tWomen\tChildren\n" "%3d\t%5d\t%8d\n"

参考答案

#include <stdio.h>

main()

{

int x, y, z;

printf("Man\tWomen\tChildren\n");

for (x = 0; x <= 30; x++)

{

for (y = 0; y <= 30; y++)

{

for (z = 0; z <= 30; z++)

{

if (x + y + z == 30 && 3 \* x + 2 \* y + z == 50)

{

printf("%3d\t%5d\t%8d\n", x, y, z);

}

}

}

}

}

Q536.(10分)利用静态变量计算n的阶乘。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："%d! = %ld\n"

程序运行示例如下：

Input n:10

1! = 1

2! = 2

3! = 6

4! = 24

5! = 120

6! = 720

7! = 5040

8! = 40320

9! = 362880

10! = 3628800

参考答案

#include <stdio.h>

long Func(int n);

int main()

{

int i, n;

printf("Input n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

printf("%d! = %ld\n", i, Func(i));

}

return 0;

}

long Func(int n)

{

static long p = 1; /\*定义静态变量\*/

p = p \* n;

return p;

}

Q281.(10分)韩信点兵。韩信有一队兵，他想知道有多少人，便让士兵排队报数。

按从1至5报数，最末一个士兵报的数为1；

按从1至6报数，最末一个士兵报的数为5；

按从1至7报数，最末一个士兵报的数为4；

最后再按从1至11报数，最末一个士兵报的数为10。

你知道韩信至少有多少兵吗？

编写程序计算并输出结果。

\*\*输出格式要求为"x = %d\n"。

参考答案

#include <stdio.h>

main()

{

int x = 1; //1

int find = 0;

while (!find) //1

{

if (x % 5 == 1 && x % 6 == 5 && x % 7 == 4 && x % 11 == 10) //2

{

printf("x = %d\n", x);

find = 1; //1

}

x++; //1

}

}

Q609.(10分)编程计算s=a+aa+aaa+aaaa+aa...a的值，其中a是一个数字。例如2+22+222+2222+22222(此时共有5个数相加)，a和相加的项数由用户输入控制。

\*\*输入格式要求："%d,%d" 提示信息："please input a and n\n"

\*\*输出格式要求："a=%d,n=%d\n" "a+aa+...=%ld\n"

程序运行示例如下：

please input a and n

3,5

a=3,n=5

a+aa+...=37035

参考答案

int main()

{

int a, n, count = 1;

long int sn = 0, tn = 0;

printf("please input a and n\n");

scanf("%d,%d", &a, &n);

printf("a=%d,n=%d\n", a, n);

while (count <= n)

{

tn = tn + a;

sn = sn + tn;

a = a \* 10;

++count;

}

printf("a+aa+...=%ld\n", sn);

}

Q279.(20分)百鸡问题：公鸡每只5元，母鸡每只3元，小鸡3只1元。

用100元买100只鸡，问公鸡、母鸡和小鸡各能买多少只？

\*\*要求输入提示信息为：无输入提示信息和输入数据

\*\*输出格式要求为："x=%d,y=%d,z=%d\n"，

其中x,y,z分别表示公鸡、母鸡和小鸡只数。

参考答案

#include <stdio.h>

main()

{

int x, y, z;

for (x = 0; x <= 20; x++) //2

{

for (y = 0; y <= 33; y++) //2

{

z =100-x-y; //2

if (5\*x + 3\*y + z/3.0 == 100) //2

{

printf("x=%d,y=%d,z=%d\n", x, y, z); //2

}

}

}

}

Q185.(25分)鸡兔同笼，共有98个头，386只脚，请用穷举法编程计算鸡、兔各多少只。

\*\*输入提示信息格式要求：无输入数据

\*\*输出格式要求："x=%d,y=%d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

main()

{

int x, y;

for (x = 1; x <= 98; x++)//4

{

for (y = 1; y < 98; y++)//4

{

if (x + y == 98 && 2 \* x + 4 \* y == 386) //4

{

printf("x=%d,y=%d\n", x, y); //2

}

}

}

}

Q401.(10分)用下列公式求pi的近似值，直到最后一项的绝对值小于10e-4为止：

π4=1−13+15−17+…

\*\*输出格式要求："pi=%10.6f\n"

程序运行示例如下：

pi= 3.141393

参考答案

#include "math.h"

#include <stdio.h>

main()

{ int s;

double n,t,pi;

t=1;pi=0;n=1;s=1;

while((fabs(t))>=1e-4)

{pi+=t;

n+=2;

s=-s;

t=s/n;

}

pi=pi\*4;

printf("pi=%10.6f\n",pi);

}

Q223.(10分)编程计算并输出1到n之间的所有数的平方与立方。其中，n值由用户从键盘输入。

\*\*输入格式要求："%d" 提示信息："Please enter n:"

\*\*输出格式要求："%d\*%d = %d\n" "%d\*%d\*%d = %d\n"

程序运行示例如下：

Please enter n:5

1\*1 = 1

2\*2 = 4

3\*3 = 9

4\*4 = 16

5\*5 = 25

1\*1\*1 = 1

2\*2\*2 = 8

3\*3\*3 = 27

4\*4\*4 = 64

5\*5\*5 = 125

参考答案

#include <stdio.h>

main()

{

int i, n;

printf("Please enter n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

printf("%d\*%d = %d\n", i, i, i \* i); /\* 输出所有数的平方 \*/

}

for (i = 1; i <= n; i++)

{

printf("%d\*%d\*%d = %d\n", i, i, i, i \* i \* i); /\* 输出所有数的立方 \*/

}

}

Q1555.(20分)用循环编程计算1+3+5+7+…+99+101的值。

\*\*输入格式要求：无任何信息提示；

\*\*输出格式要求："sum=%d\n"

参考答案

#include <stdio.h>

main()

{

int i, sum = 0;

for (i = 1; i <= 101; i = i+2)

{

sum = sum + i;

}

printf("sum=%d\n", sum);

}

Q232.(10分)有一天，一位百万富翁遇到一个陌生人，陌生人找他谈一个换钱的计划，陌生人对百万富翁说：“我每天给你10万元，而你第一天只需给我1分钱，第二天我仍给你10万元，你给我2分钱，第三天我仍给你10万元，你给我4分钱……。你每天给我的钱是前一天的两倍，直到满一个月（30天）为止”，百万富翁很高兴，欣然接受了这个契约。请编程计算在这一个月中陌生人总计给百万富翁多少钱，百万富翁总计给陌生人多少钱。

\*\*输出格式要求："百万富翁给陌生人: %8ld元\n陌生人给百万富翁: %8ld元\n"

参考答案

#include <stdio.h>

main()

{

int j;

long sumToStranger = 0; /\* 百万富翁给陌生人的钱，以'元'为单位 \*/

long sumToRichman = 0; /\* 陌生人给百万富翁的钱，以'分'为单位 \*/

long term = 1; /\* 陌生人第一天给百万富翁1分钱 \*/

for (j = 1; j <= 30; j++)

{

sumToRichman = sumToRichman + 100000; /\* 富翁每天给陌生人10万元 \*/

sumToStranger = sumToStranger + term;

term = term \* 2; /\* 陌生人每天给富翁的钱是前一天的两倍 \*/

}

sumToStranger = sumToStranger / 100;/\* 百万富翁给陌生人的钱换算成'元' \*/

printf("百万富翁给陌生人: %8ld元\n陌生人给百万富翁: %8ld元\n",

sumToStranger, sumToRichman);

}

Q230.(10分)输入一些整数，编程计算并输出其中所有正数的和，输入负数时不累加，继续输入下一个数。输入零时，表示输入数据结束。要求最后统计出累加的项数。

\*\*输入格式要求："%d" 提示信息："Input a number:"

\*\*输出格式要求："sum = %d, count = %d\n"

程序运行示例如下：

Input a number:1

Input a number:2

Input a number:3

Input a number:-4

Input a number:4

Input a number:5

Input a number:0

sum = 15, count = 5

参考答案

#include <stdio.h>

main()

{

int i = 0, n, sum = 0;

printf("Input a number:");

scanf("%d", &n);

while (n != 0)

{

if (n > 0)

{

sum = sum + n;

i++;

}

printf("Input a number:");

scanf("%d", &n);

}

printf("sum = %d, count = %d\n", sum, i);

}

Q412.(10分)三色球问题。若一个口袋中放有12个球，其中有3个红，3个白和6个黑的，从中任取8个球，问共有多少种不同的颜色搭配？

\*\*输出格式要求：" RED BALL WHITE BALL BLACK BALL\n" "----------------------------------------\n" "%2d: %d %d %d\n"

程序运行示例如下：

RED BALL WHITE BALL BLACK BALL

----------------------------------------

1: 0 2 6

2: 0 3 5

3: 1 1 6

4: 1 2 5

5: 1 3 4

6: 2 0 6

7: 2 1 5

8: 2 2 4

9: 2 3 3

10: 3 0 5

11: 3 1 4

12: 3 2 3

13: 3 3 2

参考答案

#include <stdio.h>

main()

{ int i,j,count=0;

printf(" RED BALL WHITE BALL BLACK BALL\n");

printf("----------------------------------------\n");

for(i=0;i<=3;i++)

for(j=0;j<=3;j++)

if((8-i-j)<=6)

printf("%2d: %d %d %d\n",++count,i,j,8-i-j);

}

Q405.(10分)穷举法，百钱百鸡问题。鸡翁一，值钱五；鸡母一，值钱三；鸡雏三，值钱一；百钱买百鸡，翁、母、雏各几何？

\*\*输出格式要求："%2d:cock=%2d hen=%2d chicken=%2d\n"

程序运行示例如下：

1:cock= 0 hen=25 chicken=75

2:cock= 4 hen=18 chicken=78

3:cock= 8 hen=11 chicken=81

4:cock=12 hen= 4 chicken=84

参考答案

#include <stdio.h>

main()

{int x,y,z,j=0;

for(x=0;x<=20;x++)

for(y=0;y<=33;y++)

for(z=0;z<=100;z++)

if(z%3==0&&x+y+z==100&&5\*x+3\*y+z/3==100)

printf("%2d:cock=%2d hen=%2d chicken=%2d\n",++j,x,y,z);

}

Q411.(10分)爱因斯坦的数学题。爱因斯坦出了一道这样的数学题。有一条长阶梯，若每步跨2阶，则最后剩1阶，若每步跨3阶，则最后剩2阶，若每步跨5阶，则最后剩4阶，若每步跨6阶，则最后剩5阶，若每步跨7阶，最后才正好一阶不剩。请问，这条阶梯共多少阶？

\*\*输出格式要求："x = %d\n"

参考答案

#include <stdio.h>

main()

{int i=1;

while(!((i%2==1)&&(i%3==2)&&(i%5==4)&&(i%6==5)&&(i%7==0)))

++i;

printf("x = %d\n",i);

}

Q1694.(10分)

利用sinx= ，计算sinx（x为弧度值）的值，直到最后一项的绝对值小于10e-5时为止，输出sinx的值并统计累加的项数。

\*\*输入格式要求："%f" 提示信息："Input x:"

\*\*输出格式要求："sin(x) = %f, count = %d\n"

程序运行示例如下：

Input x:10↙

sin(x) = -0.544022, count = 18

参考答案

#include <stdio.h>

#include <math.h>

main()

{

int n = 1, count = 1;

float x;

double sum, term;

printf("Input x:");

scanf("%f", &x);

sum = x;

term = x;

do

{

term = -term \* x \* x / ((n + 1) \* (n + 2 ));

sum = sum + term;

n = n + 2;

count++;

}

while (fabs(term) >= 1e-5);

printf("sin(x) = %f, count = %d\n", sum, count);

}

Q296.(15分)鸡兔同笼，共有98个头，386只脚，请用穷举法编程计算鸡、兔各多少只。

\*\*输入提示信息格式要求：无输入数据

\*\*输出格式要求："x=%d,y=%d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

main()

{

int x, y;

for (x = 1; x <= 98; x++)//4

{

for (y = 1; y < 98; y++)//4

{

if (x + y == 98 && 2 \* x + 4 \* y == 386) //4

{

printf("x=%d,y=%d\n", x, y); //2

}

}

}

}

Q521.(20分)用do while语句编程，输入一组整型数据，然后显示每次将输入数据进行累加运算后的结果。当输入0时，停止输入数据，结束程序的运行。

\*\*输入格式要求："%d" 提示信息："Input num:"

\*\*输出格式要求："sum = %d\n"

程序运行示例如下：

Input num:1

sum = 1

Input num:2

sum = 3

Input num:3

sum = 6

Input num:4

sum = 10

Input num:0

sum = 10

参考答案

#include <stdio.h>

main()

{

int num;

int sum = 0; /\* 累加求和变量初始化为0 \*/

do

{

printf("Input num:");

scanf ("%d", &num); /\* 输入数据 \*/

sum = sum + num; /\* 做累加运算 \*/

printf("sum = %d\n", sum); /\* 输出累加运算结果 \*/

}

while (num != 0); /\* num为0时停止累加运算 \*/

}

Q1696.(10分)输入一些整数，编程计算并输出其中所有正数的和，输入负数时不累加，继续输入下一个数。输入零时，表示输入数据结束。要求最后统计出累加的项数。

输入提示信息："Input a number:"

输入格式："%d"

输出提示信息和格式："sum = %d, count = %d\n"

参考答案

#include <stdio.h>

main()

{

int i = 0, n, sum = 0;

printf("Input a number:");

scanf("%d", &n);

while (n != 0)

{

if (n > 0)

{

sum = sum + n;

i++;

}

printf("Input a number:");

scanf("%d", &n);

}

printf("sum = %d, count = %d\n", sum, i);

}

Q520.(10分)编程从键盘输入n，然后计算并输出1+2+3+…+n的值。用do-while语句编程实现。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："sum = %d\n"

程序运行示例如下：

Input n:100

sum = 5050

参考答案

#include <stdio.h>

main()

{

int i, n, sum;

printf("Input n:");

scanf("%d", &n);

sum = 0; /\* 累加求和变量初始化为0 \*/

i = 1;

do

{

sum = sum + i; /\* 做累加运算 \*/

i++; /\* 累加计数器加1 \*/

}

while (i <= n);

printf("sum = %d\n", sum);

}

Q524.(10分)从键盘输入n，然后计算并输出1～n之间的所有数的阶乘值。

\*\*输入格式要求："%d" 提示信息："Please enter n:"

\*\*输出格式要求："%d! = %ld\n"

程序运行示例如下：

Please enter n:10

1! = 1

2! = 2

3! = 6

4! = 24

5! = 120

6! = 720

7! = 5040

8! = 40320

9! = 362880

10! = 3628800

参考答案

#include <stdio.h>

main()

{

int i, n;

long p = 1;

printf("Please enter n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

p = p \* i;

printf("%d! = %ld\n", i, p); /\* 输出1～n之间的所有数的阶乘值 \*/

}

}

Q527.(10分)读入5个正整数并且显示它们。当程序读入的数据为负数时，程序立即终止。

\*\*输入格式要求："%d" 提示信息："Please enter n:"

\*\*输出格式要求："n = %d\n" "Program is over!\n"

程序运行示例如下：

Please enter n:10

n = 10

Please enter n:20

n = 20

Please enter n:-30

Program is over!

参考答案

#include <stdio.h>

main()

{

int i, n;

for (i = 1; i <= 5; i++)

{

printf("Please enter n:");

scanf("%d", &n);

if (n < 0) break;

printf("n = %d\n", n);

}

printf("Program is over!\n");

}

Q194.(30分)用100元人民币兑换10元,5元和1元的纸币(每一种都要有)共50张,请用穷举法编程计算共有几种兑换方案,每种方案各兑换多少张纸币.

\*\*\*\*输入提示信息要求为：无

\*\*\*\*输出格式要求为：

"x = %d, y = %d, z = %d\n"

"count = %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

main()

{

int x, y, z, count = 0;//1

for (x=1; x<50; x++)//3

{

for (y=1; y<50; y++)//3

{

for (z=1; z<50; z++)//3

{

if (x+y+z == 50 && 10\*x+5\*y+z == 100)//2

{

count++;//1

printf("x = %d, y = %d, z = %d\n", x, y, z);//1

}

}

}

}

printf("count = %d\n", count);//1

}

Q228.(10分)利用e=1+11!+12!+13!+⋯+1n!，编程计算e的近似值，直到最后一项的绝对值小于10−5时为止，输出e的值并统计累加的项数。

\*\*输出格式要求："e = %f, count = %d\n"

参考答案

#include <math.h>

#include <stdio.h>

main()

{

int n = 1, count = 1;

double e = 1.0, term = 1.0;

long fac = 1;

for (n = 1; fabs(term) >= 1e-5; n++)

{

fac = fac \* n;

term = 1.0 / fac; /\* 计算累加项 \*/

e = e + term; /\* 累加 \*/

count++; /\* 记录累加的项数 \*/

}

printf("e = %f, count = %d\n", e, count);

}

Q522.(10分)用while语句编程，输入一组整型数据，然后显示每次将输入数据进行累加运算后的结果。当输入0时，停止输入数据，结束程序的运行。

\*\*输入格式要求："%d" 提示信息："Input num:"

\*\*输出格式要求："sum = %d\n"

程序运行示例如下：

Input num:1

sum = 1

Input num:2

sum = 3

Input num:3

sum = 6

Input num:4

sum = 10

Input num:0

sum = 10

参考答案

#include <stdio.h>

main()

{

int num = 1; /\* 变量num必须初始化为1 \*/

int sum = 0; /\* 累加求和变量初始化为0 \*/

while (num != 0) /\* num为0时停止累加运算 \*/

{

printf("Input num:");

scanf ("%d", &num); /\* 输入数据 \*/

sum = sum + num; /\* 做累加运算 \*/

printf("sum = %d\n", sum); /\* 输出累加运算结果 \*/

}

}

Q227.(35分)利用π4=1−13+15−17+…，编程计算π的近似值，直到最后一项的绝对值小于10−5时为止，输出π的值并统计累加的项数count。

\*\*输出格式要求： "pi = %f\ncount = %d\n"

程序运行示例如下：

pi = 3.141793

count = 5001

参考答案

#include <math.h>

#include <stdio.h>

main()

{

double pi, sum = 0, term, sign = 1.0;

int count = 0, n = 1;

do

{

term = sign / n; /\* 计算累加项 \*/

sum = sum + term; /\* 累加 \*/

count++; /\* 记录累加的项数 \*/

sign = -sign; /\* 累加项分子的变化规律 \*/

n = n + 2; /\* 累加项分母的变化规律 \*/

}

while (fabs(term) >= 1e-4); /\* 判断累加项是否满足循环终止条件 \*/

pi = sum \* 4;

printf("pi = %f\ncount = %d\n", pi, count);

}

Q1692.(10分)从键盘任意输入一个年号，判断它是否是闰年。若是闰年输出“Yes”，否则输出“No”。已知符合下列条件之一者是闰年：（1）能被4整除，但不能被100整除；（2）能被400整除。

输入提示信息："Input a year:"

输入格式："%d"

输出提示信息和格式：

"%d is a leap year!\n"

"%d is not a leap year!\n"

参考答案

#include <stdio.h>

main()

{

int year, flag;

printf("Input a year:");

scanf("%d", &year);

if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0)

flag = 1; /\* 如果year是闰年，则标志变量flag置1 \*/

else

flag = 0; /\* 否则，标志变量flag置0 \*/

if (flag)

printf("%d is a leap year!\n", year); /\* 打印“是闰年”\*/

else

printf("%d is not a leap year!\n", year); /\* 打印“不是闰年”\*/

}

Q514.(10分)使用双分支条件语句编程，计算并输出两个整数的最大值。

\*\*输入格式要求："%d,%d" 提示信息："Input a, b:"

\*\*输出格式要求："max = %d\n"

程序运行示例如下：

Input a, b:3,5

max = 5

参考答案

#include <stdio.h>

main()

{

int a, b, max;

printf("Input a, b:");

scanf("%d,%d", &a, &b);

if (a > b) max = a;

else max = b; /\* 相当于a<=b的情况 \*/

printf("max = %d\n", max);

}

Q216.(10分)从键盘任意输入三角形的三边长为a，b，c，编程判断a，b，c的值能否构成一个三角形，若能构成三角形，则计算并输出三角形的面积，否则提示不能构成三角形。已知构成三角形的条件是：任意两边之和大于第三边。

\*\*输入格式要求："%f,%f,%f" 提示信息："Input a,b,c:"

\*\*输出格式要求："area = %f\n" "It is not a triangle\n"

程序运行示例1如下：

Input a,b,c:3,4,5

area = 6.000000

程序运行示例2如下：

Input a,b,c:3,4,8

It is not a triangle

参考答案

#include <stdio.h>

#include <math.h>

main()

{

float a, b, c, s, area;

printf("Input a,b,c:");

scanf("%f,%f,%f", &a, &b, &c);

if (a + b > c && b + c > a && a + c > b)

{

s = (float)(a + b + c) / 2;

area = sqrt(s \* (s - a) \* (s - b) \* (s - c));

printf("area = %f\n", area);

}

else

{

printf("It is not a triangle\n");

}

}

Q517.(10分)编程实现简单的计算器功能，要求用户按如下格式从键盘输入算式：

操作数1 运算符op 操作数2

计算并输出表达式的值，其中算术运算符包括：加（+）、减（-）、乘（\*）、除（/）。

要求使其能进行浮点数的算术运算，同时允许使用字符\*、x与X作为乘号，并且允许输入的算术表达式中的操作数和运算符之间可以加入任意多个空格符。

\*\*输入格式要求："%f %c%f" 提示信息："Please enter the expression:\n"

\*\*输出格式要求："%f + %f = %f \n" "%f - %f = %f \n" "%f \* %f = %f \n" "%f / %f = %f \n" "Division by zero!\n" "Invalid operator! \n"

程序运行示例1如下：

Please enter the expression: 2.0 + 4.0

2.000000 + 4.000000 = 6.000000

程序运行示例2如下：

Please enter the expression: 2.0 - 4.0

2.000000 - 4.000000 = -2.000000

程序运行示例3如下：

Please enter the expression: 2.0 \* 4.0

2.000000 \* 4.000000 = 8.000000

程序运行示例4如下：

Please enter the expression: 2.0 x 4.0

2.000000 x 4.000000 = 8.000000

程序运行示例5如下：

Please enter the expression: 2.0 X 4.0

2.000000 X 4.000000 = 6.000000

程序运行示例6如下：

Please enter the expression: 2.0 / 4.0

2.000000 / 4.000000 = 0.5000000

程序运行示例7如下：

Please enter the expression: 2.0 / 0

Division by zero!

程序运行示例8如下：

Please enter the expression: 2.0 \ 4.0

Invalid operator!

参考答案

#include <math.h>

#include <stdio.h>

main()

{

float data1, data2;

char op;

printf("Please enter the expression:\n");

scanf("%f %c%f", &data1, &op, &data2); /\* %c前有一个空格 \*/

switch (op)

{

case '+':

printf("%f + %f = %f \n", data1, data2, data1 + data2);

break;

case '-':

printf("%f - %f = %f \n", data1, data2, data1 - data2);

break;

case '\*':

case 'x':

case 'X':

printf("%f \* %f = %f \n", data1, data2, data1 \* data2);

break;

case '/':

if (fabs(data2) <= 1e-7) /\* 实数与0比较 \*/

{

printf("Division by zero!\n");

}

else

{

printf("%f / %f = %f \n", data1, data2, data1 / data2);

}

break;

default:

printf("Invalid operator! \n");

}

}

Q219.(10分)从键盘任意输入一个字符，编程判断该字符是数字字符、大写字母、小写字母、空格还是其他字符。

\*\*输入格式要求：提示信息："Press a key and then press Enter:"

\*\*输出格式要求："It is an English character!\n" "It is a digit character!\n" "It is a space character!\n" "It is other character!\n"

程序运行示例1如下：

Press a key and then press Enter:A

It is an English character!

程序运行示例2如下：

Press a key and then press Enter:2

It is a digit character!

程序运行示例3如下：

Press a key and then press Enter:

It is a space character!

程序运行示例4如下：

Press a key and then press Enter:#

It is other character!

参考答案

#include <stdio.h>

main()

{

char ch;

printf("Press a key and then press Enter:");

ch = getchar();

if (ch >= 'a' && ch <= 'z' || ch >= 'A' && ch <= 'Z')

{

printf("It is an English character!\n");

}

else if (ch <= '9' && ch >= '0')

{

printf("It is a digit character!\n");

}

else if (ch == ' ')

{

printf("It is a space character!\n");

}

else

{

printf("It is other character!\n");

}

}

Q221.(25分)编程从键盘输入某年某月（包括闰年），用switch语句编程输出该年的该月拥有的天数。要求考虑闰年以及输入月份不在合法范围内的情况。已知闰年的2月有29天，平年的2月有28天。

\*\*输入格式要求："%d, %d" 提示信息："Input year,month:"

\*\*输出格式要求："31 days\n" "29 days\n" "28 days\n"

程序运行示例如下：

Input year,month:2004,2

29 days

参考答案

#include <stdio.h>

main()

{

int year, month;

printf("Input year,month:");

scanf("%d, %d", &year, &month);

switch (month)

{

case 1:

case 3:

case 5:

case 7:

case 8:

case 10:

case 12:

printf("31 days\n");

break;

case 2:

if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))

{

printf("29 days\n"); /\* 闰年的2月有29天 \*/

}

else

{

printf("28 days\n"); /\* 平年的2月有28天 \*/

}

break;

case 4:

case 6:

case 9:

case 11:

printf("30 days\n");

break;

default:

printf("Input error!\n");

}

}

Q218.(20分)从键盘输入一个英文字母，如果它是大写英文字母，则将其转换为小写英文字母，如果它是小写英文字母，则将其转换为大写英文字母，然后将它及其ASCII码值显示到屏幕上，如果不是英文字母，则不转换直接输出到屏幕上。

\*\*输入格式要求：提示信息："Press a key and then press Enter:"

\*\*输出格式要求："%c, %d\n"

程序运行示例1如下：

Press a key and then press Enter:A

a, 97

程序运行示例2如下：

Press a key and then press Enter:a

A, 65

参考答案

#include <stdio.h>

main()

{

char ch;

printf("Press a key and then press Enter:");

ch = getchar();

if (ch >= 'A' && ch <= 'Z')

{

ch = ch + 32;

}

else if (ch >= 'a' && ch <= 'z')

{

ch = ch - 32;

}

printf("%c, %d\n", ch, ch);

}

Q213.(10分)从键盘任意输入a，b，c的值，编程计算并输出一元二次方程ax2+bx+c=0的根。根据一元二次方程的求根公式，令

p=−b2a,q=∣∣b2−4ac∣∣√2a

当b2−4ac=0时，输出两个相等的实根x1=x2=p；当b2−4ac>0时，输出两个不相等的实根：x1=p+q，x2=p−q；当b2−4ac<0时，输出一对共轭复根：x1=p+qi，x2=p−qi。当a=0时，输出"It is not a quadratic equation!\n"。

\*\*输入格式要求："%f,%f,%f" 提示信息："Please enter the coefficients a,b,c:"

\*\*输出格式要求：

相等实根: "x1 = x2 = %.2f\n"

不相等的实根: "x1 = %.2f, x2 = %.2f\n"

一对共轭复根: "x1 = %.2f+%.2fi, x2 = %.2f-%.2fi\n"

程序运行示例如下：

Please enter a,b,c: 0,10,2

It is not a quadratic equation!

参考答案

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#define EPS 1e-6

main()

{

float a, b, c, disc, p, q;

printf("Please enter the coefficients a,b,c:");

scanf("%f,%f,%f", &a, &b, &c);

if (fabs(a) <= EPS) /\* a=0时，输出“不是二次方程” \*/

{

printf("It is not a quadratic equation!\n");

exit(0);

}

disc = b \* b - 4 \* a \* c; /\* 计算判别式 \*/

p = - b / (2 \* a);

q = sqrt(fabs(disc)) / (2 \* a);

if (fabs(disc) <= EPS) /\* 判别式等于0时，输出两相等实根 \*/

{

printf("x1 = x2 = %.2f\n", p);

}

else

{

if (disc > EPS) /\* 判别式大于0时，输出两不等实根 \*/

{

printf("x1 = %.2f, x2 = %.2f\n", p + q, p - q);

}

else /\* 判别式小于0时，输出两共轭复根 \*/

{

printf("x1 = %.2f+%.2fi, ", p, q);

printf("x2 = %.2f-%.2fi\n", p, q);

}

}

}

Q214.(10分)从键盘任意输入一个实数，不使用计算绝对值函数编程计算并输出该实数的绝对值。

\*\*输入格式要求："%f" 提示信息："Input a float number:"

\*\*输出格式要求："Absolute value of x is %f\n"

程序运行示例如下：

Input a float number:-2.3

Absolute value of x is 2.300000

参考答案

#include <stdio.h>

main()

{

float x;

printf("Input a float number:");

scanf("%f", &x);

if (x < 0) x = -x;

printf("Absolute value of x is %f\n", x);

}

Q217.(10分)从键盘任意输入a，b，c的值，编程计算并输出一元二次方程ax2+bx+c=0的根，当a=0时，输出“该方程不是一元二次方程”，当a≠0时，分b2-4ac>0、b2-4ac=0、b2-4ac<0三种情况计算并输出方程的根。

\*\*输入格式要求："%f,%f,%f" 提示信息："Please enter the coefficients a,b,c:"

\*\*输出格式要求："It is not a quadratic equation!\n" "x1 = x2 = %.2f\n" "x1 = %.2f, x2 = %.2f\n"

"x1 = %.2f+%.2fi, " "x2 = %.2f-%.2fi\n"

程序运行示例1如下：

Please enter the coefficients a,b,c:0,10,2

It is not a quadratic equation!

程序运行示例2如下：

Please enter the coefficients a,b,c:1,2,1

x1 = x2 = -1.00

程序运行示例3如下：

Please enter the coefficients a,b,c:2,3,2

x1 = -0.75+0.66i, x2 = -0.75-0.66i

参考答案

#include <stdlib.h>

#include <math.h>

#include <stdio.h>

#define EPS 1e-6

main()

{

float a, b, c, disc, p, q;

printf("Please enter the coefficients a,b,c:");

scanf("%f,%f,%f", &a, &b, &c);

if (fabs(a) <= EPS) /\* 处理不是二次方程的情况 \*/

{

printf("It is not a quadratic equation!\n");

exit(0);

}

disc = b \* b - 4 \* a \* c; /\* 计算判别式 \*/

p = - b / (2 \* a);

q = sqrt( fabs( disc ) ) / (2 \* a);

if (fabs(disc) <= EPS) /\* 处理有两相等实根的情况 \*/

{

printf("x1 = x2 = %.2f\n", p);

}

else

{

if (disc > EPS) /\* 处理有两不等实根的情况 \*/

{

printf("x1 = %.2f, x2 = %.2f\n", p + q, p - q);

}

else /\* 处理有两复数根的情况 \*/

{

printf("x1 = %.2f+%.2fi, ", p, q);

printf("x2 = %.2f-%.2fi\n", p, q);

}

}

}

Q516.(10分)编程实现简单的计算器功能，要求用户按如下格式从键盘输入算式：

操作数1 运算符op 操作数2

计算并输出表达式的值，其中算术运算符包括：加（+）、减（-）、乘（\*）、除（/）。

\*\*输入格式要求："%d%c%d" 提示信息："Please enter an expression:"

\*\*输出格式要求："%d + %d = %d \n" "%d - %d = %d \n" "%d \* %d = %d \n" "%d / %d = %d \n" "Division by zero!\n" "Invalid operator! \n"

程序运行示例1如下：

Please enter an expression:22+12

22 + 12 = 34

程序运行示例2如下：

Please enter an expression:22-12

22 - 12 = 10

程序运行示例3如下：

Please enter an expression:22\*12

22 \* 12 = 264

程序运行示例4如下：

Please enter an expression:22/12

22 / 12 = 1

程序运行示例5如下：

Please enter an expression:22/0

Division by zero!

程序运行示例6如下：

Please enter an expression:22\12

Invalid operator!

参考答案

#include <stdio.h>

main()

{

int data1, data2;

char op;

printf("Please enter an expression:");

scanf("%d%c%d", &data1, &op, &data2); /\* 输入运算表达式 \*/

switch (op) /\* 根据输入的运算符确定执行的运算 \*/

{

case '+': /\* 加法运算 \*/

printf("%d + %d = %d \n", data1, data2, data1 + data2);

break;

case '-': /\* 减法运算 \*/

printf("%d - %d = %d \n", data1, data2, data1 - data2);

break;

case '\*': /\* 乘法运算 \*/

printf("%d \* %d = %d \n", data1, data2, data1 \* data2);

break;

case '/': /\* 除法运算 \*/

if (0 == data2) /\* 为避免除0错误，检验除数是否为0 \*/

{

printf("Division by zero!\n");

}

else

{

printf("%d / %d = %d \n", data1, data2, data1 / data2);

}

break;

default: /\* 处理非法运算符 \*/

printf("Invalid operator! \n");

}

}

Q220.(20分)

用if-else语句编程根据输入的百分制成绩score，转换成相应的五分制成绩grade后输出。已知转换标准为：

0-59 E

60-69 D

70-79 C

80-89 B

90-100 A

\*\*输入格式要求："%d" 提示信息："Please enter score:"

\*\*输出格式要求："Input error!\n" "%d——A\n"

程序运行示例1如下：

Please enter score:15

15——E

程序运行示例2如下：

Please enter score:85

15——B

参考答案

#include <stdio.h>

main()

{

int score;

printf("Please enter score:");

scanf("%d", &score);

if (score < 0 || score > 100)

{

printf("Input error!\n");

}

else if (score >= 90)

{

printf("%d——A\n", score);

}

else if (score >= 80)

{

printf("%d——B\n", score);

}

else if (score >= 70)

{

printf("%d——C\n", score);

}

else if (score >= 60)

{

printf("%d——D\n", score);

}

else

{

printf("%d——E\n", score);

}

}

Q513.(10分)使用单分支的条件语句编程，计算并输出两个整数的最大值。

\*\*输入格式要求："%d,%d" 提示信息："Input a, b:"

\*\*输出格式要求："max = %d\n"

程序运行示例如下：

Input a, b:3,5

max = 5

参考答案

#include <stdio.h>

main()

{

int a, b, max;

printf("Input a, b:");

scanf("%d,%d", &a, &b);

if (a > b) max = a;

if (a <= b) max = b;

printf("max = %d\n", max);

}

Q348.(10分)已知银行整存整取存款不同期限的月息利率分别为：

年利息率=⎧⎩⎨⎪⎪⎪⎪⎪⎪⎪⎪⎪⎪2.25%2.43%2.70%2.88%3.00%期限1年期限2年期限3年期限5年期限8年

要求输入存钱的期限和本金，求到期时能从银行得到的利息与本金的合计。

\*\*输入格式要求："%d,%lf" 提示信息："Please enter year,capital:"

\*\*输出格式要求："rate = %f, deposit = %f\n"，输入错误时显示"Error rate!\n"

程序运行示例1如下：

Please enter year,capital:2,10000↙

rate = 0.024300, deposit = 10491.904900

程序运行示例2如下：

Please enter year,capital:4,10000↙

Error rate!

参考答案

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

main()

{

int year;

double rate, capital, deposit;

printf("Please enter year,capital:");

scanf("%d,%lf", &year, &capital);

switch (year)

{

case 1:

rate = 0.0225;

break;

case 2:

rate = 0.0243;

break;

case 3:

rate = 0.0270;

break;

case 5:

rate = 0.0288;

break;

case 8:

rate = 0.0300;

break;

default:

printf("Error rate!\n");

exit(0);

}

deposit = capital \* pow(1 + rate, year);

printf("rate = %f, deposit = %f\n", rate, deposit);

return 0;

}

Q515.(10分)使用条件运算符编程，计算并输出两个整数的最大值。

\*\*输入格式要求："%d,%d" 提示信息："Input a, b:"

\*\*输出格式要求："max = %d\n"

程序运行示例如下：

Input a, b:3,5

max = 5

参考答案

#include <stdio.h>

main()

{

int a, b, max;

printf("Input a, b:");

scanf("%d,%d", &a, &b);

max = a > b ? a : b; /\* 用条件表达式计算两整数的最大值\*/

printf("max = %d\n", max);

}

Q215.(10分)从键盘任意输入一个整数，编程判断它的奇偶性。

\*\*输入格式要求："%d" 提示信息："Input an integer number:"

\*\*输出格式要求："a is an even number\n" "a is an odd number\n"

程序运行示例1如下：

Input an integer number:2

a is an even number

程序运行示例2如下：

Input an integer number:5

a is an odd number

参考答案

#include <stdio.h>

main()

{

int a;

printf("Input an integer number:");

scanf("%d", &a);

if (0 == a % 2)

{

printf("a is an even number\n");

}

else

{

printf("a is an odd number\n");

}

}

Q610.(10分)有一分数序列：2/1，3/2，5/3，8/5，13/8，21/13...编程求出这个数列的前20项之和。

\*\*输出格式要求："sum is %9.6f\n"

参考答案

int main()

{

int n, t, number = 20;

float a = 2, b = 1, s = 0;

for (n = 1; n <= number; n++)

{

s = s + a / b;

t = a;

a = a + b;

b = t;

}

printf("sum is %9.6f\n", s);

return 0;

}

Q1348.(10分)输入：工资数，小时数（整数，空格分隔），

输出：工资/小时数（精确到小数点后2位），并根据四舍五入取整，然后将取整的数平方后计算一共有几位，后三位分别是什么。

程序的运行示例如下：

2345 2 （输入）

1172.50 （本行及以下为输出）

1173

7

9 2 9

参考答案

#include <stdio.h>

#include <math.h>

int main()

{

int i, salary, hours, a, b, c1, c2, c3;

float m;

scanf("%d %d", &salary, &hours);

m = (float)salary / hours;

a = (int)(m + 0.5);

b = pow(a, 2);

c3 = b % 10;

c2 = (b % 100 - c3) / 10;

c1 = (b % 1000 - c3 - c2 \* 10) / 100;

for (i = 0; b >= 1; i++)

{

b = b / 10;

}

printf("%.2f\n", m);

printf("%d\n", a);

printf("%d\n", i);

printf("%d %d %d\n", c1, c2, c3);

return 0;

}

Q272.(10分)华氏和摄氏温度的转换公式为:

C = 5/9 x (F-32)

其中，C表示摄氏温度，F表示华氏温度。

要求：华氏0℉～300℉，每隔20℉输出一个华氏温度对应的摄氏温度值。

\*\*输入提示信息要求:无

\*\*输入数据格式要求:无

\*\*输出数据格式要求: "%4.0f%10.1f\n" (第一个浮点数为华氏温度，第二个浮点数为摄氏温度)

参考答案

#include <stdio.h>

int main()

{

int upper, step;

float f = 0, c;

upper = 300; //1

step = 20; //1

while (f <= upper) //2

{

c = 5.0 / 9 \* (f - 32) ; //1

printf("%4.0f%10.1f\n", f, c);

f = f + step; //2

}

return 0;

}

Q1338.(10分)打印100~1000之间能同时被3、5、17整除的数。

\*\*输出格式要求："%d\n"

参考答案

#include <stdio.h>

int main()

{

int i;

for (i = 100; i <= 1000; i++)

{

if (i % 3 == 0 && i % 5 == 0 && i % 17 == 0)

{

printf("%d\n", i);

}

}

return 0;

}

Q326.(10分)编程从键盘上输入一行字符，依次读入字符显示在屏幕上。

程序运行的输入输出样例：

输入为

abcdefg

输出为

Abcdefg

参考答案

#include <stdio.h>

main()

{

char c;

c=getchar();

while(c!='\n')

{

putchar(c);

c=getchar();

}

}

Q510.(10分)从键盘输入一个大写英文字母，将其转换为小写字母后，再显示到屏幕上。

\*\*输入格式要求：提示信息："Press a key and then press Enter:"

程序运行示例如下：

Press a key and then press Enter:B

B

参考答案

#include <stdio.h>

main()

{

char ch;

printf("Press a key and then press Enter:");

ch = getchar(); /\* 从键盘输入一个字符,按回车键结束输入,该字符被存入变量ch \*/

ch = ch + 32; /\* 将大写英文字母转换为小写英文字母 \*/

putchar(ch); /\* 在屏幕上显示变量ch中的字符 \*/

putchar('\n'); /\* 输出一个回车换行控制符 \*/

}

Q508.(10分)编程从键盘输入圆的半径r，计算并输出圆的周长和面积。圆周率取值3.14159.

\*\*输入格式要求："%f" 提示信息："Input r:"

\*\*输出格式要求："circumference = %.2f\n" "area = %.2f\n"

程序运行示例如下：

Input r:3.5

circum = 21.99

area = 38.48

参考答案

#include <stdio.h>

main()

{

float r;

float circum;

float area;

printf("Input r:"); /\* 提示用户输入半径的值\*/

scanf("%f", &r); /\* 从键盘输入半径的值 \*/

circum = 2 \* 3.14159 \* r;

area = 3.14159 \* r \* r;

printf("circum = %.2f\n", circum);

printf("area = %.2f\n", area);

}

Q575.(10分)从键盘任意输入一个4位整数,编程计算并输出它的逆序数(忽略前面的正负号).例如:输入1234,分离出千位1、百位2、十位3和个位4，然后计算4\*1000+3\*100+2\*10+1=4321，并输出4321。

\*\*要求输入提示信息为:无

\*\*要求输入格式为: "%d"

\*\*要求输出格式为："%d"

程序运行示例如下：

1234 此处为输入

4321 此处为输出

参考答案

#include <stdio.h>

#include <stdlib.h>

int main()

{

int x;

int y;

int b, c, d, e;

scanf("%d", &x);

b = x / 1000;

c = (x - b \* 1000) / 100;

d = (x - 1000 \* b - c \* 100) / 10;

e = x - 1000 \* b - c \* 100 - d \* 10;

y = 1000 \* e + 100 \* d + 10 \* c + b;

printf("%d", y);

return 0;

}

Q1691.(10分)从键盘任意输入a，b，c的值，编程计算并输出一元二次方程ax2+bx+c=0的根。根据一元二次方程的求根公式，令

p=−b2a,q=b2−4ac√2a

假设a，b，c的值能保证方程有两个不相等的实根（即b2-4ac>0）

\*\*输入提示信息："Please enter the coefficients a,b,c:"

\*\*输入格式要求："%f,%f,%f"

\*\*输出格式要求：

"x1=%7.4f, x2=%7.4f\n"

参考答案

#include <math.h>

#include <stdio.h>

main()

{

float a, b, c, disc, p, q;

printf("Please enter the coefficients a,b,c:");

scanf("%f,%f,%f", &a, &b, &c); /\* 要求输入保证方程有两不等实根 \*/

disc = b \* b - 4 \* a \* c; /\* 计算判别式 \*/

p = - b / (2 \* a);

q = sqrt(disc) / (2 \* a);

printf("x1=%7.4f, x2=%7.4f\n", p + q, p - q);

}

Q211.(20分)从键盘任意输入一个3位整数，编程计算并输出它的逆序数（忽略整数前的正负号）。例如，输入-123，则忽略负号，由123分离出其百位1、十位2、个位3，然后计算3\*100+2\*10+1 = 321，并输出321。

\*\*输入格式要求："%d" 提示信息："Input x:"

\*\*输出格式要求："y = %d\n"

程序运行示例如下：

Input x:-123

y = 321

参考答案

#include <math.h>

#include <stdio.h>

main()

{

int x, b0, b1, b2, y;

printf("Input x:");

scanf("%d", &x);

x = (int)fabs(x);

b2 = x / 100; /\* 计算百位数字 \*/

b1 = (x - b2 \* 100) / 10; /\* 计算十位数字 \*/

b0 = x % 10; /\* 计算个位数字 \*/

y = b2 + b1 \* 10 + b0 \* 100;

printf("y = %d\n", y);

}

Q512.(10分)使用const常量定义圆周率pi，编程从键盘输入圆的半径r，计算并输出圆的周长和面积。输出的数据保留两位小数点。

\*\*输入格式要求："%lf" 提示信息："Input r:"

\*\*输出格式要求："printf WITHOUT width or precision specifications:\n" "circumference = %f, area = %f\n" "printf WITH width and precision specifications:\n" "circumference = %7.2f, area = %7.2f\n"

程序运行示例如下：

Input r:5.3

printf WITHOUT width or precision specifications:

circumference = 33.300854, area = 88.247263

printf WITH width and precision specifications:

circumference = 33.30, area = 88.25

参考答案

#include <stdio.h>

main()

{

const double pi = 3.14159;

double r;

double circum;

double area;

printf("Input r:");

scanf("%lf", &r);

circum = 2 \* pi \* r;

area = pi \* r \* r;

printf("printf WITHOUT width or precision specifications:\n");

printf("circumference = %f, area = %f\n", circum, area);

printf("printf WITH width and precision specifications:\n");

printf("circumference = %7.2f, area = %7.2f\n", circum, area);

}

Q509.(10分)已知三角形的三边长为a，b，c，计算三角形面积的公式为：

area=sqrt(s(s-a)(s-b)(s-c))，其中s=(a+b+c)/2。

试编程从键盘输入a，b，c的值（假设a，b，c的值可以保证其构成一个三角形），计算并输出三角形的面积。

\*\*输入格式要求："%f,%f,%f" 提示信息："Input a,b,c:"

\*\*输出格式要求："area = %f\n"

程序运行示例如下：

Input a,b,c:3,4,5

area = 6.000000

参考答案

#include <stdio.h>

#include <math.h>

main()

{

float a, b, c, s, area;

printf("Input a,b,c:");

scanf("%f,%f,%f", &a, &b, &c);

s = (a + b + c) / 2;

area = (float)sqrt(s \* (s - a) \* (s - b) \* (s - c));

printf("area = %f\n", area);

}

Q507.(10分)计算并输出一个三位整数的个位、十位和百位数字之和。

\*\*输入格式要求："%d" 提示信息："请输入一个三位整数："

\*\*输出格式要求："b2=%d, b1=%d, b0=%d, sum=%d\n"

程序运行示例如下：

请输入一个三位整数：153

b2=1, b1=5, b0=3, sum=9

参考答案

#include <stdio.h>

main()

{

int x, b0, b1, b2, sum;

printf("请输入一个三位整数：");

scanf("%d", &x);

b2 = x / 100; /\*计算百位数字\*/

b1 = (x - b2 \* 100) / 10; /\*计算十位数字\*/

b0 = x % 10; /\*计算个位数字\*/

sum = b2 + b1 + b0;

printf("b2=%d, b1=%d, b0=%d, sum=%d\n", b2, b1, b0, sum);

}

Q212.(10分)设银行定期存款的年利率rate为2.25%，已知存款期为n年，存款本金为capital元，试编程计算并输出n年后的本利之和deposit。

\*\*输入格式要求："%lf,%d,%lf" 提示信息："Please enter rate, year, capital:"

\*\*输出格式要求："deposit = %f\n"

程序运行示例如下：

Please enter rate, year, capital:0.0225,2,10000

deposit = 10455.062500

参考答案

#include <math.h>

#include <stdio.h>

main()

{

int n; /\* 存款期变量声明 \*/

double rate; /\* 存款年利率变量声明 \*/

double capital; /\* 存款本金变量声明 \*/

double deposit; /\* 本利之和变量声明 \*/

printf("Please enter rate, year, capital:");

scanf("%lf,%d,%lf", &rate, &n, &capital);

deposit = capital \* pow(1 + rate, n); /\* 计算存款利率之和 \*/

printf("deposit = %f\n", deposit); /\* 打印存款利率之和 \*/

}

Q367.(25分)用指针作为函数的形式参数，编写字符串复制函数。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："a=%s\nb=%s\n"

要求不能使用标准的字符串处理函数strcpy

字符串的输入、输出在主函数中完成

程序运行示例如下：

Enter String:happy

a=happy

b=happy

参考答案

|  |
| --- |
| main()  {      char a[30],b[30];      printf("Enter String:");      scanf("%s",a);      strcopy(a,b);      printf("a=%s\nb=%s\n",a,b);  }  strcopy(str1,str2)      char \*str1,\*str2;  {      while(\*str2++=\*str1++);  } |

Q1239.(10分)

|  |
| --- |
| 下面程序用于利用π/2 = 2/1 \* 2/3 \* 4/3 \* 4/5 \* 6/5 \* 6/7 \*... 前200项之积，编程计算π找出以下程序中的3处错误，并改正之。    #include<stdio.h>  main()  {      double term, result;      int n;      for (n = 2; n <= 100; n = n + 2)      {          term =( n \* n)/( n - 1) \* ( n + 1);          result = result \* term;      }      printf("result = %f\n", 2\*result);  } |

参考答案

|  |
| --- |
| #include <stdio.h>    main()    {        double term, result=1;        int n;        for (n = 2; n <= 100; n = n + 2)        {            term =(double)( n \* n)/(( n - 1) \* ( n + 1));            result = result \* term;        }    printf("result = %f\n", 2\*result);  } |

Q1341.(10分)计算并输出1到20的平方，当结果大于100时停止。

\*\*输出格式要求："%d \* %d = %d\n"

程序的运行示例如下：

1\*1=1

2\*2=4

3\*3=9

4\*4=16

5\*5=25

6\*6=36

7\*7=49

8\*8=64

9\*9=81

10\*10=100

### 参考答案

|  |
| --- |
| #include "stdio.h"  main()  {      int i,a,n=1;      while(n<=7)      {          do          {              scanf("%d",&a);          }while(a<1 || a>50);          for(i=1;i<=a;i++)              printf("\*");          printf("\n");          n++;      }  } |

Q437.(10分)和数能表示1~23的5个正整数

已知五个互不相同的正整数之和为23，且从这五个数中挑选若干个加起来可以表示从1

到23之内的全部自然数，问这五个数都是什么？

\*\*输入格式要求：提示信息："There are following possible result:\n"

\*\*输出格式要求："[%d]:%d,%d,%d,%d,%d\n"

程序运行示例如下：

There are following possible result:

[1]:1,2,3,5,12

[2]:1,2,3,6,11

[3]:1,2,3,7,10

[4]:1,2,4,5,11

[5]:1,2,4,6,10

[6]:1,2,4,7,9

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>    main()  {      int a,b,c,d,e,i,j,k,l,m,x,count=0,f=0;      printf("There are following possible result:\n");      for(a=1;a<=23;a++)      for(b=1+a;b<=23-a;b++)          for(c=1+b;c<=23-a-b;c++)          for(d=1+c;d<=23-a-b-c;d++)          {              f=1;              if((e=23-a-b-c-d)>d)                  for(f=0,x=1;x<24&&!f;x++)                      for(f=1,i=0;i<2&&f;i++)                          for(j=0;j<2&&f;j++)                              for(k=0;k<2&&f;k++)                                  for(l=0;l<2&&f;l++)                                      for(m=0;m<2&&f;m++)                                          if(x==a\*i+b\*j+c\*k+d\*l+e\*m) f=0;              if(!f)              printf("[%d]:%d,%d,%d,%d,%d\n",++count,a,b,c,d,e);          }  } |

Q460.(10分)读入一个整数n和整数k，程序输出n的从右向左的第k个十进制数字位值。例如，读入n=1234, k=2，则程序输出3。

\*\*输入格式要求："%d%d" 提示信息："input integer n and k:\n"

\*\*输出格式要求："%d"

程序运行示例如下：

input integer n and k:

1234 2

3

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int n, k, i, t;        printf("input integer n and k:\n");      scanf("%d%d", &n, &k);        for (i = 0; i < k; i++)      {          t = n % 10;          n = n / 10;      }        printf("%d", t);  } |

Q1306.(10分)猴子吃桃问题。猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个。第二天早上又将剩下的桃子吃掉一半，又多吃了一个。以后每天早上都吃了前一天剩下的一半零一个。到第10天早上再想吃时，见只剩一个桃子。求第一天共摘了多少桃子。

\*\*输出格式要求："桃子总数=%d\n"

程序运行示例如下：

桃子总数=xxxx

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int day, x1, x2;        day = 10;      x2 = 1;      while (day > 0)      {          x1 = (x2 + 1) \* 2; /\* 第1天的桃子数是第2天桃子数加1后的2倍 \*/          x2 = x1;          day--;      }      printf("桃子总数=%d\n", x1);        return 0;  } |

0分程序

Q307.(15分)利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

参考答案

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

Q133.(5分)设计一个函数MinCommonMultiple,其功能是计算两个正整数的最小公倍数。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include <stdio.h>

int MinCommonMultiple(int a,int b)

main()

{

int a,b,x;

printf("Input a,b:");

scanf("%d,%d",a,b);

x = MinCommonMultiple(a,b);

printf("MinCommonMultiple = %d\n", x);

}

int MinCommonMultiple(int a,int b);

{

int i;

for(i=1; i<a\*b; i++)

{

if(i%a == 0 || i%b == 0)

return i;

}

}

参考答案

int MinCommonMultiple(int a,int b);

main()

{

int a,b,x;

printf(&quot;Input a,b:&quot;);

scanf(&quot;%d,%d&quot;,&amp;a,&amp;b);

x = MinCommonMultiple(a,b);

printf(&quot;MinCommonMultiple = %d\n&quot;, x);

}

int MinCommonMultiple(int a,int b)

{

int i;

for(i=1; i&lt;a\*b; i++)

{

if(i%a == 0 &amp;&amp; i%b == 0)

return i;

}

}

Q173.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

参考答案

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q191.(15分)用二维数组编程计算并输出n(n<20)行杨辉三角形。

\*\*输入提示信息要求："Input n(n<20):\n"

\*\*输入格式："%d"

\*\*输出要求："%4d"

每输出一行后就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include<stdio.h>

#define N 20

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

main() {

int a[N][N] = {0}, n;//1

printf("Input n(n<20):\n");

scanf("%d", &n);//1

CaculateYH(a, n);//1

PrintYH(a, n); //1

}

void CaculateYH(int a[][N], int n) {

int i, j;

for (i = 0; i < n; i++) //1

{

a[i][0] = 1;//1

a[i][i] = 1;//1

}

for (i = 2; i < n; i++)//1

{

for (j = 1; j <= i - 1; j++) //1

{

a[i][j] = a[i - 1][j - 1] + a[i - 1][j];//1

}

}

}

void PrintYH(int a[][N], int n) {

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j <= i; j++) //1

{

printf("%4d", a[i][j]);//1

}

printf("\n");//1

}

}

Q235.(10分)用函数编程计算两整数的最大值，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大值。

\*\*输入格式要求："%d%d" 提示信息："Input a,b:"

\*\*输出格式要求："max = %d\n"

程序运行示例如下：

Input a,b:5 8

max = 8

参考答案

#include <stdio.h>

int Max(int a, int b);

int main()

{

int a, b;

printf("Input a,b:");

scanf("%d%d", &a, &b);

printf("max = %d\n", Max(a, b));

return 0;

}

/\* 函数功能：计算a和b的最大值 \*/

int Max(int a, int b)

{

return a > b ? a : b;

}

Q602.(10分)一辆卡车违反了交通规则，撞人后逃逸。现场有三人目击该事件，但都没有记住车号，只记住车号的一些特征。甲说：车号的前两位数字是相同的；乙说：车号的后两位数字是相同的，但与前两位不同；丙是位数学家，他说：4位的车号正好是一个整数的平方。现在请根据以上线索帮助警方找出车号以便尽快破案。

\*\*输出格式要求："The number is:%d\n"

参考答案

#include<stdio.h>

int main()

{

int a, b, n;

for (a = 1; a <= 9; a++)

{

for (b = 1; b <= 9; b++)

{

for (n = 1; n <= 100; n++)

{

if (1000 \* a + 100 \* a + 10 \* b + b == n \* n)

printf("The number is:%d\n", n \* n);

}

}

}

return 0;

}

Q282.(10分)打印所有的水仙花数。

所谓“水仙花数”是指一个三位数，其各位数字立方和等于该数本身。

例如153是“水仙花数”，因为153 = 1\*1\*1+3\*3\*3+5\*5\*5。

\*\*输出格式要求为"%6d"。

参考答案

#include <stdio.h>

int main()

{

int i,j,k,n;

for(n=100;n<1000;n++) //1

{

i=n/100; //1

j=(n-i\*100)/10; //1

k=n-i\*100-j\*10; //1

if(i\*i\*i+j\*j\*j+k\*k\*k==n) //1

printf("%6d",n);

}

printf("\n");

return 0;

}

Q413.(10分)马克思手稿中的数学问题。马克思手稿中有一道趣味数学题：有30个人，其中有男人、女人和小孩，在一家饭馆吃饭共花了50先令：每个男人花3先令，每个女人花2先令，每个小孩花1先令，问男人、女人和小孩各几人？

\*\*输入格式要求：无

\*\*输出格式要求："\tMEN\tWOMEN\tCHILDREN\n"

"-----------------------------------------\n"

"%2d:\t%d\t%d\t%d\n"

程序运行示例如下：

MEN WOMEN CHILDREN

-----------------------------------------

1: 0 20 10

2: 1 18 11

3: 2 16 12

4: 3 14 13

5: 4 12 14

6: 5 10 15

7: 6 8 16

8: 7 6 17

9: 8 4 18

10: 9 2 19

11: 10 0 20

12: 11 -2 21

13: 12 -4 22

14: 13 -6 23

15: 14 -8 24

16: 15 -10 25

17: 16 -12 26

参考答案

#include <stdio.h>

main()

{

int x, y, z, count = 0;

printf("\tMEN\tWOMEN\tCHILDREN\n");

printf("-----------------------------------------\n");

for (x = 0; x <= 16; x++)

{

y = 20 - 2 \* x;

z = 30 - x - y;

if (3 \* x + 2 \* y + z == 50)

printf("%2d:\t%d\t%d\t%d\n", ++count, x, y, z);

}

}

Q319.(10分)从键盘输入一个字符，如果它是大写的英文字母，则将其转换成小写

的英文字母，然后将转换的字符及其ASCII码值显示在屏幕上；如果它

是小写的英文字母，则将其转换成大写的英文字母，然后将转换的字符

及其ASCII码值显示在屏幕上；如果输入的是其它字符则显示输入错误。

\*\*输入提示信息格式: 无

\*\*输入数据格式要求: %c

\*\*输出数据格式要求:

若输入的是大小写英文字母，则输出: "%c,%d"

若输入的是其它字符，则输出: "Input error!"

注：已知大写英文字母A的ASCII码值是65

小写英文字母a的ASCII码值是97

参考答案

#include <stdio.h>

int main()

{

char ch;

scanf("%c", &ch); //1

if (ch >= 'A' && ch <= 'Z') //1

{

ch = ch + 32; //1

printf("%c,%d", ch, ch);

}

else if (ch >= 'a' && ch <= 'z') //1

{

ch = ch - 32; //1

printf("%c,%d", ch, ch);

}

else //1

{

printf("Input error!");

}

return 0;

}

Q408.(10分)四位反序数。设N是一个四位数，它的9倍恰好是其反序数，求N。反序数就是将整数的数字倒过来形成的整数，例如：1234的反序数是4321。

\*\*输入格式要求：无

\*\*输出格式要求："The number satisfied states condition is : %d\n"

参考答案

#include <stdio.h>

main()

{

int i;

for(i=1002;i<1111;i++)

if(i%10\*1000+i/10%10\*100+i/100%10\*10+i/1000==i\*9)

printf("The number satisfied states condition is : %d\n",i);

}

Q209.(10分)已知三角形的三边长为a，b，c，计算三角形面积的公式为：

area =sqrt(s(s-a)(s-b)(s-c)),s=1/2(a+b+c)

其中，a，b，c为浮点数，sqrt为开平方计算。

要求编写程序，从键盘输入a，b，c的值，计算并输出三角形的面积。

注意：不用判断输入的边长值是否满足三角形要求，

只需进行简单计算即可。

\*\*要求输入提示信息格式为："Input a,b,c:\n"；

\*\*输出格式要求："area=%.2f\n"

参考答案

#include <stdio.h>

#include <math.h>

main()

{

float a, b, c;

float s, area;

printf("Input a,b,c:");

scanf("%f,%f,%f",&a,&b,&c); //1

s = 1.0 / 2 \* (a + b + c); //1

area = sqrt(s \* (s - a) \* (s - b) \* (s - c)); //1

printf("area=%.2f\n", area); //1

}

Q1738.(10分)编程打印以下图案。

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*输出格式要求："%c"

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int i, j, k;      for (i = 1; i <= 4; i++)     //i控制行数      {          for (j = 1; j <= 4 - i; j++) // 随着行数的增加，输出递减数目的空格          {              printf(" ");          }          for (k = 1; k <= 6; k++) //每行输出6个\*字符          {              printf("\*");          }          printf("\n");           //将光标移到下一行起始位置处      }  } |

Q1319.(10分)下面程序用于打印所有的水仙花数。所谓“水仙花数”，是指一个三位数，其各位数字的立方和等于该数本身。例如，153是“水仙花数”，因为153=1^3+3^3+5^3。

参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i, j, k, n;        printf("result is:");      for (n=100; n<1000; n++)      {          i = n / 100;          j = (n - i \* 100) / 10;          k = n % 10;          if (i\*100+j\*10+k == i\*i\*i+j\*j\*j+k\*k\*k)          {              printf("%d\t ",n);          }      }      printf("\n");  } |

Q488.(10分)输出100以内的质数，每行5个。

\*\*输出格式要求："%3d"

程序运行结果如下：

2 3 5 7 11

13 17 19 23 29

31 37 41 43 47

53 59 61 67 71

73 79 83 89 97

参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int m, k, i, n = 0;//n为累加质数个数        for (m = 2; m <= 100; m++)      {          k = m - 1;          for (i = 2; i <= k; i++)              if (m % i == 0)                  break;          if (i <= k)              continue;          n++;          printf("%3d", m);          if (n % 5 == 0)  //如果本行已经打印5个质数，换行              printf("\n");      }        return 0;  } |

Q1308.(10分)用牛顿迭代法求方程2x^3-4x^2+3x-6=0在1.5附近的根。

\*\*输出格式要求："方程的根=%6.2f\n"

程序的运行示例如下：

方程的根= 2.00

参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      float x, x0, f, f1;        x = 1.5;        do      {          x0 = x;          f = ((2 \* x0 - 4) \* x0 + 3) \* x0 - 6;          f1 = (6 \* x0 - 8) \* x0 + 3;          x = x0 - f / f1;      }      while (fabs(x - x0) >= 1e-5);      printf("方程的根=%6.2f\n", x);        return 0;  } |

Q489.(10分)百元买百鸡问题：“公鸡5元一只；母鸡3元一只；雏鸡一元3只。百钱买百鸡，问鸡翁、母、雏各几只？”

\*\*输出格式要求："公鸡是%d只，母鸡是%d只，雏鸡是%d只.\n"

程序运行结果如下：

公鸡是0只，母鸡是25只，雏鸡是75只.

公鸡是4只，母鸡是18只，雏鸡是78只.

公鸡是8只，母鸡是11只，雏鸡是81只.

公鸡是12只，母鸡是4只，雏鸡是84只.

参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int x, y, z;        for (x = 0; x <= 19; x++)      {          for (y = 0; y <= 33; y++)          {              z = 100 - x - y;              if (5 \* x + 3 \* y + z / 3 == 100 && z % 3 == 0)                  printf("公鸡是%d只，母鸡是%d只，雏鸡是%d只.\n", x, y, z);          }      }        return 0;  } |

Q1734.(10分)

按如下下三角格式，编程输出九九乘法表。

\*\*输出格式要求："%4d"

参考答案

|  |
| --- |
| #include  <stdio.h>    main()  {      int  m, n;      for (m = 1; m < 10; m++)               //外层循环控制被乘数m从1变化到9      {          for (n = 1; n <= m; n++)           //内层循环控制乘数n从1变化到m          {              printf("%4d", m \* n);          //输出第m行n列的表值          }          printf("\n");                      //控制输出换行，准备打印下一行      }  } |

Q415.(10分)递推法。 8除不尽的自然数。一个自然数被8除余1，所得的商被8除也余1，再将第二次的商被8除后余7，最后得到一个商为a，又知这个自然数被被17除余4，所得的商被17除余15，最后得到一个商为a的2倍，求这个自然数。

\*\*输出格式要求："The required number is :%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int i;   for(i=0; ;i++)     if(((i\*8+7)\*8+1)\*8+1==(34\*i+15)\*17+4)     {         printf("The required number is :%d\n",(34\*i+15)\*17+4);         break;     }    } |

Q406.(10分)穷举，抓交通肇事犯。一辆卡车违犯交通规则，撞人后逃跑。现场有三人目击事件，但都没记住车号，只记下车号的一些特征。甲说：牌照的前两位数字是相同的；乙说：牌照的后两位数字是相同的，但与前两位不同；丙是位数学家，他说：四位的车号刚好是一个整数的平方。请根据以上线索求出车号。

\*\*输出格式要求："Lorry\_No. is %d .\n"

### 参考答案

|  |
| --- |
| #include "math.h"  main()  {int i,j,k,c;   for(i=1;i<=9;i++)     for(j=0;j<=9;j++)       if(i!=j)       { k=i\*1000+i\*100+j\*10+j;         for(c=31;c\*c<k;c++);         if(c\*c==k)            printf("Lorry\_No. is %d .\n",k);       }  } |

Q425.(10分)回文数的形成。任取一个十进制整数，将其倒过来后与原来的整数相加，得到一个新的整数后，重复以上步骤，最终可得到一个回文数，请编程验证。

\*\*输入格式要求："%ld" 提示信息："please enter a number optionaly:" "The generation process of palindrome:\n" " input error, break.\n"

\*\*输出格式要求：" [%d]: %ld+%ld=%ld\n" "Here we reached the aim at last !\n"

程序运行示例如下：

please enter a number optionaly:345

The generation process of palindrome:

[1]: 345+543=888

Here we reached the aim at last !

### 参考答案

|  |
| --- |
| #define MAX 2147483647   #include <stdio.h>   main()   {long int n,m,re();    int  count=0;    printf("please enter a number optionaly:");    scanf("%ld",&n);    printf("The generation process of palindrome:\n");    while(!nonre((m=re(n))+n))    {         if(m+n>=MAX)         {  printf("  input error, break.\n");            break;         }         else         {printf("   [%d]: %ld+%ld=%ld\n",++count,n,m,m+n);          n+=m;         }    }    printf("   [%d]: %ld+%ld=%ld\n",++count,n,m,m+n);      printf("Here we reached the aim at last !\n");  }  long re(a)     long int a;  {  long int t;     for(t=0;a>0;a/=10)       t=t\*10+a%10;       return(t);  }  nonre(s)     long int s;     {if(re(s)==s)        return(1);      else        return(0);     } |

Q500.(10分)利用指针能直接操作内存的特点来编程输出字符串中每个字符在内存中的存储编码（字符串中可以包含英文数字和汉字）。

程序的运行示例1如下：

请输入一个字符串，长度小于等于100：abcABC012中国人

该字符串的内存编码为： 61 62 63 41 42 43 30 31 32 d6 d0 b9 fa c8 cb

程序的运行示例2如下：

请输入一个字符串，长度小于等于100：12345上山打老虎

该字符串的内存编码为： 31 32 33 34 35 c9 cf c9 bd b4 f2 c0 cf bb a2

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main(int argc, char \*argv[])  {      char str[101] = {'\0'};      char \*p1, \*p2;        printf("请输入一个字符串，长度小于等于100：");      scanf("%s", str);        p1 = (char \*)(str + strlen(str));      p2 = (char \*)str;        printf("该字符串的内存编码为： ");      while (p1 > p2)      {          printf("%x ", (unsigned char)\*p2 & 0xff);          p2++;      }        return 0;  } |

Q617.(10分)编写一个函数，输入n为偶数时，调用函数求1/2+1/4+...+1/n,当输入n为奇数时，调用函数 求1/1+1/3+...+1/n(要求利用函数指针编程)

程序运行示例1如下：

10

Even=2.141666

程序运行示例2如下：

9

Odd=1.787302

### 参考答案

|  |
| --- |
| #include "stdio.h"    float peven(int n)  {      float s;      int i;      s = 1;      for (i = 2; i <= n; i += 2)          s += 1 / (float)i;      return(s);  }    float podd(int n)  {      float s;      int i;      s = 0;      for (i = 1; i <= n; i += 2)          s += 1 / (float)i;      return(s);  }    float dcall(float (\*fp)(), int n)  {      float s;      s = (\*fp)(n);      return(s);  }    main()  {      float sum;      int n;      while (1)      {          scanf("%d", &n);          if (n > 1)              break;      }      if (n % 2 == 0)      {          printf("Even=");          sum = dcall(peven, n);      }      else      {          printf("Odd=");          sum = dcall(podd, n);      }      printf("%f", sum);  } |

Q534.(10分)编程求解汉诺塔问题。

汉诺塔（Hanoi）是必须用递归方法才能解决的经典问题。它来自于印度神话。上帝创造世界时作了三根金刚石柱子，在第一根柱子上从下往上按大小顺序摞着64片黄金圆盘，如图7-3所示。上帝命令婆罗门把圆盘从下面开始按大小顺序重新摆放到第二根柱子上，并且规定，每次只能移动一个圆盘，在小圆盘上不能放大圆盘。有人预言说，这件事完成时宇宙会在一瞬间闪电式毁灭，也有人相信婆罗门至今仍在一刻不停地搬动着圆盘。

\*\*输入格式要求："%d" 提示信息："Input the number of disks:"

\*\*输出格式要求："Steps of moving %d disks from A to B by means of C:\n" "Move %d: from %c to %c\n"

程序运行示例如下：

Input the number of disks:3

Steps of moving 3 disks from A to B by means of C:

Move 1: from A to B

Move 2: from A to C

Move 1: from B to C

Move 3: from A to B

Move 1: from C to A

Move 2: from C to B

Move 1: from A to B

参考答案

#include <stdio.h>

void Hanoi(int n, char a, char b, char c);

void Move(int n, char a, char b);

int main()

{

int n;

printf("Input the number of disks:");

scanf("%d", &n);

printf("Steps of moving %d disks from A to B by means of C:\n", n);

Hanoi(n, 'A', 'B', 'C'); /\*调用递归函数Hanoi()将n个圆盘借助于C由A移动到B\*/

return 0;

}

/\* 函数功能：用递归方法将n个圆盘借助于柱子c从源柱子a移动到目标柱子b上 \*/

void Hanoi(int n, char a, char b, char c)

{

if (n == 1)

{

Move(n, a, b); /\* 将第n个圆盘由a移到b \*/

}

else

{

Hanoi(n - 1, a, c, b); /\* 递归调用Hanoi()，将第n-1个圆盘借助于b由a移动到c\*/

Move(n, a, b); /\* 第n个圆盘由a移到b \*/

Hanoi(n - 1, c, b, a); /\*递归调用Hanoi()，将第n-1个圆盘借助于a由c移动到b\*/

}

}

/\* 函数功能： 将第n个圆盘从源柱子a移到目标柱子b上 \*/

void Move(int n, char a, char b)

{

printf("Move %d: from %c to %c\n", n, a, b);

}

Q247.(10分)输入m×n阶矩阵A和B，用函数编程计算并输出A与B之和。

\*\*输入格式要求："%d,%d" 提示信息："Input m, n:" "Input %d\*%d matrix a:\n" "Input %d\*%d matrix b:\n"

\*\*输出格式要求："Results:\n" "%6d"

程序运行示例如下：

Input m, n:2,3

Input 2\*3 matrix a:

1 2 3

4 5 6

Input 2\*3 matrix b:

11 12 13

14 15 16

Results:

12 14 16

18 20 22

### 参考答案

|  |
| --- |
| #include<stdio.h>  #define  ROW 10  #define  COL 10  void  InputMatrix(int a[ROW][COL], int m, int n);  void  AddMatrix(int a[ROW][COL], int b[ROW][COL], int c[ROW][COL],                  int m, int n);  void  PrintMatrix(int a[ROW][COL], int m, int n);  int main()  {      int a[ROW][COL], b[ROW][COL], c[ROW][COL], m, n;      printf("Input m, n:");      scanf("%d,%d", &m, &n);      printf("Input %d\*%d matrix a:\n", m, n);      InputMatrix(a, m, n);      printf("Input %d\*%d matrix b:\n", m, n);      InputMatrix(b, m, n);      AddMatrix(a, b, c, m, n);      printf("Results:\n");      PrintMatrix(c, m, n);      return 0;  }  /\*  函数功能： 输入m×n矩阵的元素值，存于数组a中 \*/  void  InputMatrix(int a[ROW][COL], int m, int n)  {      int i, j;      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              scanf("%d", &a[i][j]);          }      }  }  /\* 函数功能：计算m×n矩阵之和，即对应位置元素之和，结果存于数组c中 \*/  void  AddMatrix(int a[ROW][COL], int b[ROW][COL], int c[ROW][COL],                  int m, int n)  {      int i, j;      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              c[i][j] = a[i][j] + b[i][j];          }      }  }  /\* 函数功能： 输出m×n矩阵的元素值 \*/  void  PrintMatrix(int a[ROW][COL], int m, int n)  {      int i, j;      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              printf("%6d", a[i][j]);          }          printf("\n");      }  } |

Q254.(10分)用函数编程实现两个数组中对应元素值的交换。

\*\*输入格式要求："%d" 提示信息："Input array size(n<=10):" "Input array a:" "Input array b:"

\*\*输出格式要求："Output array a:" "Output array b:" "%5d"

程序运行示例如下：

Input array size(n<=10):5

Input array a:

1 2 3 4 5

Input array b:

6 7 8 9 10

Output array a: 6 7 8 9 10

Output array b: 1 2 3 4 5

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #define N 10  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void Swap(int \*x, int \*y);  int main()  {      int  a[N], b[N], i, n;      printf("Input array size(n<=10):");      scanf("%d", &n);      printf("Input array a:");      ReadData(a, n);      printf("Input array b:");      ReadData(b, n);      for (i = 0; i < n; i++)      {          Swap(&a[i], &b[i]);      }      printf("Output array a:");      PrintData(a, n);      printf("Output array b:");      PrintData(b, n);      return 0;  }  /\* 函数功能：输入数组a的n个元素值 \*/  void ReadData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          scanf("%d", &a[i]);      }  }  /\* 函数功能：输出数组a的n个元素值 \*/  void PrintData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          printf("%5d", a[i]);      }      printf("\n");  }  /\* 函数功能：两整数值互换 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  } |

Q1326.(10分)有n个人围成一圈，顺序排号。从第一人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

程序的运行示例如下：

请输入人数：30

30个人中最后留下的是第29号。

### 参考答案

|  |
| --- |
| /\* n个人循环报数逢3退出求最后一人是第几号 \*/    #include <stdio.h>    #define     NMAX    50 /\* 定义最多人数为50人 \*/    int main(void)  {      int i, k, m, n, num[NMAX], \*p;        printf("请输入人数：");      scanf("%d", &n);        p = num;      for (i = 0; i < n; i++)      {          \*(p + i) = i + 1; /\* 以1至n为序给每个人编号 \*/      }      i = 0;      k = 0; /\* k为按1,2,3报数时的计数变量 \*/      m = 0; /\* m为退出人数 \*/      while (m < n - 1)      {          if (\*(p + i) != 0)          {              k++;          }          if (k == 3)          {              \*(p + i) = 0; /\* 对退出人员的编号置为0 \*/              k = 0;              m++;          }          i++;          if (i == n)          {              i = 0; /\* 报数到尾后i恢复为0 \*/          }      }      while (\*p == 0)      {          p++;      }      printf("%d个人中最后留下的是第%d号。", n, \*p);        return 0;  } |

Q502.(10分)编程统计候选人的得票数。设有3个候选人，每次输入一个得票候选人的名字，不考虑弃权情况，要求最后输出各个候选人的得票结果（候选人姓名和参加投票人数由程序运行时输入）。

要求输入、输出在主函数中实现，统计候选人由函数实现。

按如下函数原型编写程序

void Vote(struct Person ld[], int n)

程序的运行示例如下：

输入3个候选人的基本信息：姓名

wang

song

zhao

输入参加投票的人数:5

输入5个参加投票人的投票情况

wang

wang

song

song

zhao

输出3个候选人的基本信息：姓名，票数

wang 2

song 2

zhao 1

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define N 3    struct Person  {      char name[16];      int count;  };    void Input(struct Person ld[], int n);  //输入n个候选人的基本信息  void Vote(struct Person ld[], int n);   //对n个候选人进行投票  void Output(struct Person ld[], int n); //输出n个候选人的基本信息    int main(int argc, char \*argv[])  {      struct Person leader[N];        Input(leader, N);      Vote(leader, N);      Output(leader, N);        return 0;  }    void Input(struct Person ld[], int n)  {      int i;        printf("输入%d个候选人的基本信息：姓名\n", n);      for (i = 0; i < n; i++)          scanf("%s", ld[i].name);  }    void Vote(struct Person ld[], int n)  {      int m, i, j; //m表示参加投票的人数      char ename[16];        printf("输入参加投票的人数:");      scanf("%d", &m);      printf("输入%d个参加投票人的投票情况\n", m);      for (i = 0; i < m; i++)      {          ld[i].count = 0;      }        for (i = 0; i < m; i++)      {          scanf("%s", ename);          for (j = 0; j < n; j++)              if (strcmp(ld[j].name, ename) == 0)                  ld[j].count++;      }  }    void Output(struct Person ld[], int n)  {      int i;        printf("输出%d个候选人的基本信息：姓名，票数\n", n);      for (i = 0; i < n; i++)          printf("%s %d\n", ld[i].name, ld[i].count);  } |

Q384.(30分)求复数之积。利用结构变量求解两个复数之积。

za={3,4}, zb={5,6}

za={10,20}, zb={30,40}

\*\*输出格式要求："(%d+%di)\*(%d+%di)=" "(%d+%di)\n"

程序运行示例如下：

(3+4i)\*(5+6i)=(-9+38i)

(10+20i)\*(30+40i)=(-500+1000i)

### 参考答案

|  |
| --- |
| #include "stdio.h"    struct complx  {    int real;    int im;  };  main()  {    static struct complx za={3,4};    static struct complx zb={5,6};    struct complx x,y,z;    struct complx cmult();    void cpr();    z=cmult(za,zb);    cpr(za,zb,z);    x.real = 10; x.im = 20;    y.real = 30; y.im = 40;    z=cmult(x,y);    cpr(x,y,z);  }  struct complx cmult (za,zb)    struct complx za,zb;  {        struct complx w;        w.real = za.real\*zb.real - za.im\*zb.im;        w.im   = za.real\*zb.im+za.im\*zb.real;        return (w);  }  void cpr(za,zb,z)     struct complx za,zb,z;  {      printf("(%d+%di)\*(%d+%di)=",za.real,za.im,zb.real,zb.im);      printf("(%d+%di)\n",z.real,z.im);  } |

Q1352.(10分)有这样一个小游戏，你可以按照下面的步骤一步一步地完成：

（1）你一般一个星期中哪一天最忙呢？如果是星期一，则选1，如果是星期二，则选2，依此类推，从1、2、3、4、5、6、7中选一个数字；

（2）将这个数字乘2；

（3）如果你是男生，则将得到的结果加4，如果你是女生，则将得到的结果加8；

（4）将得到的结果乘以50；

（5）如果你是在1~6月出生，则将得到的结果加2002，如果你是在7~12月出生，则将得到的结果加3002；

（6）将得到的结果减去你出生的年份（例如1988）；

（7）取结果的后两位，得到一个两位数，将这个两位数加10。

你一定会感到惊奇，最后得到的结果竟然是你现在的年龄。请编程实现上述游戏的计算过程，并输出年龄的计算结果。(要求对不符合要求的输入提示"Invalid input." 并不再继续执行以下程序)

输入输出格式：

输出格式控制字符串：

"Please input your busiest day.\n"

"Please input your gender.\n"

"Please input your birthday.\n"

"Your age is:%d\n"

"Invalid input.\n"

输入格式控制字符串：

"%d"

"%c"

"%4d-%2d-%2d"

输入输出样例：

1)

输出：Please input your busiest day.

输入：1

输出：Please input your gender.

输入：f

输出：Please input your birthday.

输入：1988-01-01

输出：Your age is:24

2)

输出：Please input your busiest day.

输入：1

输出：Please input your gender.

输入：m

输出：Please input your birthday.

输入：1988-01-01

输出：Your age is:24

3)

输出：Please input your busiest day.

输入：8

输出：Invalid input.

4）

输出：Please input your busiest day.

输入：1

输出：Please input your gender.

输入：w

输出：Invalid input.

5）

输出：Please input your busiest day.

输入：1

输出：Please input your gender.

输入：m

输出：Please input your birthday.

输入：1987-02-29

输出：Invalid input.

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int sum, year, month, day;      char gender;      printf("Please input your busiest day.\n");      scanf("%d", &sum);      if (sum < 1 || sum > 7)          printf("Invalid input.\n");      else      {          sum \*= 2;          printf("Please input your gender.\n");          scanf(" %c", &gender);          if (gender != 'f' && gender != 'm')          {              printf("Invalid input.\n");              return 0;          }          if (gender == 'f')          {              sum += 4;          }          if (gender == 'm')          {              sum += 8;          }          printf("Please input your birthday.\n");          scanf("%4d-%2d-%2d", &year, &month, &day);          switch (month)          {          case 1:          case 3:          case 5:          case 7:          case 8:          case 10:          case 12:              if (day < 1 || day > 31)              {                    printf("Invalid input.\n");                  return 0;              }              break;          case 4:          case 6:          case 9:          case 11:              if (day < 1 || day > 30)              {                    printf("Invalid input.\n");                  return 0;              }              break;          case 2:              if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))              {                  if (day < 1 || day > 29)                  {                        printf("Invalid input.\n");                      return 0;                  }              }              else              {                  if (day < 1 || day > 28)                  {                        printf("Invalid input.\n");                      return 0;                  }              }              break;          default:              printf("Invalid input.\n");              return 0;          }          sum \*= 50;          if (1 <= month && month <= 6)              sum += 2002;          else              sum += 3002;          sum -= year;          sum = sum % 100 + 10;          printf("Your age is:%d\n", sum);          return 0;      }      return 0;  } |

Q1324.(10分)用递归法将一个整数n转换成字符串，例如输入483，应输出字符串“483”。n的位数不确定，可以是任意位数的整数。

程序的运行示例如下：

输入整数：345

输出是：345

### 参考答案

|  |
| --- |
| #include <stdio.h>    void convert(int n)  {      int i;        if ((i = n / 10) != 0)      {          convert(i);      }      putchar(n % 10 + '0');  }    int main()  {      int number;        printf("\n输入整数：");      scanf("%d", &number);      printf("\n输出是：");      if (number < 0)      {          putchar('-');          number = -number;      }      convert(number);        return 0;  } |

Q451.(10分)计算由键盘输入的任意两个双精度数据的平均值。

\*\*输入格式要求："%lf,%lf"

\*\*输出格式要求："The average is :%f"

程序运行示例如下：

30，90

The average is :60.000000

### 参考答案

|  |
| --- |
| #include <stdio.h>    void main()  {      double x, y, a;        scanf("%lf,%lf", &x, &y);      a = (x + y) / 2;        printf("The average is :%f", a);  } |

Q1367.(10分)下面程序实现折半查找算法，当找到输入元素后显示其在数组中的下标。找出其中的错误，并改正之

|  |
| --- |
| #include <stdio.h>  main()  {      int  up=10, low=1, mid, found, find;      int  a[10]={1, 5, 6, 9, 11, 17, 25, 34, 38, 41};        scanf("%d", find);      printf("\n");        while (up>=low  ||  !found)      {          mid=(up+low)/2;          if( a[mid] = find )          {                      found=1;                  break;          }           else  if(a[mid]>find)                  up=mid+1;              else                          low=mid+1;     }        if(found)  printf("found  number  is  %dth", mid);      else    printf("no  found");  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  main( )  {  int  up=9, low=0, mid,found=0, find;    int  a[10]={1, 5, 6, 9, 11, 17, 25, 34, 38, 41};      scanf("%d", &find);    printf("\n");    while (up>=low  &&  !found)         {                  mid=(up+low)/2;              if(a[mid]==find)                  {                              found=1;                              break;                          }              else  if(a[mid]>find)                  up=mid-1;              else                          low=mid+1;          }  if(found) printf("found number is %dth",mid);  else  printf("no  found");  } |

Q543.(10分)从键盘输入某班学生某门课的成绩（每班人数最多不超过40人），当输入为负值时，表示输入结束，试编程从键盘任意输入一个学号，查找该学号学生的成绩。

\*\*输入格式要求："%ld"(学号) "%ld%d" 提示信息："Total students are %d\n" "Input the searching ID:" "Input student’s ID and score:"

\*\*输出格式要求："score = %d\n" "Not found!\n"

程序的两次运行示例如下：

① Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

Input the searching ID:070310123

score = 83

② Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

Input the searching ID:070310128

Not found!

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define N 40  int ReadScore(int score[], long num[]);          /\*ReadScore()函数原型\*/  int LinSearch(long num[], long x, int n);        /\* LinSearch()原函数型\*/  int main()  {      int score[N], n, pos;      long num[N], x;      n = ReadScore(score, num);  /\* 输入成绩和学号，返回学生总数 \*/      printf("Total students are %d\n", n);      printf("Input the searching ID:");      scanf("%ld", &x);            /\* 以长整型格式从键盘输入待查找的学号x \*/      pos = LinSearch(num, x, n); /\* 查找学号为num的学生 \*/      if (pos != -1)                   /\* 若找到，则打印其分数 \*/      {          printf("score = %d\n", score[pos]);      }      else                            /\* 若未找到，则打印"未找到"提示信息\*/      {          printf("Not found!\n");      }      return 0;  }  /\* 函数功能：输入学生的学号及其某门课成绩，当输入负值时，结束输入，返回学生人数 \*/  int ReadScore(int score[], long num[]) /\* ReadScore()函数定义 \*/  {      int i = -1;         /\*i初始化为-1，循环体内增1后可保证数组下标从0开始\*/      do      {          i++;          printf("Input student’s ID and score:");          scanf("%ld%d", &num[i], &score[i]);      }      while (num[i] > 0 && score[i] >= 0);  /\* 输入负值时结束成绩输入 \*/      return i;                                   /\* 返回学生总数 \*/  }  /\*按线性查找法查找值为x的数组元素，若找到则返回x在数组中的下标位置，否则返回-1\*/  int LinSearch(long num[], long x, int n) /\* LinSearch()函数定义 \*/  {      int  i;      for (i = 0; i < n; i++)      {          if (num[i] == x)    return (i); /\* 若找到则返回x在数组中的下标 \*/      }      return (-1);                           /\* 若循环结束仍未找到，则返回-1 \*/  } |

Q554.(10分)人工方式下比较分数大小最常用的方法是：进行分数的通分后比较分子的大小。编程实现模拟该过程。

\*\*输入格式要求："%d/%d,%d/%d" 提示信息："Input two fractions\n"

\*\*输出格式要求："%d/%d>%d/%d\n" "%d/%d=%d/%d\n" "%d/%d<%d/%d\n"

运行结果示例：

Input two fractions

4/5,6/7

4/5<6/7

### 参考答案

|  |
| --- |
| #include<stdio.h>  int LCM(int a, int b)  {      long int c;      int d;      if (a < b)      {          c = a;          a = b;          b = c;      }      for (c = a \* b; b != 0;)      {          d = b;          b = a % b;          a = d;      }      return (int)c / a;  }  int main()  {      int i, j, k, l, m, n, lcm;      printf("Input two fractions\n");      scanf("%d/%d,%d/%d", &i, &j, &k, &l);      lcm = LCM(j, l);      m = lcm / j \* i;      n = lcm / l \* k;      if (m > n)          printf("%d/%d>%d/%d\n", i, j, k, l );      else if (m == n)          printf("%d/%d=%d/%d\n", i, j, k, l );      else          printf("%d/%d<%d/%d\n", i, j, k, l );        return 0;  } |

Q601.(10分)在一个字符串中找出最大的字符并放在第一个位置上，并将该字符前的原字符往后顺序移动，如：cbyab变成ycbab。

\*\*输入格式要求：用gets读入

\*\*输出格式要求：用puts输出

程序运行的输入输出样例：

输入为：

cbyab

输出为：

Ycbab

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char str[80], \*p = str, \*q, max;        gets(p);      max = \*(p++);        while (\*p != 0)      {          if (max < \*p)          {              max = \*p;              q = p;          }          p++;      }        p = q;        while (p > str)      {          \*p = \*(p - 1);          p--;      }        \*p = max;      puts(p);  } |

Q1740.(10分)编程打印以下图案。

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int  i , j, k ;      for (i = 1; i <= 4; i++)     // 控制行数      {          for (j = 1; j <= 8 - i; j++) // 随着行数的增加，输出递减数目的空格          {              printf(" ");          }          for (k = 1; k <= (2 \* i - 1); k++)          {              printf("\*");          }          printf("\n");      }  } |

Q563.(10分)求具有abcd=(ab+cd)2性质的四位数。

\*\*输入格式要求：提示信息："There are following numbers:\n"

\*\*输出格式要求："%d "

程序运行示例如下：

There are following numbers:

2025 3025 9801

### 参考答案

|  |
| --- |
| #include<stdio.h>  void main()  {      int n, a, b;      printf("There are following numbers:\n");      for (n = 1000; n < 10000; n++)      /\*四位数N的取值范围1000~9999\*/      {          a = n / 100;                    /\*截取N的前两位数存于a\*/          b = n % 100;                    /\*截取N的后两位存于b\*/          if ((a + b) \* (a + b) == n) /\*判定N是否为符合题目所规定的性质的四位数\*/              printf("%d  ", n);      }  } |

Q276.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define SIZE 80  int MyStrcmp(char s[],char t[]);  main()  {      char s[SIZE],t[SIZE],i;      printf("Input s\n");      gets(s);//1      printf("Input t\n");      gets(t);//1      i=MyStrcmp(s,t);//2      if(i>0)//1          printf("string s>string t.\n");      else if(i<0)//1          printf("string s<string t.\n");      else//1          printf("string s=string t.\n");  }  int MyStrcmp(char s[], char t[])  {      int i;      for (i=0; s[i] == t[i]; i++ ) //6      {              if (s[i] ==  '\0' )  //2              return 0 ;//2      }      return (s[i]-t[i]); //2  } |

Q400.(10分)已知求正弦sin(x)的近似值的多项式公式为：

sin(x)=x−x 3  3!  +x 5  5!  −x 7  7!  +⋯+(−1) n x 2n+1  (2n+1)!  +…

要求输入x和e，按上述公式计算sin(x)的近似值，要求计算的误差小于给定的e。

\*\*输入格式要求："%f%f" 提示信息："Enter x & eps:"

\*\*输出格式要求："sin(%f)=%f\n" "%d,sin(%f)=%f\n"

程序运行示例如下：

Enter x & eps:5 0.000001

sin(5.000000)=-0.958924

13,sin(5.000000)=-0.958924

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  { float x,eps,s,y=0,y0,t;    int n,j;    printf("Enter x & eps:");    scanf ("%f%f",&x,&eps);    n=t=j=1;    s=x;    do    { y0=y;      if(n%2==0) y=y-s/t;      else       y=y+s/t;      s\*=x\*x;      t\*=(j+1)\*(j+2);      j+=2;      n++;    }while(fabs(y0-y)>eps);    printf("sin(%f)=%f\n",x,sin(x));    printf("%d,sin(%f)=%f\n",n,x,y);  } |

Q1353.(10分)某人三天打渔两天晒网，假设他从1990年1月1日开始打渔三天，然后晒网两天，请编程回答任意的一天他在打渔还是晒网。

A boy works for 3 days while has a 2 days off. If he is working on 1st, Jan, 1990, then for a date entered from the keyboard, please write a program to determine what the boy is doing, working or resting?

Examples of input and output:

1)Input:

1990-01-05

Output:

He is having a rest.

2)Input:

1990-01-07

Output:

He is working.

3)Input:

1990-01-33

Output:

Invalid input.

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>    int main()  {      int y, m, d;      int sum = 0;      scanf("%4d-%2d-%2d", &y, &m, &d);      if (y < 1990)      {          printf("Invalid input.");          return 0;      }      if ((y % 100 != 0 && y % 4 == 0) || (y % 400 == 0))      {          switch (m)          {            case 1:          case 3:          case 5:          case 7:          case 8:          case 10:          case 12:              if (d > 31 || d < 1)              {                  printf("Invalid input.");                  return 0;              }              break;          case 2:              if (d > 29 || d < 1)              {                  printf("Invalid input.");                  return 0;              }              break;          case 4:          case 6:          case 9:          case 11:              if (d > 30 || d < 1)              {                  printf("Invalid input.");                  return 0;              }              break;          default:              printf("Invalid input.");              return 0;              break;          }      }      else      {          switch (m)          {          case 1:          case 3:          case 5:          case 7:          case 8:          case 10:          case 12:              if (d > 31 || d < 1)              {                  printf("Invalid input.");                  return 0;              }              break;          case 2:              if (d > 28 || d < 1)              {                  printf("Invalid input.");                  return 0;              }              break;          case 4:          case 6:          case 9:          case 11:              if (d > 30 || d < 1)              {                  printf("Invalid input.");                  return 0;              }              break;          default:              printf("Invalid input.");              return 0;              break;          }      }      int i;      if (y > 1990)      {          for (i = 1990; i < y; i++)          {              if ((i % 100 != 0 && i % 4 == 0) || (i % 400 == 0))              {                  sum = sum + 366;              }              else              {                  sum = sum + 365;              }          }        }          int i1;      if (m > 1)      {          for (i1 = 1; i1 < m; i1++)          {              if ((y % 100 != 0 && y % 4 == 0) || (y % 400 == 0))              {                  switch (i1)                  {                  case 1:                  case 3:                  case 5:                  case 7:                  case 8:                  case 10:                  case 12:                      sum = sum + 31;                      break;                  case 2:                      sum = sum + 29;                      break;                  case 4:                  case 6:                  case 9:                  case 11:                      sum = sum + 30;                      break;                  }              }              else              {                  switch (i1)                  {                  case 1:                  case 3:                  case 5:                  case 7:                  case 8:                  case 10:                  case 12:                      sum = sum + 31;                      break;                  case 2:                      sum = sum + 28;                      break;                  case 4:                  case 6:                  case 9:                  case 11:                      sum = sum + 30;                      break;                  }              }          }      }      sum = sum + d;      sum = sum % 5;      if (sum == 0 || sum == 4)      {          printf("He is having a rest.");      }      else      {          printf("He is working.");      }      return 0;  } |

Q277.(20分)编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #include  <string.h>  #define N 80  void main()  {      int    n;      char  str[N], min[N];      printf("Input five countries' names:\n");      gets(str);//2      strcpy(min, str);  //2      for (n=1; n<5; n++)//3      {          gets(str); //2          if (strcmp(str, min) < 0)    //4          {              strcpy(min, str);  //2          }      }      printf("The minimum is:%s\n", min);//2  } |

Q121.(30分)编程计算分段：

y= x x<1

y= 2x-1 1<=x<10

y= 3x-11 x>=10

从键盘输入一个单精度实数x，打印出y值。

\*\*输入提示信息格式要求为："Please input x:"；

\*\*输出格式要求为"y = %.2f\n"。

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {     float x,y;     printf("Please input x:");     scanf("%f",&x);  //1     if(x<1)   //1        y=x;     else if(x>=1 && x<10)  //1        y=2\*x-1;     else if(x>=10)    //1        y=3\*x-11;     printf("y = %.2f\n",y);  //1  } |

Q196.(25分)利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 80  void Inverse(char str[]);  main()  {      char a[N];      printf("Input a string:\n");      gets(a);//1      Inverse(a);//2      printf("Inversed results:\n");      puts(a);//1  }  void Inverse(char str[])  {      int   len, i, j;      char  temp;//1      len = strlen(str);//1      for (i=0, j=len-1; i<j; i++, j--)//4      {      temp = str[i];//1          str[i] = str[j];//1          str[j] = temp;//1      }  } |

Q779.(15分)请按给定的函数原型编程实现两个数组中对应该元素值的交换（数组的长度定义成5）。

函数原型：void exchange(int a[5],int b[5])

要求：

在主函数中输入两个数组中各元素的内容;

调用函数exchange实现两个数组中对应该元素值的交换;

在主函数中输出交换后两个数组的内容。

\*\*\*\*输入提示信息格式: 无

\*\*\*\*输入数据格式要求: "%d"

\*\*\*\*输出提示信息格式:"Output array a:"

"Output array b:"

\*\*\*\*输出数据格式要求:"%5d"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  void exchange(int a[5], int b[5]);    int main()  {      int a[5], b[5], i;      for (i = 0; i < 5; i++)  //1      {          scanf("%5d", &a[i]);      }      for (i = 0; i < 5; i++) //1      {          scanf("%5d", &b[i]);      }      exchange(a, b);//1      printf ("Output array a:");//1      for (i = 0; i < 5; i++)//1      {          printf("%5d", a[i]);//1        }      printf("\n");      printf ("Output array b:");//1      for (i = 0; i < 5; i++)      {          printf("%5d", b[i]);//1      }      return 0;  }  void exchange(int a[5], int b[5])  {      int i, temp;      for (i = 0; i < 5; i++)//1      {          temp = a[i];//1          a[i] = b[i];//1          b[i] = temp;//1      }  } |

Q593.(10分)编写程序，显示从3起小于100的所有双素数（双素数：如果p和q都是素数且q=p+2，则p/q称为双素数，例如3/5，5/7都是双素数）。

\*\*输出格式要求："%4d/%d"

程序运行示例如下：

3/5 5/7 11/13 17/19 29/31 41/43 59/61 71/73

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int prime(int a);  int main()  {      int i;      for (i = 3; i <= 100; i++)      {          if (prime(i) && prime(i + 2))              printf("%4d/%d", i, i + 2);      }      printf("\n");      return 0;  }    int prime(int a)  {      int b, k;      if (a == 2)return 1;      else if (a % 2 == 0)return 0;      else      {          b = 1;          k = 3;          while (k <= sqrt(a) && b)          {              if (a % k == 0)b = 0;              k = k + 2;          }          return b;      }  } |

Q493.(10分)输入6个字符串，输出最小串及最大串。

\*\*输入格式要求："%s" 提示信息："请输入6行字符串：\n"

\*\*输出格式要求："The max string is: %s\n" "The min string is: %s\n"

程序示例运行如下：

请输入6行字符串：

hello,world

vb

vc

Java

c++

c#

The max string is: vc

The min string is: Java

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>    #define N 6  #define N1 20    int main()  {      char str[N1], min[N1], max[N1];      int i;        printf("请输入6行字符串：\n");      gets(min);  //假设第1个串最小        strcpy(max, min);//假设最大串的值也为min        //循环输入其它串，并与最小的串及最大的串比较      for (i = 2; i <= N; i++)      {          gets(str);          if (strcmp(str, min) < 0)              strcpy(min, str);          if (strcmp(str, max) > 0)              strcpy(max, str);      }        printf("The max string is: %s\n", max);      printf("The min string is: %s\n", min);        return 0;  } |

Q241.(10分)在一种室内互动游戏中，魔术师要每位观众心里想一个三位数abc（a、b、c分别是百位、十位和个位数字），然后魔术师让观众心中记下acb、bac、bca、cab、cba五个数以及这5个数的和值。只要观众说出这个和是多少，则魔术师一定能猜出观众心里想的原数abc是多少。例如，观众甲说他计算的和值是1999，则魔术师立即说出他想的数是443，而观众乙说他计算的和值是1998，则魔术师说：“你算错了！”。请编程模拟这个数字魔术游戏。

\*\*输入格式要求："%d" 提示信息："Input a sum:"

\*\*输出格式要求："The sum you calculated is wrong!\n" "The number is %d\n"

程序运行示例如下：

Input a sum:1999

The number is 443

### 参考答案

|  |
| --- |
| #include <stdio.h>  int Magic(int m);  int main()  {      int m, ret;      printf("Input a sum:");      scanf("%d", &m);      ret = Magic(m);      if (ret != 1)      {          printf("The sum you calculated is wrong!\n");      }      return 0;  }  int Magic(int m)  {      int a, b, c, n;      for (a = 1; a < 9; a++)      {          for (b = 1; b < 9; b++)          {              for (c = 1; c < 9; c++)              {                  n = 122 \* a + 212 \* b + 221 \* c;                  if (m == n)                  {                      printf("The number is %d\n", 100 \* a + 10 \* b + c);                      return 1;                  }              }          }      }      return 0;  } |

Q251.(10分)冒泡排序（Bubble Sort），也称为沉降排序（Sinking Sort），之所以称其为冒泡排序，是因为算法中值相对较小的数据会像水中的气泡一样逐渐上升到数组的最顶端。与此同时，较大的数据逐渐地下沉到数组的底部。这个处理过程需在整个数组范围内反复执行多遍。每一遍执行时，比较相邻的两个元素。若顺序不对，则将其位置交换，当没有数据需要交换时，数据也就排好序了。编程将排序函数DataSort()改用冒泡法实现。

\*\*输入格式要求："%d" 提示信息："Input n:" "Input %d numbers:"

\*\*输出格式要求："Sorting results:" "%4d"

程序运行示例如下：

Input n:10

Input 10 numbers:10 2 9 3 4 0 6 8 7 5 1

Sorting results: 0 1 2 3 4 5 6 7 8 9

### 参考答案

|  |
| --- |
| #include<stdio.h>  #define  N  10  void  BubbleSort(int a[], int n);  int main()  {      int i, n, a[N];      printf("Input n:");      scanf("%d", &n);      printf("Input %d numbers:", n);      for (i = 0; i < n; i++)      {          scanf("%d", &a[i]);      }      BubbleSort(a, n);      printf("Sorting results:");      for (i = 0; i < n; i++)      {          printf("%4d", a[i]);      }      printf("\n");      return 0;  }  /\* 函数功能：冒泡法实现数组a 的n个元素的升序排序 \*/  void  BubbleSort(int a[], int n)  {      int  i, j, temp;      for (i = 0; i < n - 1; i++)      {          for (j = 1; j < n - i; j++)          {              if (a[j] < a[j - 1])              {                  temp = a[j];                  a[j] = a[j - 1];                  a[j - 1] = temp;              }          }      }  } |

Q1636.(10分)按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char dstStr[], char srcStr[]);

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 80  void MyStrcat(char dstStr[], char srcStr[]);  int main()  {      char  s[N], t[N];      printf("Input a string:");      gets(s);      printf("Input another string:");      gets(t);      MyStrcat(s, t);      printf("Concatenate results:%s\n", s);      return 0;  }  /\*  函数功能：将字符串srcStr连接到字符串dstStr之后 \*/  void MyStrcat(char dstStr[], char srcStr[])  {      unsigned int  i, j;      i = strlen(dstStr);               /\* 将下标移动到字符串dstStr的末尾 \*/      for (j = 0; j <= strlen(srcStr); j++, i++)      {          dstStr[i] = srcStr[j];      }  } |

Q1644.(10分)用动态数组编程输入任意m个班学生（每班n个学生）的某门课的成绩，计算最高分，并指出具有该最高分成绩的学生是第几个班的第几个学生。其中，m和n的值由用户从键盘任意输入（不限定m和n的上限值）。

输入提示信息：

"Input array size m,n:"

"Input %d\*%d array:\n"

输入格式："%d,%d"

输出提示信息和格式："maxScore = %d, class = %d, number = %d\n"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #include  <stdlib.h>  void InputScore(int \*p, int m, int n);  int  FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);  int main()  {      int  \*pScore, m, n, maxScore, row, col;      printf("Input array size m,n:");      scanf("%d,%d", &m, &n);      pScore = (int \*)calloc(m \* n, sizeof (int)); /\* 申请动态内存 \*/      if (pScore == NULL)      {          printf("No enough memory!\n");          exit(0);      }      InputScore(pScore, m, n);      maxScore = FindMax(pScore, m, n, &row, &col);      printf("maxScore = %d, class = %d, number = %d\n", maxScore, row + 1, col + 1);      free(pScore);                                      /\* 释放动态内存 \*/      return 0;  }  /\* 函数功能：输入m行n列二维数组的值 \*/  void InputScore(int \*p, int m, int n)  {      int i, j;      printf("Input %d\*%d array:\n", m, n);      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              scanf("%d", &p[i \* n + j]);          }      }  }  /\*  函数功能：计算任意m行n列二维数组中元素的最大值，并指出其所在行列下标值 \*/  int  FindMax(int \*p, int m, int n, int \*pRow, int \*pCol)  {      int  i, j, max = p[0];      \*pRow = 0;      \*pCol = 0;      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              if (p[i \* n + j] > max)              {                  max = p[i \* n + j];                  \*pRow = i;              /\*记录行下标\*/                  \*pCol = j;              /\*记录列下标\*/              }          }      }      return max;  } |

Q255.(10分)从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并返回它们所在数组中的位置。

\*\*输入格式要求："%d" 提示信息："Input n(n<=10):" "Input %d numbers:"

\*\*输出格式要求："%5d" "Exchange results:"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void  MaxMinExchang(int a[], int n);  void Swap(int \*x, int \*y);  int main()  {      int  a[10], n;      printf("Input n(n<=10):");      scanf("%d", &n);      printf("Input %d numbers:", n);      ReadData(a, n);      MaxMinExchang(a, n);      printf("Exchange results:");      PrintData(a, n);      return 0;  }  /\* 函数功能：输入数组a的n个元素值 \*/  void ReadData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          scanf("%d", &a[i]);      }  }  /\* 函数功能：输出数组a的n个元素值 \*/  void PrintData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          printf("%5d", a[i]);      }      printf("\n");  }  /\* 函数功能：将数组a中的最大数与最小数位置互换 \*/  void  MaxMinExchang(int a[], int n)  {      int  maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;      int  i;      for (i = 1; i < n; i++)      {          if (a[i] > maxValue)          {              maxValue = a[i];              maxPos = i;          }          if (a[i] < minValue)          {              minValue = a[i];              minPos = i;          }      }      Swap(&a[maxPos], &a[minPos]);  }  /\* 函数功能：两整数值互换 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  } |

Q1688.(10分)按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

输入某年某月某日，计算并输出它是这一年的第几天。

/\* 函数功能： 对给定的某年某月某日，计算它是这一年的第几天

函数参数： 整型变量year、month、day，分别代表年、月、日

函数返回值：这一年的第几天 \*/

int DayofYear(int year, int month, int day);

输入提示信息："Please enter year, month, day:"

输入格式："%d,%d,%d"

输出提示信息和格式："yearDay = %d\n"

### 参考答案

|  |
| --- |
| #include    <stdio.h>  int  DayofYear(int year, int month, int day);  int dayTab[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},                           {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};  int main()  {      int year, month, day, yearDay;      printf("Please enter year, month, day:");      scanf("%d,%d,%d", &year, &month, &day);      yearDay = DayofYear(year, month, day);      printf("yearDay = %d\n", yearDay);      return 0;  }  /\* 函数功能：对给定的某年某月某日，计算并返回它是这一年的第几天 \*/  int  DayofYear(int year, int month, int day)  {      int  i, leap;      /\* 若year为闰年，即leap值为1，则用第1行元素dayTab[1][i]计算；      否则leap值为0，用第0行dayTab[0][i]计算 \*/      leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);      for (i = 1; i < month; i++)      {          day = day + dayTab[leap][i];      }      return day;         /\* 返回计算出的day的值 \*/  } |

Q1690.(10分)按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

/\* 函数功能： 对给定的某年某月某日，计算它是这一年的第几天

函数参数： 整型变量year、month、day，分别代表年、月、日

函数返回值：这一年的第几天 \*/

int DayofYear(int year, int month, int day);

/\* 函数功能： 对给定的某一年的第几天，计算它是这一年的第几月第几日

函数入口参数：整型变量year，存储年

整型变量yearDay，存储这一年的第几天

函数出口参数：整型指针pMonth，指向存储这一年第几月的整型变量

整型指针pDay，指向存储第几日的整型变量

函数返回值： 无 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

输出如下菜单，用switch语句实现根据用户输入的选择执行相应的操作。

1. year/month/day -> yearDay

2. yearDay -> year/month/day

3. Exit

Please enter your choice:

输入提示信息：

"Please enter year, month, day:"

"Please enter year, yearDay:"

输入格式：

"%d,%d,%d"

"%d,%d"

输出提示信息和格式：

"yearDay = %d\n"

"month = %d, day = %d\n"

### 参考答案

|  |
| --- |
| #include    <stdio.h>  #include    <stdlib.h>  int  DayofYear(int year, int month, int day);  void  MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);  void  Menu(void);  int dayTab[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},                           {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};  int main()  {      int    year, month, day, yearDay;      char   c;      Menu();             /\* 调用Menu函数显示一个固定式菜单 \*/      c = getchar();      /\* 输入选择 \*/      switch (c)          /\* 判断选择的是何种操作 \*/      {      case '1':          printf("Please enter year, month, day:");          scanf("%d,%d,%d", &year, &month, &day);          yearDay = DayofYear(year, month, day);          printf("yearDay = %d\n", yearDay);          break;      case '2':          printf("Please enter year, yearDay:");          scanf("%d,%d", &year, &yearDay);          MonthDay(year, yearDay, &month, &day);          printf("month = %d,day = %d\n", month, day);          break;      case '3':          exit(0);   /\* 退出程序的运行 \*/      default:          printf("Input error!");      }      return 0;  }  /\* 函数功能：对给定的某年某月某日，计算并返回它是这一年的第几天 \*/  int  DayofYear(int year, int month, int day)  {      int  i, leap;      /\* 若year为闰年，即leap值为1，则用第1行元素dayTab[1][i]计算；      否则leap值为0，用第0行dayTab[0][i]计算 \*/      leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);      for (i = 1; i < month; i++)      {          day = day + dayTab[leap][i];      }      return day;         /\* 返回计算出的day的值 \*/  }  /\* 函数功能：对给定的某一年的第几天，计算它是这一年的第几月第几日 \*/  void  MonthDay(int year, int yearDay, int \*pMonth, int \*pDay)  {      int  i, leap;      leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);      for (i = 1; yearDay > dayTab[leap][i]; i++)      {          yearDay = yearDay - dayTab[leap][i];      }      \*pMonth = i;        /\* 将计算出的月份值赋值给pMonth所指向的变量 \*/      \*pDay = yearDay;   /\* 将计算出的日号赋值给pDay所指向的变量 \*/  }  /\* 函数功能：显示菜单 \*/  void  Menu(void)  {      printf("1. year/month/day -> yearDay\n");      printf("2. yearDay -> year/month/day\n");      printf("3. Exit\n");      printf("Please enter your choice:");  } |

Q1615.(10分)按如下函数原型编写程序，用函数编程实现两个数组中对应元素值的交换。

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void Swap(int \*x, int \*y);

输入提示信息：

"Input array size(n<=10):"

"Input array a:"

"Input array b:"

输入格式："%d"

输出提示信息：

"Output array a:"

"Output array b:"

输出格式："%5d"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #define N 10  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void Swap(int \*x, int \*y);  int main()  {      int  a[N], b[N], i, n;      printf("Input array size(n<=10):");      scanf("%d", &n);      printf("Input array a:");      ReadData(a, n);      printf("Input array b:");      ReadData(b, n);      for (i = 0; i < n; i++)      {          Swap(&a[i], &b[i]);      }      printf("Output array a:");      PrintData(a, n);      printf("Output array b:");      PrintData(b, n);      return 0;  }  /\* 函数功能：输入数组a的n个元素值 \*/  void ReadData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          scanf("%d", &a[i]);      }  }  /\* 函数功能：输出数组a的n个元素值 \*/  void PrintData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          printf("%5d", a[i]);      }      printf("\n");  }  /\* 函数功能：两整数值互换 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  } |

Q1616.(50分)按如下函数原型编写程序，从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

void Swap(int \*x, int \*y);

输入提示信息：

"Input n(n<=10):"

"Input %d numbers:"

输入格式："%d"

输出提示信息：

"Exchange results:"

输出格式："%5d"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void  MaxMinExchang(int a[], int n);  void Swap(int \*x, int \*y);  int main()  {      int  a[10], n;      printf("Input n(n<=10):");      scanf("%d", &n);      printf("Input %d numbers:", n);      ReadData(a, n);      MaxMinExchang(a, n);      printf("Exchange results:");      PrintData(a, n);      return 0;  }  /\* 函数功能：输入数组a的n个元素值 \*/  void ReadData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          scanf("%d", &a[i]);      }  }  /\* 函数功能：输出数组a的n个元素值 \*/  void PrintData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          printf("%5d", a[i]);      }      printf("\n");  }  /\* 函数功能：将数组a中的最大数与最小数位置互换 \*/  void  MaxMinExchang(int a[], int n)  {      int  maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;      int  i;      for (i = 1; i < n; i++)      {          if (a[i] > maxValue)          {              maxValue = a[i];              maxPos = i;          }          if (a[i] < minValue)          {              minValue = a[i];              minPos = i;          }      }      Swap(&a[maxPos], &a[minPos]);  }  /\* 函数功能：两整数值互换 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  } |

Q1686.(10分)用字符指针作函数参数编程实现在字符串每个字符间插入一个空格的功能。

要求：字符串的输入和输出在主函数中完成。

\*\*输入提示信息："Input a string:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Insert results:%s\n"

程序运行示例1如下：

Input a string:Howareyou

Insert results:H o w a r e y o u

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #include <stdlib.h>  #define N 100  void Insert(char \*s);  int main()  {      char str[N];      printf("Input a string:");      gets(str);      Insert(str);      printf("Insert results:%s\n", str);      return 0;  }  void Insert(char \*s)  {      char str[N];      char \*t = str;      strcpy(t, s);      for (; \*t != '\0'; s++, t++)      {          \*s = \*t;          s++;          \*s = ' ';      }      \*s = '\0';      /\* 在字符串s的末尾添加字符串结束标志 \*/  } |

Q537.(10分)编程实现显示用户输入的月份（不包括闰年的月份）拥有的天数。

\*\*输入格式要求："%d" 提示信息："Input a month:" "The number of days is %d\n"

\*\*输出格式要求："%d"

程序的运行示例如下：

Input a month:13

Input a month:2

The number of days is 28

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  MONTHS 12  int main()  {      int days[MONTHS] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};      int month;      do      {          printf("Input a month:");          scanf("%d", &month);      }      while (month < 1 || month > 12); /\* 处理不合法数据的输入 \*/      printf("The number of days is %d\n", days[month - 1]);      return 0;  } |

Q243.(10分)输入某班学生某门课的成绩（最多不超过40人，具体人数由用户键盘输入），用函数编程统计不及格人数。

\*\*输入格式要求："%d" 提示信息："How many students?"

\*\*输出格式要求："Fail students = %d\n"

程序运行示例如下：

How many students?5

45 56 78 99 100

Fail students = 2

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #define N 40  int GetFailNum(int score[], int n);  int main()  {      int i, n, fail, score[N];      printf("How many students?");      scanf("%d", &n);      for (i = 0; i < n; i++)      {          scanf("%d", &score[i]);      }      fail = GetFailNum(score, n);      printf("Fail students = %d\n", fail);      return 0;  }  /\* 函数功能：统计不及格人数 \*/  int GetFailNum(int score[], int n)  {      int  i, count = 0;      for (i = 0; i < n; i++)      {          if (score[i] < 60)  count++;      }      return count;  } |

Q187.(15分)输入某班学生某门课的成绩（最多不超过40人），当输入为负值时，表示输入结束，

用函数编程统计成绩高于平均分的学生人数。

要求按如下原型编写统计成绩在全班平均分及平均分之上的学生人数

int GetAboveAver(int score[], int n);

\*\*输入要求：无输入提示

\*\*输出要求：

"Total students are %d\n"

"Average score is %d\n"

"Students of above average is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define N 40  int Average(int score[], int n);  int ReadScore(int score[]);  int GetAboveAver(int score[], int n);  main() {      int score[N], m, n;      n = ReadScore(score);//1      printf("Total students are %d\n", n);//1      m = GetAboveAver(score, n);//1      printf("Students of above average is %d\n", m);//1  }  int Average(int score[], int n) {      int i, sum = 0;      for (i = 0; i < n; i++)//1      {          sum += score[i];//1      }      return sum / n;//1  }  int ReadScore(int score[]) {      int i = -1;      do {          i++;          scanf("%d", &score[i]);//1      } while (score[i] >= 0);//e2      return i;//1  }  int GetAboveAver(int score[], int n) {      int i, count = 0, aver;      aver = Average(score, n);//1      printf("Average score is %d\n", aver);//1      for (i = 0; i < n; i++)//1      {          if (score[i] >= aver) //1              count++;//1      }      return count;//1  } |

Q541.(10分)从键盘输入某班学生某门课的成绩（每班人数最多不超过40人），当输入为负值时，表示输入结束，试编程计算并打印最高分。

\*\*输入格式要求："%d" 提示信息："Total students are %d\n" "The highest score is %d\n" "Input score:"

\*\*输出格式要求："%d"

程序的运行示例如下：

Input score:80

Input score:100

Input score:90

Input score:60

Input score:-1

Total students is 4

The highest score are 100

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define N 40  int ReadScore(int score[]);       /\* ReadScore()函数原型 \*/  int FindMax(int score[], int n);  /\* FindMax()函数原型 \*/  int main()  {      int score[N], max, n;      n = ReadScore(score);      /\*调用函数ReadScore()输入成绩，返回学生人数\*/      printf("Total students are %d\n", n);      max = FindMax(score, n);   /\*调用函数FindMax()计算最高分，返回最高分\*/      printf("The highest score is %d\n", max);      return 0;  }  /\* 函数功能：输入学生某门课的成绩，当输入负值时，结束输入，返回学生人数 \*/  int ReadScore(int score[])     /\* ReadScore()函数定义 \*/  {      int i = -1;         /\*i初始化为-1，循环体内增1后可保证数组下标从0开始\*/      do      {          i++;          printf("Input score:");          scanf("%d", &score[i]);      }      while (score[i] >= 0);      /\* 输入负值时结束成绩输入 \*/      return i;                      /\* 返回学生人数 \*/  }  /\* 函数功能：计算最高分 \*/  int FindMax(int score[], int n)/\* FindMax()函数定义 \*/  {      int max, i;      max = score[0];            /\* 假设score[0]值为当前最大值 \*/      for (i = 1; i < n; i++)      {          if (score[i] > max)      /\* 若score[i]值较大 \*/          {              max = score[i];      /\* 则用score[i]值替换当前最大值 \*/          }      }      return max;                    /\* 返回最高分 \*/  } |

Q1335.(10分)输入一行字符，分别统计其中每个元音字母（a、e、i、o、u不分大小写）的个数。

\*\*输入格式要求：信息提示："Input a line of characters:\n"

\*\*输出格式要求："4d"

程序的运行示例如下：

Input a line of characters:

How old are you?

1 1 0 3 1

### 参考答案

|  |
| --- |
| #include<stdio.h>  int main()  {      char ch[80], \*p;      int i, count[5] = {0};      printf("Input a line of characters:\n");      gets(ch);      for (p = ch; \*p != '\0'; p++)      {          switch (\*p)          {          case 'a':              count[0]++;              break;          case 'e':              count[1]++;              break;          case 'i':              count[2]++;              break;          case 'o':              count[3]++;              break;          case 'u':              count[4]++;          }      }      for (i = 0; i < 5; i++)      {          printf("%4d", count[i]);      }      return 0;  } |

Q294.(5分)从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str)==0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

### 参考答案

|  |
| --- |
| #include "stdio.h"  #include "string.h"  main()  {      int i=0, findFlag=1;      char x[13];      char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };        printf("Input string:\n");      gets(x);        while( i<6 && findFlag)      {          if(strcmp(x,str[i])==0)          {              findFlag = 0;          }          i++;      }      if ( !findFlag )      {          printf("%s\n",x);      }      else      {          printf("Not find!\n");      }  } |

Q403.(10分)冒泡排序法。设有N个杂乱无序的整数，要求将这N个整数从小到大排序后输出。

\*\*输入格式要求："%d" 提示信息："Enter No.%2d:"

\*\*输出格式要求："%d"

程序运行示例如下：

Enter No. 1:5

Enter No. 2:7

Enter No. 3:3

Enter No. 4:9

Enter No. 5:8

35789

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define N 5  main()  {int i,j,t,a[N];   for(i=0;i<N;i++)   { printf("Enter No.%2d:",i+1);     scanf("%d",&a[i]);   }   for(i=0;i<N-1;i++)     for(j=0;j<N-i-1;j++)       if(a[j]>a[j+1])       { t=a[j];         a[j]=a[j+1];         a[j+1]=t;       }   for(i=0;i<N;i++)     printf("%d",a[i]);  } |

小明假期同爸爸一起去书店，他选中了六本书，每本书的单价分别：3.1，1.7，2，5.3，0.9和7.2元。不巧的是，爸爸兜里只带了十几块钱，为了让小明过一个愉快的假期，爸爸仍然同意买书，但提出了一个要求，要小明从六本书中选出若干本，使得单价相加所得的和同10最接近。你能够帮助小明解决这个问题吗？

\*\*输出格式要求：" 10(+ -)%.2f=" "%.2f+"

程序运行示例如下：

3.1 1.7 2 5.3 0.9 7.2

10(+ -)0.10=2.00+0.90+7.20

10(+ -)0.10=1.70+2.00+5.30+0.90

10(+ -)0.10=3.10+1.70+5.30

|  |
| --- |
| #include <math.h>  main()  {    int d[6],m,i,j;    long b[63],flag;    float c[6]={3.1,1.7,2,5.3,0.9,7.2},min,x;    /\*printf("Please enter the prices of 6 books:");    for(i=0;i<6;i++)      scanf("%f",&c[i]);\*/    for(i=0,min=-1,d[0]=0;d[0]<2;d[0]++)      for(d[1]=0;d[1]<2;d[1]++)        for(d[2]=0;d[2]<2;d[2]++)          for(d[3]=0;d[3]<2;d[3]++)            for(d[4]=0;d[4]<2;d[4]++)             for(d[5]=0;d[5]<2;d[5]++)             {               for(flag=0,x=0.,j=5;j>=0;j--)               {                 x+=c[j]\*d[j];flag=flag\*10+d[j];               }               x=((x-10>0)?x-10:10-x);               if(min<0)               {                 min=x;                 b[i++]=flag;               }               else if(min-x>1.e-6)               {                   min=x;b[0]=flag;i=1;               }               else if(fabs((double)x-min)<1.e-6)                   b[i++]=flag;             }      for(m=0;m<i;m++)      {        printf("  10(+ -)%.2f=",min);        for(flag=b[m],j=0;flag>0;j++,flag/=10)              if(flag%10)                  if(flag>1) printf("%.2f+",c[j]);                  else     printf("%.2f\n",c[j]);      }  } |

Q1343.(10分)魔术师利用一副牌中的13张红桃，预先将它们排好后迭在一起，牌面朝下。对观众说：“我不看牌”，只数数就可以猜到每张牌是什么，我大声数，你们听，不信？你们就看。魔术师将最上面的那张牌数为1，把它翻过来正好是红桃A，将红桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌的下面，将第二张牌翻过来，正好是红桃2，也将它放在桌子上。第三次数1、2、3，将前面两张依次放在这迭牌的下面，再翻第三张牌正好是红桃3.这样依次将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

输入格式要求：无

输出格式要求："%d "（%d后面有一个空格），全部数据输出后再输出一个换行"\n"

程序运行示例：

1 8 2 5 10 3 12 11 9 4 7 6 13

|  |
| --- |
| #include <stdio.h>    int main()  {      int cards[14] = {0};      int i, j = 1, n;      for (i = 1 ; i <= 13 ; i++)      {          n = 1;          do          {              if (j > 13)                  j = 1;              if (cards[j] > 0)                  j++;              else              {                  if (n == i)                      cards[j] = i;                  j++;                  n++;              }          }          while (n <= i);      }      for (i = 1; i <= 13; i++)          printf("%d ", cards[i]);      printf("\n");        return 0;  } |

Q361.(10分)编写一个C函数，实现保留小数点后第N-1位，从第N位四舍五入。用该函数对数组a[5]中的各元素从小数点后第2位开始四舍五入。

a[5]={2.33,2.56,2.65,2.66,2.30}

\*\*输出格式要求："Old array:\n" "a[%d]=%.3f\t" "\nnew array:\n" "a[%d]=%.2f\t"

程序运行示例如下：

Old array:

a[0]=2.330 a[1]=2.560 a[2]=2.650 a[3]=2.660 a[4]=2.300

new array:

a[0]=2.30 a[1]=2.60 a[2]=2.70 a[3]=2.70 a[4]=2.30

|  |
| --- |
| #include <stdio.h>  #include <math.h>  float a[5] = {2.33, 2.56, 2.65, 2.66, 2.30};  float t(float x,int n)  {      float x1, x2;      int i;      for (x1 = x, i = 0; i < n - 1; i++)          x1 \*= 10.0;      x2 = floor((double)x1);      x1 = floor((double)(x1 - x2) \* 10);      if (x1 >= 5)          x2 += 1;      for (i = 0; i < n - 1; i++)          x2 /= 10.0;      return x2;  }  main()  {      float t();      int i;      printf("Old array:\n");      for (i = 0; i < 5; i++)          printf("a[%d]=%.3f\t", i, a[i]);      printf("\nnew array:\n");      for (i = 0; i < 5; i++)          printf("a[%d]=%.2f\t", i, t(a[i], 2));  } |

Q285.(15分)从键盘输入10个整型数据，放入数组a中，求其最大值，并输出。

\*\*要求输入提示信息为：无输入提示信息

\*\*要求输入格式为："%d"

\*\*要求输出格式为："max=%d\n"

|  |
| --- |
| #include <stdio.h>  main()  {      int a[10],i,max;      for(i=0;i<10;i++)      {          scanf("%d",&a[i]);  //2      }      max=a[0];  //2      for(i=1;i<10;i++)  //2      {          if(max<a[i])  //2          {              max=a[i];  //2          }      }      printf("max=%d\n",max);  //2  } |

#include<stdio.h>

int main()

{

    int cards[14] = {0};

    int i, j = 1, n;

    for (i = 1 ; i <= 13 ; i++)

    {

        n = 1;

        do

        {

            if (j > 13)

                j = 1;

            if (cards[j] > 0)

                j++;

            else

            {

                if (n == i)

                    cards[j] = i;

                j++;

                n++;

            }

        }

        while (n <= i);

    }

    for (i = 1; i <= 13; i++)

        printf("%d ", cards[i]);

    printf("\n");

    return 0;

}

Q1335.(10分)输入一行字符，分别统计其中每个元音字母（a、e、i、o、u不分大小写）的个数。

\*\*输入格式要求：信息提示："Input a line of characters:\n"

\*\*输出格式要求："4d"

程序的运行示例如下：

Input a line of characters:

How old are you?

1 1 0 3 1

|  |
| --- |
| #include<stdio.h>  int main()  {      char ch[80], \*p;      int i, count[5] = {0};      printf("Input a line of characters:\n");      gets(ch);      for (p = ch; \*p != '\0'; p++)      {          switch (\*p)          {          case 'a':              count[0]++;              break;          case 'e':              count[1]++;              break;          case 'i':              count[2]++;              break;          case 'o':              count[3]++;              break;          case 'u':              count[4]++;          }      }      for (i = 0; i < 5; i++)      {          printf("%4d", count[i]);      }      return 0;  } |

Q294.(5分)从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str)==0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

|  |
| --- |
| #include "stdio.h"  #include "string.h"  main()  {      int i=0, findFlag=1;      char x[13];      char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };        printf("Input string:\n");      gets(x);        while( i<6 && findFlag)      {          if(strcmp(x,str[i])==0)          {              findFlag = 0;          }          i++;      }      if ( !findFlag )      {          printf("%s\n",x);      }      else      {          printf("Not find!\n");      }  } |

Q1283.(10分)

|  |
| --- |
| 程序改错。以下程序用于打印输出结构体数组的所有元素。  struct s  {      char name[10];      int age;  }  main()  {      struct s a[3] = {”John”,19,”Paul”,17,”Marry”,18};      int \*p;        for(p = a; p < 3; p++)      {          printf("%s,%d\n", p->name, p->age );      }    } |
| struct s  {      char name[10];      int age;  };  main()  {      struct s a[3] = {"John",19,"Paul",17,"Marry",18};      struct s \*p;        for(p = a; p < a+3; p++)      {          printf("%s,%d\n", p->name, p->age );      }    } |

Q1315.(10分)有一电文，已按下面规律译成密码。

A->Z a->z

B->Y b->y

C->X c->x

… …

即第一个字母变成第26个字母，第i个字母变成第（26-i+1）个字母，非字母字符不变。要求编程将密码译回原文，并打印出密码和原文。

程序的运行示例如下：

请输入字符：hello,world!

密码是：hello,world!

原文是：svool,dliow!

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, n;      char ch[80], tran[80];        printf("请输入字符：");      gets(ch);      printf("\n密码是：%s", ch);        i = 0;      while (ch[i] != '\0')      {          if ((ch[i] >= 'A') && (ch[i] <= 'Z'))          {              tran[i] = 26 + 64 - ch[i] + 1 + 64;          }          else if ((ch[i] >= 'a') && (ch[i] <= 'z'))          {              tran[i] = 26 + 96 - ch[i] + 1 + 96;          }          else          {              tran[i] = ch[i];          }          i++;      }      n = i;      printf("\n原文是：");      for (i = 0; i < n; i++)      {          putchar(tran[i]);      }        return 0;  } |

Q404.(10分)选择排序法。用选择排序法将N个数从小到大排序后输出。

\*\*输入格式要求："%d" 提示信息："Enter No.%2d:"

\*\*输出格式要求："%d"

程序运行示例如下：

Enter No. 1:5

Enter No. 2:7

Enter No. 3:3

Enter No. 4:9

Enter No. 5:8

35789

|  |
| --- |
| #include <stdio.h>  #define N 5  main()  { int i,j,t,p,a[N];    for(i=0;i<N;i++)   { printf("Enter No.%2d:",i+1);     scanf("%d",&a[i]);   }   for(i=0;i<N-1;i++)   { p=i;     for(j=i+1;j<N;j++)       if(a[p]>a[j])       p=j;       if(p!=i)       {t=a[p];a[p]=a[i];a[i]=t;       }    }    for(i=0;i<N;i++)        printf("%d",a[i]);  } |

Q444.(10分)选美比赛

在选美大奖赛的半决赛现场，有一批选手参加比赛，比赛的规则是最后得分越高，名次

越低。当半决赛结束时，要在现场按照选手的出场顺序宣布最后得分和最后名次，获得

相同分数的选手具有相同的名次，名次连续编号，不用考虑同名次的选手人数。例如：

选手序号： 1，2，3，4，5，6，7

选手得分： 5，3，4，7，3，5，6

则输出名次为：3，1，2，5，1，3，4

请编程帮助大奖赛组委会完成半决赛的评分排名工作。

\*\*输出格式要求：" Player\_NO Score Rank\n" " %3d %4d %4d\n"

程序运行示例如下：

Player\_NO Score Rank

1 5 3

2 3 1

3 4 2

4 7 5

5 3 1

6 5 3

7 6 4

|  |
| --- |
| #include <stdio.h>  #define NUM 7  int a[NUM+1]={0,5,3,4,7,3,5,6};  int m[NUM+1],l[NUM+1];  main()  {    int i,n,smallest,num,k,j;    num=1;    for(i=1;i<=NUM;i++)      if(m[i]==0)      {          smallest=a[i];          k=1;          l[k]=i;          for(j=i+1;j<=NUM;j++)            if(m[j]==0)               if(a[j]<smallest)               {                  smallest=a[j];                  k=0;                  l[++k]=j;               }               else if(a[j]==smallest)                      l[++k]=j;           for(j=1;j<=k;j++)             m[l[j]]=num;             num++;             i=0;      }      printf("  Player\_NO  Score  Rank\n");      for(j=1;j<=NUM;j++)         printf("   %3d    %4d   %4d\n",j,a[j],m[j]);  } |

Q596.(10分)编写程序：用整数型数组a存放一个8位长整型数（每个数组元素为一个一位整数），判断该数是否为对称数。该数由键盘输入。例如：132231为对称数。

\*\*输入格式要求："%d"

\*\*输出格式要求："no,it is not\n" "yes,it is\n"

程序运行示例1如下：

1

2

3

4

4

3

2

1

输出

yes,it is

程序运行2示例如下：

2

3

4

4

3

2

1

0

输出

no,it is not

|  |
| --- |
| #include <stdio.h>  main()  {      int i, j, a[8];        for (i = 0; i < 8; i++)          scanf("%d", &a[i]);        for (i = 0, j = 7; i <= j; i++, j--)          if (a[i] != a[j])  break;        if (i < j)  printf("no,it is not\n");      else  printf("yes,it is\n");  } |

Q619.(10分)某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下： 每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

\*\*输入格式要求："%d"

\*\*输出格式要求："%d"

程序运行示例如下：

输入:3421

输出:8976

|  |
| --- |
| int main()  {      int i, aa[4], t;      for (i = 0; i <= 3; i++)      {          scanf("%d", &aa[i]);      }      for (i = 0; i <= 3; i++)      {          aa[i] += 5;          aa[i] %= 10;      }      for (i = 0; i <= 3 / 2; i++)      {          t = aa[i];          aa[i] = aa[3 - i];          aa[3 - i] = t;      }      for (i = 3; i >= 0; i--)          printf("%d", aa[i]);  } |

Q253.(10分)大奖赛现场统分。已知某大奖赛有n个选手参赛，m（m>2）个评委为参赛选手评分（最高10分，最低0分）。统分规则为：在每个选手的m个得分中，去掉一个最高分和一个最低分后，取平均分作为该选手的最后得分。要求编程实现：

（1）根据n个选手的最后得分，从高到低输出选手的得分名次表，以确定获奖名单；

（2）根据各选手的最后得分与各评委给该选手所评分数的差距，对每个评委评分的准确性和评分水准给出一个定量的评价，从高到低输出各评委得分的名次表。

程序运行示例如下：

How many Athletes?5

How many judges?5

Scores of Athletes:

Athlete 1 is playing.

Please enter his number code:11

Judge 1 gives score:9.5

Judge 2 gives score:9.6

Judge 3 gives score:9.7

Judge 4 gives score:9.4

Judge 5 gives score:9.0

Delete a maximum score:9.7

Delete a minimum score:9.0

The final score of Athlete 11 is 9.500

Athlete 2 is playing.

Please enter his number code:12

Judge 1 gives score:9.0

Judge 2 gives score:9.2

Judge 3 gives score:9.1

Judge 4 gives score:9.3

Judge 5 gives score:8.9

Delete a maximum score:9.3

Delete a minimum score:8.9

The final score of Athlete 12 is 9.100

Athlete 3 is playing.

Please enter his number code:13

Judge 1 gives score:9.6

Judge 2 gives score:9.7

Judge 3 gives score:9.5

Judge 4 gives score:9.8

Judge 5 gives score:9.4

Delete a maximum score:9.8

Delete a minimum score:9.4

The final score of Athlete 13 is 9.600

Athlete 4 is playing.

Please enter his number code:14

Judge 1 gives score:8.9

Judge 2 gives score:8.8

Judge 3 gives score:8.7

Judge 4 gives score:9.0

Judge 5 gives score:8.6

Delete a maximum score:9.0

Delete a minimum score:8.6

The final score of Athlete 14 is 8.800

Athlete 5 is playing.

Please enter his number code:15

Judge 1 gives score:9.0

Judge 2 gives score:9.1

Judge 3 gives score:8.8

Judge 4 gives score:8.9

Judge 5 gives score:9.2

Delete a maximum score:9.2

Delete a minimum score:8.8

The final score of Athlete 15 is 9.000

Order of Athletes:

order final score number code

1 9.600 13

2 9.500 11

3 9.100 12

4 9.000 15

5 8.800 14

Order of judges:

order final score number code

1 9.937 1

2 9.911 2

3 9.859 3

4 9.833 4

5 9.714 5

Over!Thank you!

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #include  <math.h>  #define ATHLETE 40                              /\* 选手人数最高限 \*/  #define JUDGE        20                             /\* 评委人数最高限 \*/  void  CountAthleteScore(int sh[], float sf[], int n, float f[], int m);  void  Sort(int h[], float f[], int n);  void  Print(int h[], float f[], int n);  void  CountJudgeScore(int ph[], float pf[], int m, float sf[], float f[],                        int n);  int main()  {      int  j, m, n;      int  sh[ATHLETE];                           /\* 选手的编号 \*/      int  ph[JUDGE];                             /\* 评委的编号 \*/      float  sf[ATHLETE];                         /\* 选手的最后得分 \*/      float  pf[JUDGE];                           /\* 评委的得分 \*/      float  f[ATHLETE][JUDGE];                   /\* 评委给选手的评分 \*/      printf("How many Athletes?");      scanf("%d", &n);                                /\* 输入选手人数 \*/      printf("How many judges?");      scanf("%d", &m);                                /\* 输入评委人数 \*/      for (j = 1; j <= m; j++)      {          ph[j] = j;      }      printf("Scores of Athletes:\n");      CountAthleteScore(sh, sf, n, \*f, m);        /\* 现场为选手统计分数 \*/      CountJudgeScore(ph, pf, m, sf, \*f, n);      /\* 为各个评委打分 \*/      printf("Order of Athletes:\n");      Sort(sh, sf, n);                                /\* 选手得分排序 \*/      Print(sh, sf, n);                               /\* 打印选手名次表 \*/        printf("Order of judges:\n");      Sort(ph, pf, m);                                /\* 评委得分排序 \*/      Print(ph, pf, m);                               /\* 打印评委名次表 \*/      printf("Over!Thank you!\n");      return 0;  }  /\* 函数功能：统计参赛选手的得分 \*/  void  CountAthleteScore(int sh[], float sf[], int n, float f[], int m)  {      int  i, j;      float  max, min;      for (i = 1; i <= n; i++)                     /\* 第i个选手 \*/      {          printf("\nAthlete %d is playing.", i);          printf("\nPlease enter his number code:");          scanf("%d", &sh[i]);          sf[i] = 0;          max = 0;                                /\* 最高分初值设为最小值 \*/          min = 100;                              /\* 最低分初值设为最大值 \*/          for (j = 1; j <= m; j++)              /\* 第j个评委 \*/          {              printf("Judge %d gives score:", j);              scanf("%f", &f[i \* m + j]);              sf[i] = sf[i] + f[i \* m + j];   /\* 累加评委对第i个选手的评分 \*/              if (max < f[i \* m + j])          /\* 找出最高分 \*/              {                  max = f[i \* m + j];              }              if (min > f[i \* m + j])          /\* 找出最低分 \*/              {                  min = f[i \* m + j];              }          }          printf("Delete a maximum score:%.1f\n", max);          printf("Delete a minimum score:%.1f\n", min);          sf[i] = (sf[i] - max - min) / (m - 2); /\*去掉一个最高分和最低分\*/          printf("The final score of Athlete %d is %.3f\n", sh[i], sf[i]);      }  }  /\* 函数功能：对分数从高到低排序 \*/  void Sort(int h[], float f[], int n)  {      int  i, j, k, temp2;      float  temp1;      for (i = 1; i <= n - 1; i++)      {          k = i;          for (j = i + 1; j <= n; j++)          {              if (f[j] > f[k])  k = j;          }          if (k != i)          {              temp1 = f[k];              f[k] = f[i];              f[i] = temp1;              temp2 = h[k];              h[k] = h[i];              h[i] = temp2;          }      }  }  /\* 函数功能：打印名次表 \*/  void Print(int h[], float f[], int n)  {      int  i;      printf("order\tfinal score\tnumber code\n");      for (i = 1; i <= n; i++)      {          printf("%5d\t%11.3f\t%6d\n", i, f[i], h[i]);      }  }  /\* 函数功能：统计评委的得分 \*/  void CountJudgeScore(int ph[], float pf[], int m, float sf[], float f[],                       int n)  {      int  i, j;      for (j = 1; j <= m; j++)         /\* 第j个评委 \*/      {          pf[j] = 0;          for (i = 1; i <= n; i++)    /\* 第i个选手 \*/          {              pf[j] = pf[j] + (f[i \* m + j] - sf[i]) \* (f[i \* m + j] - sf[i]);          }          pf[j] = 10 - sqrt(pf[j] / n);      }  } |

Q1604.(10分)按如下函数原型编写程序，用一个整型数组feedback保存调查的40个反馈意见。用函数编程计算反馈意见的平均数（Mean）、中位数（Median）和众数（Mode）。中位数指的是排列在数组中间的数。如果原始数据的个数是偶数，那么中位数等于中间那两个元素的算术平均值。众数是数组中出现次数最多的那个数（不考虑两个或两个以上的反馈意见出现次数相同的情况）。

int Mean(int answer[], int n);

int Median(int answer[], int n);

int Mode(int answer[], int n);

void DataSort(int a[], int n);

输入提示信息："Input the feedbacks of 40 students:\n"

输入格式："%d"

输出提示信息和输出格式：

"Mean value = %d\n"

"Median value = %d\n"

"Mode value = %d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  M   40  #define  N   11  int Mean(int answer[], int n);  int Median(int answer[], int n);  int Mode(int answer[], int n);  void DataSort(int a[], int n);  int main()  {      int  i, feedback[M];      printf("Input the feedbacks of 40 students:\n");      for (i = 0; i < M; i++)      {          scanf("%d", &feedback[i]);      }      printf("Mean value = %d\n", Mean(feedback, M));      printf("Median value = %d\n", Median(feedback, M));      printf("Mode value = %d\n", Mode(feedback, M));      return 0;  }  /\* 函数功能：若n>0则计算并返回n个数的平均数，否则返回-1 \*/  int Mean(int answer[], int n)  {      int i, sum = 0;      for (i = 0; i < n; i++)      {          sum += answer[i];      }      return  n > 0 ? sum / n : -1;  }  /\* 函数功能：计算n个数的中位数 \*/  int Median(int answer[], int n)  {      DataSort(answer, n);      if (n % 2 == 0)          return  (answer[n / 2] + answer[n / 2 - 1]) / 2;      else          return  answer[n / 2];  }  /\* 函数功能：计算n个数的众数 \*/  int Mode(int answer[], int n)  {      int  i, grade, max = 0, modeValue = 0, count[N] = {0};      for (i = 0; i < n; i++)      {          count[answer[i]]++;      }      for (grade = 1; grade <= N - 1; grade++)      {          if (count[grade] > max)          {              max = count[grade];              modeValue = grade;          }      }      return modeValue;  }  /\*  函数功能：按选择法对数组a中的n个元素进行排序 \*/  void DataSort(int a[], int n)  {      int i, j, k, temp;      for (i = 0; i < n - 1; i++)      {          k = i;          for (j = i + 1; j < n; j++)          {              if (a[j] > a[k]) k = j;          }          if (k != i)          {              temp = a[k];              a[k] = a[i];              a[i] = temp;          }      }  } |

Q538.(10分)从键盘输入某年某月（包括闰年），编程输出该年的该月拥有的天数。

\*\*输入格式要求："%d,%d" 提示信息："Input year,month:" "The number of days is %d\n"

\*\*输出格式要求："%d"

程序运行示例1如下：

Input year,month:1984,2

The number of days is 29

程序运行示例2如下：

Input year,month:2000,2

The number of days is 29

程序运行示例3如下：

Input year,month:1985,2

The number of days is 28

程序运行示例4如下：

Input year,month:1983,13

Input year,month:1983,-1

Input year,month:1983,1

The number of days is 31

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  MONTHS 12  int main()  {      int days[2][MONTHS] = {{31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},                                      {31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};      int year, month;      do      {          printf("Input year,month:");          scanf("%d,%d", &year, &month);      }      while (month < 1 || month > 12);   /\* 处理不合法数据的输入 \*/      if (((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0)) /\*闰年\*/          printf("The number of days is %d\n", days[1][month - 1]);      else  /\*非闰年\*/          printf("The number of days is %d\n", days[0][month - 1]);      return 0;  } |

Q1598.(10分)

利用矩阵相乘公式cij= aik\*bkj，编程计算m×n阶矩阵A和n×m阶矩阵B之积。

输入提示信息："Input 2\*3 matrix a:\n"  "Input 3\*2 matrix b:\n"

输入格式："%d"

输出提示信息："Results:\n"

输出格式："%6d"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  ROW 2  #define  COL 3  /\* 函数功能：计算矩阵相乘之积，结果存于二维数组c中 \*/  MultiplyMatrix(int a[ROW][COL], int b[COL][ROW], int int c[ROW][ROW])  {      int i, j, k;      for (i = 0; i < ROW; i++)      {          for (j = 0; j < ROW; j++)          {              c[i][j] = 0;              for (k = 0; k < COL; k++)              {                  c[i][j] = c[i][j] + a[i][k] \* b[k][j];              }          }      }  }  /\* 函数功能：输出矩阵a中的元素 \*/  void  PrintMatrix(int a[ROW][ROW])  {      int i , j ;      for (i = 0; i < ROW; i++)      {          for (j = 0; j < ROW; j++)          {              printf("%6d", a[i][j]);          }          printf("\n");      }  }  int main()  {      int a[ROW][COL], b[COL][ROW], c[ROW][ROW], i, j;      printf("Input 2\*3 matrix a:\n");      for (i = 0; i < ROW ; i++)      {          for (j = 0; j < COL; j++)          {              scanf("%d", &a[i][j]);          }      }      printf("Input 3\*2 matrix b:\n");      for (i = 0; i < COL; i++)      {          for (j = 0; j < ROW; j++)          {              scanf("%d", &b[i][j]);          }      }      MultiplyMatrix(a, b, c);      printf("Results:\n");      PrintMatrix(c);      return 0;  } |

Q250.(10分)插入排序（Insertion Sort）。用函数编程实现在一个按升序排序的数组中查找x应插入的位置，将x插入数组中，使数组元素仍按升序排列。

\*\*输入格式要求："%d" 提示信息："Input array size:" "Input array:" "Input x:"

\*\*输出格式要求："After insert %d:\n" "%4d"

程序运行示例如下：

Input array size:5

Input array:1 3 5 7 9

Input x:4

After insert 4:

1 3 4 5 7 9

### 参考答案

|  |
| --- |
| #include<stdio.h>  #define  N  20                   /\* 插入前数组最大元素个数 \*/  void  Insert(int a[], int n, int x);  main()  {      int  a[N+1];              /\* 定义数组长度为插入前的数组元素个数加1 \*/      int  x, i, n;      printf("Input array size:");      scanf("%d", &n);        /\* 输入插入前数组元素个数 \*/      printf("Input array:");      for (i=0; i<n; i++)      {        scanf("%d", &a[i]);   /\* 输入插入前已按升序排序的数组元素 \*/      }      printf("Input x:");      scanf("%d", &x);        /\* 输入待插入的元素x \*/      Insert(a, n, x);             /\* 插入元素x到已排序数组中 \*/      printf("After insert %d:\n", x);      for (i=0; i<n+1; i++)      {        printf("%4d", a[i]); /\* 输出插入x后的数组元素 \*/      }      printf("\n");  }  /\* 函数功能：将x插入到一个已按升序排序的数组中 \*/  void  Insert(int a[], int n, int x)  {      int  i = 0, pos;      while (i < n && x > a[i]) /\* 查找待插入位置 \*/      {        i++;      }      pos = i;                       /\* 记录元素x应插入的数组下标位置pos \*/      for (i = n-1; i>= pos; i--)/\* 从尾部开始移动pos及其后所有的元素 \*/      {        a[i+1] = a[i];           /\* 向后复制数组元素 \*/      }      a[pos] = x;                   /\* 插入元素x到位置pos \*/  } |

Q113.(30分)输入某班学生某门课的成绩（最多不超过40人），

当输入为负值时，表示输入结束，

用函数编程统计成绩不低于平均分的学生人数。

要求：

(1)按如下函数原型进行编程：

int GetAboveAver(int score[], int n);

(2)在主函数中：

输入学生成绩，

然后调用函数GetAboveAver计算成绩不低于平均分的学生人数，

最后输出该人数。

要求:

(1)学生成绩和平均分均定义为int类型

(2)\*\*无输入提示信息

\*\*输入格式为："%d"

\*\*输出格式为："Students of above average is %d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define N 40  int GetAboveAver(int score[], int n);  void main()  {      int score[N], m, n;      n = -1;   //1      do{          n++;          scanf("%d", &score[n]);//1      }while (score[n] >= 0);   //e2      m = GetAboveAver(score, n);   //2      printf("Students of above average is %d\n", m);  }  int GetAboveAver(int score[], int n)  {      int   i, sum,  aver,count;      count = 0;//1      sum = 0;//1      for (i=0; i<n; i++)//1      {          sum += score[i];//2      }      aver=sum/n;//1      for (i=0; i<n; i++)//1      {          if (score[i] >= aver)//2              count++;//1      }      return count;//1  } |

Q417.(10分)捕鱼和分鱼。ABCDE五人在某天夜里合伙去捕鱼，到第二天凌晨时都疲惫不堪，于是各自找地方睡觉。日上三竿，A第一个醒来，将鱼分为5份，把多余的1条鱼扔掉，拿走自己的1份。B第二个本来，也将鱼分5份，把多余的1条鱼扔掉，拿走自己的1份。CDE依次醒来，也按同样的方法拿鱼。问他们合伙至少捕了多少条鱼？

\*\*输出格式要求："Total number of fish catched=%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int n,i,x,flag=1;   for(n=6;flag;n++)   { for(x=n,i=1;flag&&i<=5;i++)        if((x-1)%5==0)            x=4\*(x-1)/5;        else            flag=0;     if(flag)        break;     else  flag=1;   }   printf("Total number of fish catched=%d\n",n);  } |

Q1344.(10分)求500以内的10个最大素数及其和并分别输出。

程序的运行示例如下：

499 491 487 479 467 463 461 457 449 443

sum=4696

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int sum\_prime(int a[]);  int prime(int n);    int main()  {      int i, a[10], sum;      sum = sum\_prime(a);      for (i = 0; i < 10; i++)          printf("%6d", a[i]);      printf("\n sum=%d\n", sum);      return 0;  }    int sum\_prime(int a[])  {      int i, j = 499, sum = 0;      for (i = 0; i < 10; )      {          if (prime(j))          {              sum = sum + j ;              a[i] = j;              i++;          }          j--;      }      return sum;  }  int prime(int n)  {      int i, k;      k = (int) sqrt((double)n);      for (i = 2; i <= k; i++)      {          if (n % i == 0)              return 0;      }      if (i >= k + 1)          return 1;    } |

Q399.(10分)二分法。用二分法求一元方程2x^3-4x^2+3x-6=0在(-10,10)区间的根。

\*\*输入格式要求："%f,%f" 提示信息："Enter x1,x2:"

\*\*输出格式要求："x=%6.2f\n"

程序运行示例如下：

Enter x1,x2:-10,10

x= 2.00

### 参考答案

|  |
| --- |
| #include "stdio.h"  #include "math.h"  main()  {float x0,x1,x2,fx0,fx1,fx2;   do   {  printf("Enter x1,x2:");      scanf("%f,%f",&x1,&x2);      fx1=2\*x1\*x1\*x1-4\*x1\*x1+3\*x1-6;      fx2=2\*x2\*x2\*x2-4\*x2\*x2+3\*x2-6;    } while(fx1\*fx2>0);   do   {  x0=(x1+x2)/2;      fx0=2\*x0\*x0\*x0-4\*x0\*x0+3\*x0-6;      if((fx0\*fx1)<0)      {  x2=x0;         fx2=fx0;      }      else      {  x1=x0;         fx2=fx0;       }   }while(fabs((double)fx0)>=1e-5);   printf("x=%6.2f\n",x0);  } |

Q1127.(10分)求2/1,3/2,5/3,8/5,13/8,21/13,.....前20项之和。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

int i, s1 = 2, s2 = 1;

float x, sum = 0;

for (i = 1; i <= 20; i++)

{

sum += s1 / s2;

x = s1;

s1 += s2;

s2 = x;

}

printf("sum = %9.6f\n", sum);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int i, s1 = 2, s2 = 1;      float x, sum = 0;        for (i = 1; i <= 20; i++)      {          sum += (float)s1 / s2;          x = s1;          s1 += s2;          s2 = x;      }      printf("sum = %9.6f\n", sum);  } |

Q339.(10分)\*输入一个正整数，要求以相反的顺序输出该数，例如输入12345，输出为54321。

\*\*输入格式要求："%d" 提示信息："Input the number:"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      unsigned int number;      printf("Input the number:");      scanf("%d",&number);      do      {          printf("%d",number%10);          number/=10;      }while(number!=0);      printf("\n");  } |

Q587.(10分)已知xyz+yzz=532，试编程求出算式中的x，y，z的值。

\*\*输出格式要求："x=%d,y=%d,z=%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      int x, y, z, i, result = 532;      for (x = 1; x <= 5; x++)          for (y = 1; y <= 3; y++)              for (z = 1; z <= 2; z++)              {                  i = (100 \* x + 10 \* y + z) + (100 \* y + 10 \* z + z);                  if (i == result)                      printf("x=%d,y=%d,z=%d\n", x, y, z);              }      return 0;  } |

Q1310.(10分)打印出以下图案：

\*

\* \* \*

\* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, j, k;        for (i = 1; i <= 4; i++)      {          for (j = 1; j <= (4 - i) + 5; j++)          {              printf("   ");          }          for (k = 1; k <= (2 \* i - 1) + 5; k++)          {              if (k <= 5)              {                  printf("   ");              }              else              {                  printf(" \* ");              }          }          printf("\n");      }      for (i = 1; i <= 3; i++)      {          for (j = 1; j <= i + 5; j++)          {              printf("   ");          }          for (k = 1; k <= (7 - 2 \* i) + 5; k++)          {              if (k <= 5)              {                  printf("   ");              }              else              {                  printf(" \* ");              }          }          printf("\n");      }        return 0;  } |

Q490.(10分)一位法官在审理一起盗窃案时，对涉及的4名嫌疑犯A、B、C、D进行了审问。4人分别供述如下：

A：“罪犯在B、C、D三人之中。”

B：“我没有作案，是C偷的。”

C：“在A和D中间有一个是罪犯。”

D：“B说的是事实。”

经过充分的调查，证实4人中只有两人说了真话，并且罪犯只有一个。请确定真正的罪犯。

\*\*输出格式要求："%d,%d,%d,%d\n" "A说得正确.\n" "B说得正确.\n" "C说得正确.\n" "D说得正确.\n"

说明：ABCD只能有一个说得正确，输出的四个整数为0或1，分别代表ABCD说得是否正确，如假设程序输出为:

1,0,0,0 <===此处代表A说得正确

A说得正确 <===用文字解释上面的含义，程序输出必须有这两行

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      short A, B, C, D, L1, L2, L3, L4;        for (A = 0; A <= 1; A++)          for (B = 0; B <= 1; B++)              for (C = 0; C <= 1; C++)                  for (D = 0; D <= 1; D++)                  {                      L1 = (B + C + D == 1); //"罪犯在B、C、D三人之中。"                      L2 = (!B && C);        //"我没有作案，是C偷的。"                      L3 = (A + D == 1);     //"在A和D中间有一个是罪犯。"                      L4 = L2;               //"B说的是事实"                      if (L1 + L2 + L3 + L4 == 2                              && A + B + C + D == 1)                          //四人中只有两人说了真话，并且罪犯只有一个                      {                          printf("%d,%d,%d,%d\n", A, B, C, D);                          if (A)                              printf("A说得正确.\n");                          if (B)                              printf("B说得正确.\n");                          if (C)                              printf("C说得正确.\n");                          if (D)                              printf("D说得正确.\n");                      }                  }        return 0;  } |

Q1334.(10分)编写程序，把一维数组a 输入任意6个整数，假设a 为： 7 4 5 9 1 2

输出：

7 4 5 9 1 2

2 7 4 5 9 1

1 2 7 4 5 9

9 1 2 7 4 5

5 9 1 2 7 4

4 5 9 1 2 7

\*\*输入格式要求："%d"

\*\*输出格式要求："%d "

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int a[6], i, j, k, m;      for (i = 0; i < 6; i++)          scanf("%d", &a[i]);      for (i = 5; i >= 0; i--)      {          printf("\n");          for (m = 0; m < 6; m++)              printf("%d ", a[m]);          k = a[5];          for (j = 5; j >= 1; j--)              a[j] = a[j - 1];          a[0] = k;      }  } |

Q436.(10分)除式还原（2）

下列除式中仅在商中给定一个7，其他打X的位置全部是任意数字，请还原。

X7XXXX

------------

除数----XXX| XXXXXXXX --------被除数

XXXX --------1

-------------

XXX --------2

XXX --------3

----------

XXXX --------4

XXX --------5

----------

XXXX --------6

XXXX --------7

----------

0

\*\*输出格式要求："No%2d:" "%d%d%d%d%d/" "%d=" "%d%d%d%d%d\n"

程序运行示例如下：

No 1:12128316/124=97809

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>    main()  {      int a[4],b,c[3],x,d[4],i=1;      for(a[0]=1010;a[0]<=1377;a[0]++)          for(b=112;b<=142;b++)              for(c[0]=8;c[0]<=9;c[0]++)                  if(b\*c[0]>1000&&(d[0]=a[0]-b\*c[0])>=10&&d[0]<100)                      for(a[1]=0;a[1]<=9;a[1]++)                          if((d[1]=d[0]\*10+a[1]-b\*7)>=100&&d[1]<b)                              for(a[2]=0;a[2]<=9;a[2]++)                                  for(c[1]=7;c[1]<=8;c[1]++)                                      if(b\*c[1]<1000&&(d[2]=d[1]\*10+a[2]-b\*c[1])>=10&&d[2]<100)                                          for(a[3]=0;a[3]<=99;a[3]++)                                              for(c[2]=8;c[2]<=9;c[2]++)                                                  if(d[2]\*100+a[3]-b\*c[2]==0)                                                  {                                                      printf("No%2d:",i++);                                                      printf("%d%d%d%d%d/",a[0],a[1],a[2],a[3]/10,a[3]%10);                                                      printf("%d=",b);                                                      printf("%d%d%d%d%d\n",c[0],7,c[1],0,c[2]);                                                  }  } |

Q335.(10分)\*对输入的行、单词和字符进行计数。我们将单词的定义进行化简，认为单词是不包含空格、制表符\t及换行符的字符序列。例如：“a+b+c”，认为是1个单词，它由5个字符组成。又如：“xy abc”，为2个单词，6个字符。输入ctrl+z结束。

\*\*输出格式要求："Lines=%d\nWords=%d\nChars=%d\n"

程序运行示例如下：

hi~

Good moring!

I'm Julie.

Lines=3

Words=5

Chars=28

### 参考答案

|  |
| --- |
| #define EOF -1  #define YES 1  #define NO 0  #include <stdio.h>  main()  {      int c,nl,nc,nw,inword;      inword=NO;      nl=nc=nw=0;      while((c=getchar())!=EOF)      {          ++nc;          if(c=='\n')              ++nl;          if(c=='\t'||c=='\n'||c==' ')              inword=NO;          else              if(inword==NO)              {                  inword=YES;                  ++nw;              }      }      printf("Lines=%d\nWords=%d\nChars=%d\n",nl,nw,nc);  } |

Q1161.(10分)

|  |
| --- |
| 假设某袋子中装有3个红球，5个白球，6个黑球，现从该袋子中任意取出8个球，下面程序的功能是输出这8个球中至少有一个白球的所有方案。找出其中错误并改正之。    #include <stdio.h>  main()  {      int i,j,k;      for(i=0;i<=3;i++)          for(j=0;j<=5;j++)          {              k=8-i-j;              if(0<=k<=6)                  printf(“hong=%d\t,bai=%d\t,hei=%d\t\n”,i,j,k);          }  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,j,k;      for(i=0;i<=3;i++)          for(j=1;j<=5;j++)          {              k=8-i-j;              if(0<=k&&k<=6)                  printf("hong=%d\t,bai=%d\t,hei=%d\t\n",i,j,k);          }  } |

Q280.(5分)在海军节开幕式上，有A、B、C三艘军舰要同时开始鸣放礼炮各21响。

已知A舰每隔5秒放1次，B舰每隔6秒放1次，C舰每隔7秒放1次。

假设各炮手对时间的掌握非常准确，

请编程计算观众总共可以听到几次礼炮声。

下面给出的程序有错误，请找到错误并改正之。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

\*\*输入输出格式要求：不要改变以下代码的输入输出格式。

|  |
| --- |
| #include <stdio.h>  main()  {      int  n, t;      for (t=0; t<20\*7; t++);      {          if (t%5 = 0 && t < 20\*5);          {              n++;          }          if (t%6 = 0 && t < 20\*6);          {              n++;          }          if (t%7 = 0);          {              n++;          }      }      printf("n=%d\n", n);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {          int  n = 0, t;      for (t=0; t<=20\*7; t++)      {          if (t%5 == 0 && t <= 20\*5)          {              n++;              continue;          }          if (t%6 == 0 && t <= 20\*6)          {              n++;              continue;          }          if (t%7 == 0)          {              n++;          }      }      printf("n=%d\n", n);  } |

Q584.(10分)任何一个自然数m的立方均可写成m个连续奇数之和。例如：

　1^3=1

　2^3=3+5

　3^3=7+9+11

　4^3=13+15+17+19

编程实现：输入一自然数n，求组成n3的n个连续奇数。

\*\*输入格式要求："%d" 提示信息："\nInput an integer here please:"

\*\*输出格式要求："%d\*%d\*%d=%d+%d"

程序运行示例如下：

Input an integer here please:5

5\*5\*5=21+23+25+27+29

### 参考答案

|  |
| --- |
| #include<stdio.h>    int main()  {      int i, j, n, sum, count;        printf("\nInput an integer here please:");      scanf("%d", &n);      printf("\n");        i = 1;      for (; i <= n \* n \* n; i += 2)      {          sum = 0;          count = 0;          for (j = i; j <= n \* n \* n; j += 2)          {              sum = sum + j;              count++;              if ((n \* n \* n == sum) && (count == n))                  break;          }          if ((sum == n \* n \* n) && (count == n))              break;      }        printf("%d\*%d\*%d=", n, n, n);        for (; i < j; i += 2)          printf("%d+", i);      printf("%d", i);  } |

Q428.(10分)八皇后问题。在一个8\*8的国际象棋棋盘上，有八个皇后，每个皇后占一格；要求皇后间不会出现相互“攻击”的现象，即不能有两个皇后处在同一行、同一列或同一对角线上。问共有多少种不同的方法。

\*\*输入格式要求：提示信息："The possible configuration of 8 queens are:\n"

\*\*输出格式要求：" [%2d]:" " %d"

程序运行示例如下：

The possible configuration of 8 queens are:

[ 1]: 1 5 8 6 3 7 2 4 [ 2]: 1 6 8 3 7 4 2 5 [ 3]: 1 7 4 6 8 2 5 3

[ 4]: 1 7 5 8 2 4 6 3 [ 5]: 2 4 6 8 3 1 7 5 [ 6]: 2 5 7 1 3 8 6 4

[ 7]: 2 5 7 4 1 8 6 3 [ 8]: 2 6 8 3 1 4 7 5 [ 9]: 2 6 1 7 4 8 3 5

[10]: 2 7 3 6 8 5 1 4 [11]: 2 7 5 8 1 4 6 3 [12]: 2 8 6 1 3 5 7 4

[13]: 3 5 7 1 4 2 8 6 [14]: 3 5 8 4 1 7 2 6 [15]: 3 5 2 8 1 7 4 6

[16]: 3 5 2 8 6 4 7 1 [17]: 3 6 8 1 4 7 5 2 [18]: 3 6 8 1 5 7 2 4

[19]: 3 6 8 2 4 1 7 5 [20]: 3 6 2 5 8 1 7 4 [21]: 3 6 2 7 1 4 8 5

[22]: 3 6 2 7 5 1 8 4 [23]: 3 6 4 1 8 5 7 2 [24]: 3 6 4 2 8 5 7 1

[25]: 3 7 2 8 5 1 4 6 [26]: 3 7 2 8 6 4 1 5 [27]: 3 8 4 7 1 6 2 5

[28]: 3 1 7 5 8 2 4 6 [29]: 4 6 8 2 7 1 3 5 [30]: 4 6 8 3 1 7 5 2

[31]: 4 6 1 5 2 8 3 7 [32]: 4 7 1 8 5 2 6 3 [33]: 4 7 3 8 2 5 1 6

[34]: 4 7 5 2 6 1 3 8 [35]: 4 7 5 3 1 6 8 2 [36]: 4 8 1 3 6 2 7 5

[37]: 4 8 1 5 7 2 6 3 [38]: 4 8 5 3 1 7 2 6 [39]: 4 1 5 8 2 7 3 6

[40]: 4 1 5 8 6 3 7 2 [41]: 4 2 5 8 6 1 3 7 [42]: 4 2 7 3 6 8 1 5

[43]: 4 2 7 3 6 8 5 1 [44]: 4 2 7 5 1 8 6 3 [45]: 4 2 8 5 7 1 3 6

[46]: 4 2 8 6 1 3 5 7 [47]: 5 7 1 3 8 6 4 2 [48]: 5 7 1 4 2 8 6 3

[49]: 5 7 2 4 8 1 3 6 [50]: 5 7 2 6 3 1 4 8 [51]: 5 7 2 6 3 1 8 4

[52]: 5 7 4 1 3 8 6 2 [53]: 5 8 4 1 3 6 2 7 [54]: 5 8 4 1 7 2 6 3

[55]: 5 1 4 6 8 2 7 3 [56]: 5 1 8 4 2 7 3 6 [57]: 5 1 8 6 3 7 2 4

[58]: 5 2 4 6 8 3 1 7 [59]: 5 2 4 7 3 8 6 1 [60]: 5 2 6 1 7 4 8 3

[61]: 5 2 8 1 4 7 3 6 [62]: 5 3 8 4 7 1 6 2 [63]: 5 3 1 6 8 2 4 7

[64]: 5 3 1 7 2 8 6 4 [65]: 6 8 2 4 1 7 5 3 [66]: 6 1 5 2 8 3 7 4

[67]: 6 2 7 1 3 5 8 4 [68]: 6 2 7 1 4 8 5 3 [69]: 6 3 5 7 1 4 2 8

[70]: 6 3 5 8 1 4 2 7 [71]: 6 3 7 2 4 8 1 5 [72]: 6 3 7 2 8 5 1 4

[73]: 6 3 7 4 1 8 2 5 [74]: 6 3 1 7 5 8 2 4 [75]: 6 3 1 8 4 2 7 5

[76]: 6 3 1 8 5 2 4 7 [77]: 6 4 7 1 3 5 2 8 [78]: 6 4 7 1 8 2 5 3

[79]: 6 4 1 5 8 2 7 3 [80]: 6 4 2 8 5 7 1 3 [81]: 7 1 3 8 6 4 2 5

[82]: 7 2 4 1 8 5 3 6 [83]: 7 2 6 3 1 4 8 5 [84]: 7 3 8 2 5 1 6 4

[85]: 7 3 1 6 8 5 2 4 [86]: 7 4 2 5 8 1 3 6 [87]: 7 4 2 8 6 1 3 5

[88]: 7 5 3 1 6 8 2 4 [89]: 8 2 4 1 7 5 3 6 [90]: 8 2 5 3 1 7 4 6

[91]: 8 3 1 6 2 5 7 4 [92]: 8 4 1 3 6 2 7 5

### 参考答案

|  |
| --- |
| #include<stdio.h>  #define NUM 8  int a[NUM+1];  main()  {  int number,i,k,flag,not\_finish=1,count=0;  i=1;  a[1]=1;  printf("The possible configuration of 8 queens are:\n");  while(not\_finish)  {      while(not\_finish&&i<=NUM)      {      for(flag=1,k=1;flag&&k<i;k++)          if(a[k]==a[i])  flag=0;      for(k=1;flag&&k<i;k++)          if((a[i]==a[k]-(k-i))||(a[i]==a[k]+(k-i)))              flag=0;      if(!flag)      {          if(a[i]==a[i-1])          {              i--;              if(i>1&&a[i]==NUM)                  a[i]=1;              else if(i==1&&a[i]==NUM)                  not\_finish=0;              else a[i]++;         }        else          if(a[i]==NUM)a[i]=1;          else         a[i]++;      }      else if(++i<=NUM)      {    if(a[i-1]==NUM)              a[i]=1;          else              a[i]=a[i-1]+1;      }      }      if(not\_finish)      {          ++count;          printf((count-1)%3?"  [%2d]:" : "\n   [%2d]:",count);          for(k=1;k<=NUM;k++)              printf(" %d",a[k]);          if(a[NUM-1]<NUM)              a[NUM-1]++;          else a[NUM-1]=1;          i=NUM-1;      }  }  } |

Q523.(10分)编程从键盘输入n，然后计算并输出n! = 1 × 2 × 3 × … × n。

\*\*输入格式要求："%d" 提示信息："Please enter n:"

\*\*输出格式要求："%d! = %ld\n"

程序运行示例如下：

Please enter n:10

10! = 3628800

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int   i, n;      long  p = 1;        /\* 因阶乘值取值范围较大，故p定义为长整型，并赋初值1 \*/      printf("Please enter n:");      scanf("%d", &n);      for (i = 1; i <= n; i++)      {          p = p \* i;                      /\* 做累乘运算 \*/      }      printf("%d! = %ld\n", n, p);    /\* 以长整型格式输出n的阶乘值 \*/  } |

Q290.(15分)编写程序求出用数字0至9可以组成多少个各位上没有重复数字的三位偶数。

\*\*\*\*要求输入提示信息为：无

\*\*\*\*输出格式要求为：%d\n

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a,b,c,count=0;      for(a=1; a<=9; a++)  //2      {          for(b=0; b<=9; b++)  //2          {              for(c=0; c<=9; c++)  //2              {                  if(a!=b&&a!=c&&b!=c&&c%2==0)    //2                  {                      count++;  //1                  }              }          }      }      printf("%d\n",count);    //1    } |

Q1727.(10分)

已知立方和不等式为 13+ 23+…+ m3< n 对指定的n值，试求满足上述立方和不等式的m的整数解。

输入提示信息："Please enter n:"

输入格式："%ld"

输出格式："m<=%1d\n"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  int main(void)  {      long i, m, n, sum = 0; //注意这里i必须声明为long类型      printf("Please enter n:");      scanf("%ld", &n);      for (i = 1; ; i++)      {          sum = sum + i \* i \* i;          if (sum >= n)              break;      }      m = i - 1;      printf("m<=%1d\n", m);      return 0;  } |

Q343.(10分)已知一个首项大于0的等差数列的前四项和为26，前四项的积为880，求此数列。

\*\*输出格式要求："%d, " "......\n"

程序运行示例如下：

2, 5, 8, 11, 14, 17, 20, 23, 26, 29, 32, 35, 38, 41, 44, 47, 50, 53, 56, 59, 62, ......

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a,b,c,d,i,flag;      for(flag=1,a=0;flag&&a<5;++a)          for(d=1;flag&&d<5;++d)          {              b=a+(a+d)+(a+2\*d)+(a+3\*d);              c=a\*(a+d)\*(a+2\*d)\*(a+3\*d);              if(b==26&&c==880)              {                  for(i=0;i<=20;++i)                      printf("%d, ",a+i\*d);                  printf("......\n");                  flag=0;              }          }  } |

Q424.(10分)魔术师的猜牌术。魔术师利用一副牌中的十三张黑桃，预先将它们排好后迭在一起，牌面朝下。对观众说：我不看牌，只数数就可以猜到每张牌是什么，我大声数数，你们听，不信？你们看看。魔术师将最上面的那张牌数为1，把它翻过来正好是黑桃A，将黑桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌下面，将第二张牌翻过来，正好是黑桃2，也将它放在桌子上。第三次数1、2、3，将前两张牌依次放在这迭牌下面，将第三张牌翻过来，正好是黑桃3。这样依次进行将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

\*\*输入格式要求：提示信息："The original order of cards is:"

\*\*输出格式要求："%d "

程序运行示例如下：

The original order of cards is:1 8 2 5 10 3 12 11 9 4 7 6 13

### 参考答案

|  |
| --- |
| #include <stdio.h>  int a[14];  main()  {int i,n,j=1;   printf("The original order of cards is:");   for(i=1;i<=13;i++)   {n=1;    do    {if(j>13)       j=1;     if(a[j])       j++;     else     {if(n==i)         a[j]=i;         j++;         n++;     }    }while(n<=i);   }   for(i=1;i<=13;i++)       printf("%d ",a[i]);   printf("\n");   } |

Q467.(10分)程序：确定消息的长度。

为了说明字符的读取方式，下面编写一个程序来计算消息的长度。在用户输入消息后，程序显示的长度如下：

Enter a message: Brevity is the soul of wit.

Your message was 27 character(s) long.

消息的长度包括空格和标点符号，但是不包括消息结尾处的换行符。

\*\*输入格式要求：提示信息："Enter a message:"

\*\*输出格式要求："Your message was %d character(s) long.\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      char ch;      int len = 0;        printf("Enter a message:");      ch = getchar();      while (ch != '\n')      {          len++;          ch = getchar();      }      printf("Your message was %d character(s) long.\n", len);        return 0;  } |

Q353.(10分)\*已知银行的存款利息如下。某人有2000元钱，要存二十年，问怎样存才能使二十年后得到的本利合计最多（假定银行对定期存款过期部分不付利息）？

月利息率=⎧ ⎩ ⎨ ⎪ ⎪ ⎪ ⎪ ⎪ ⎪ ⎪ ⎪ ⎪ ⎪ 0.63% 0.66% 0.69% 0.75% 0.84%  期限一年 期限二年 期限三年 期限五年 期限八年

\*\*输出格式要求："8 year:%d\t 5 year:%d\t 3 year:%d\t2 year:%d\t 1 year:%d\nTotal:%.2f\n"

程序运行示例如下：

8 year:0 5 year:4 3 year:0 2 year:0 1 year:0

Total:8841.01

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  {      int i8,i5,i3,i2,i1,n8,n5,n3,n2,n1;      float max=0,term;      for(i8=0;i8<3;i8++)          for(i5=0;i5<=(20-8\*i8)/5;i5++)              for(i3=0;i3<=(20-8\*i8-5\*i5)/3;i3++)                  for(i2=0;i2<=(20-8\*i8-5\*i5-3\*i3)/2;i2++)                  {                      i1=20-8\*i8-5\*i5-3\*i3-2\*i2;                      term=2000.0\*pow(1+0.0063\*12,(double)i1)                                  \*pow(1+2\*0.0066\*12,(double)i2)                                  \*pow(1+3\*0.0069\*12,(double)i3)                                  \*pow(1+5\*0.0075\*12,(double)i5)                                  \*pow(1+8\*0.0084\*12,(double)i8);                      if(term>max)                      {                          max=term;                          n1=i1;n2=i2;n3=i3;n5=i5;n8=i8;                      }                  }      printf("8 year:%d\t 5 year:%d\t 3 year:%d\t",n8,n5,n3);      printf("2 year:%d\t 1 year:%d\n",n2,n1);      printf("Total:%.2f\n",max);  } |

Q430.(10分)乘式还原。有乘法算式如下：

 ×     o   o o  o o    o  o o  o o  o o  o    o  o  o  o  o

18个O的位置上全部是素数（2、3、5或7），请还原此算式。

程序运行示例如下：

775

\* 33

-----

2325

2325

-----

25575

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define NUM 5  #define C\_NUM 4  int a[NUM+1];  int b[]={0,2,3,5,7};  main()  {      int i,not\_finish=1;      i=2;      a[1]=1;      while(not\_finish)      {          while(not\_finish&&i<=NUM)              if(a[i]>=C\_NUM)                  if(i==1&&a[i]==C\_NUM)                      not\_finish=0;                  else a[i--]=0;              else a[i++]++;          if(not\_finish)          {              long int sum1,sum2,sum3,sum4;              sum1=b[a[1]]\*100+b[a[2]]\*10+b[a[3]];              sum2=sum1\*b[a[5]];              sum3=sum1\*b[a[4]];              if(sum2>=2222&&sum2<=7777&&f(sum2)&&sum3>=2222&&sum3<=7777&&f(sum3))                  if((sum4=sum2+sum3\*10)>=22222&&sum4<=77777&&f(sum4))                  {                      printf("  %d\n",sum1);                      printf("\*  %d%d\n",b[a[4]],b[a[5]]);                      printf("-----\n");                      printf(" %d\n",sum2);                      printf("%d\n",sum3);                      printf("-----\n");                      printf("%d\n",sum4);                  }               i=NUM;          }      }  }    f(sum)  long sum;  {   int i,k,flag;      while(sum>0)      {          i=sum%10;          for(flag=0,k=1;!flag&&k<=C\_NUM;k++)          if(b[k]==i)          {              flag=1;              break;          }          if(!flag)   return(0);          else      sum=sum/10;      }      return (1);  } |

Q336.(10分)输出如下图所示的下三角形乘法九九表。

1 2 3 4 5 6 7 8 9

------------------------------------------------------------------

1

2 4

3 6 9

4 8 12 16

…

9 18 27 36 45 54 63 72 81

\*\*输出格式要求："\n-----------------------------\n" "%4d"

程序运行示例如下：

1 2 3 4 5 6 7 8 9

-----------------------------

1

2 4

3 6 9

4 8 12 16

5 10 15 20 25

6 12 18 24 30 36

7 14 21 28 35 42 49

8 16 24 32 40 48 56 64

9 18 27 36 45 54 63 72 81

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,j;      for(i=1;i<10;i++)          printf("%4d",i);      printf("\n-----------------------------\n");      for(i=1;i<10;i++)          for(j=1;j<=i;j++)              printf((j==i)?"%4d\n":"%4d",i\*j);  } |

Q341.(10分)求555555的约数中最大的三位数是多少？

\*\*输入格式要求："%ld" 提示信息："Please input number"

\*\*输出格式要求："The max factor with 3 digits in %ld is %d.\n"

程序运行示例如下：

Please input number 555555

The max factor with 3 digits in 555555 is 777.

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int j; long n;      printf("Please input number");      scanf("%ld",&n);      for(j=999;j>=100;j--)          if(n%j==0)          {              printf("The max factor with 3 digits in %ld is %d.\n",n,j);              break;          }  } |

Q778.(20分)每个苹果0.8元，第一天买2个苹果，第二天开始，每天买前一天的2倍，直至购买的苹果个数达到不超过100的最大值。

编写程序求每天平均花多少钱？

\*\*要求输入提示信息为：无输入提示信息

\*\*要求输入格式为：无

\*\*要求输出格式为："%.2f\n"

### 参考答案

|  |
| --- |
| #include<stdio.h>  main()  {      int i = 2;      int j = 0;      int day = 0;      float k = 0;      while (i < 100) //2      {          j = j + i;//1          day = day + 1;//1          i = i \* 2; //1      }      k = j / day \* 0.8; //2      printf("%.2f\n", k); //2  } |

Q1739.(10分)编程打印以下图案。

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int  i , j , k;      for (i = 1; i <= 4; i++)              //控制行数      {          for (k = 1; k <= (2 \* i - 1); k++)   //控制每行输出的\*号个数          {              printf("\*");          }          printf("\n");                       //输出一行后换行      }  } |

Q414.(10分)减式还原。编写程序求解下式中各字母所代表的数字，不同字母代表不同的数学。

\*\*输出格式要求：

" PEAR %d%d%d%d\n"

" ERA - %d%d%d\n"

"----------- ----------------\n"

" PEA %d%d%d\n"

程序运行示例如下：

PEAR 1098

ERA - 989

----------- ----------------

PEA 109

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int p,e,a,r;   for(p=1;p<=9;p++)      for(e=0;e<=9;e++)         if(p!=e)           for(a=1;a<=9;a++)              if(a!=p&&a!=e)                for(r=0;r<=9;r++)                  if(r!=p&&r!=e&&r!=a&&p\*1000+e\*100+a\*10+r-(a\*100+r\*10+a)==p\*100+e\*10+a)                  {printf("    PEAR        %d%d%d%d\n",p,e,a,r);                   printf("     ERA       -  %d%d%d\n",a,r,a);                   printf("-----------   ----------------\n");                   printf("     PEA           %d%d%d\n",p,e,a);                  }  } |

Q137.(15分)用100元人民币兑换10元,5元和1元的纸币(每一种都要有)共50张,请用穷举法编程计算共有几种兑换方案,每种方案各兑换多少张纸币.

\*\*\*\*要求输入提示信息为：无

\*\*\*\*输出格式要求为：

"x = %d, y = %d, z = %d\n"

"count = %d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,j,k,count=0;      for(i=1;i<=9;i++)    //1      {          for(j=1;j<=19;j++)   //1          {              for(k=1;k<=99;k++)   //1              {                  if(10\*i+5\*j+k==100&&i+j+k==50)  //1                  {                      printf("x = %d, y = %d, z = %d\n",i,j,k);   //1                      count++;    //1                  }              }          }      }      printf("count = %d\n",count);   //1  } |

Q1737.(10分)编程打印以下图案。

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*输出格式要求："%c"

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int    i, j, k;      char  space = ' ';      for (i = 1; i <= 4; i++)      {          for (j = 1; j <= i; j++)          {              printf("%c", space);          }          for (k = 1; k <= 6; k++)          {              printf("\*");          }          printf("\n");      }  } |

Q331.(10分)从键盘输入十个整数，求这十个整数的和。

\*\*输入格式要求："%d" 提示信息："Enter the No.%d="

\*\*输出格式要求："Total=%d\n"

程序运行示例如下：

Enter the No.1=10

Enter the No.2=9

Enter the No.3=8

Enter the No.4=7

Enter the No.5=6

Enter the No.6=5

Enter the No.7=4

Enter the No.8=3

Enter the No.9=2

Enter the No.10=1

Total=55

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int count,num,total;      count=0; total=0;      while(count<10)      {          count++;          printf("Enter the No.%d=",count);          scanf("%d",&num);          total+=num;      }      printf("Total=%d\n",total);  } |

Q476.(10分)计算机用一个32位的硬件寄存器来计算计算机工作时所需要的时间，其起始时间为1970年1月1日0时0分0秒，底层硬件晶体震荡电路每1秒会使该寄存器加1，请问到多少年后该寄存器会溢出？

\*\*输出格式要求："%f"

程序运行示例如下：

136.192520

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      printf("%f\n", (ldexp(1.0, 32) / (365 \* 24 \* 60 \* 60)));        return 0;  } |

Q418.(10分)出售金鱼。买买提将养的一缸金鱼分五次出售：第一次卖出全部的一半加二分之一条；第二次卖出余下的三分之一加三分之一条；第三次卖出余下的四分之一加四分之一条；第四次卖出余下的五分之一加五分之一条；最后卖出余下的11条。问原来鱼缸中有多少条鱼？

\*\*输出格式要求："There are %d fishes at first.\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int i,j,n=0,x;   for(i=23;n==0;i+=2)   { for(j=1,x=i;j<=4;j++)       if((x+1)%(j+1)==0)            x-=(x+1)/(j+1);       else{x=0;break;}     if(x==11)     {  printf("There are %d fishes at first.\n",i);        n=1;         }     }  } |

Q1321.(10分)利用泰勒级数sin(x)≈ x - x^3/3! + x^5/5! - x^7/7! + x^9/9! ...

计算sin(x) 的值。要求最后一项的绝对值小于10-5，并统计出此时累加了多少项。

|  |
| --- |
| #include  <math.h>  #include  <stdio.h>    main()  {      int n = 1,count = 1;      float x;      double sum , term;        printf("Input x: ");      scanf("%d", &x);        sum = x;      term = x;      do      {            term = -term\*x\*x/(n+1)\*(n+2);          sum = sum + term;          n++;          count++;      }while (fabs(term) <= 1e-5);        printf("sin(x) = %f, count = %d\n", sum, count);  } |

### 参考答案

|  |
| --- |
| #include  <math.h>  #include  <stdio.h>  main()  {      int n = 1,count = 1;      float x;      double sum , term;        printf("Input x: ");      scanf("%f", &x);      sum = x;      term = x;      do      {            term = -term\*x\*x/((n+1)\*(n+2));          sum = sum + term;          n = n + 2;          count++;      }while (fabs(term) >= 1e-5);        printf("sin(x) = %f, count = %d\n", sum, count);  } |

Q551.(10分)小红有5本新书，借给3位小朋友，若每人每次只能借一本，则可以有多少种不同的借法？

\*\*输出格式要求："%d:%d,%d,%d\n"

程序运行示例如下：

1:1,2,3

2:1,2,4

3:1,2,5

4:1,3,2

5:1,3,4

6:1,3,5

7:1,4,2

8:1,4,3

9:1,4,5

10:1,5,2

11:1,5,3

12:1,5,4

13:2,1,3

14:2,1,4

15:2,1,5

16:2,3,1

17:2,3,4

18:2,3,5

19:2,4,1

20:2,4,3

21:2,4,5

22:2,5,1

23:2,5,3

24:2,5,4

25:3,1,2

26:3,1,4

27:3,1,5

28:3,2,1

29:3,2,4

30:3,2,5

31:3,4,1

32:3,4,2

33:3,4,5

34:3,5,1

35:3,5,2

36:3,5,4

37:4,1,2

38:4,1,3

39:4,1,5

40:4,2,1

41:4,2,3

42:4,2,5

43:4,3,1

44:4,3,2

45:4,3,5

46:4,5,1

47:4,5,2

48:4,5,3

49:5,1,2

50:5,1,3

51:5,1,4

52:5,2,1

53:5,2,3

54:5,2,4

55:5,3,1

56:5,3,2

57:5,3,4

58:5,4,1

59:5,4,2

60:5,4,3

### 参考答案

|  |
| --- |
| #include<stdio.h>  int main()  {      int i, j, k, count = 0;      for (i = 1; i <= 5; i++)          for (j = 1; j <= 5; j++)              for (k = 1; i != j && k <= 5; k++)                  if (k != i && k != j)                  {                      count++;                      printf("%d:%d,%d,%d\n", count, i, j, k);                    }        return 0;  } |

Q1706.(10分)从键盘任意输入一个4位数x，编程计算x的每一位数字相加之和（忽略整数前的正负号）。例如，输入x为1234，则由1234分离出其千位1、百位2、十位3、个位4，然后计算1+2+3+4=10，并输出10。

输入提示信息："Input data is:"

输入格式："%d"

输出格式："The sum of the total bit is %d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  {      int i1, i2, i3, i4, k, n;      printf("Input data is:");      scanf("%d", &n);      k = fabs(n);                    //取绝对值      i1 = k / 1000;                  //分离出千位      i2 = (k - i1 \* 1000) / 100;             //分离出百位      i3 = (k - i1 \* 1000 - i2 \* 100) / 10;   //分离出十位      i4 = k % 10;                    //分离出个位      printf("The sum of the total bit is %d\n", i1 + i2 + i3 + i4);  } |

Q465.(10分)程序：数列求和

编写一个程序对用户输入的整数数列进行求和计算。

\*\*输入格式要求："%d" 提示信息："This program sums a series of integers.\n" "Enter integers (0 to terminate):"

\*\*输出格式要求："The sum is: %d\n"

下面显示的是用户可见的内容：

This program sums a series of integers.

Enter integers (0 to terminate) : 8 23 71 5 0

The sum is : 107

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int n, sum = 0;        printf("This program sums a series of integers.\n");      printf("Enter integers (0 to terminate):");        scanf("%d", &n);      while (n != 0)      {          sum += n;          scanf("%d", &n);      }      printf("The sum is: %d\n", sum);        return 0;  } |

Q1736.(10分)

按如下格式打印100以内整数的平方根表。

\*\*输出格式要求："%7d" （表头）  "%d" （每行开始数字） "%7.3f" （平方根）

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    main()  {      int  m, n;      for (m = 0; m < 10; m++)      {          printf("%7d", m);           //打印表头      }      printf("\n");      for (n = 0; n < 10; n++)         //乘数n从1变化到9      {          printf("%d", n);            //输出每行的开头数字          for (m = 0; m < 10; m++)             //被乘数m从1变化到9          {              printf("%7.3f", sqrt(n \* 10 + m));          }                   //输出第m行n列中的值          printf("\n");               //输出换行符,准备打印下一行      }  } |

Q1337.(20分)计算1~100之间的所有7的倍数之和。

\*\*输出格式要求："sum=%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, sum = 0;      for (i = 1; i <= 100; i++)      {          if (i % 7 == 0)          {              sum += i;          }      }      printf("sum=%d\n", sum);      return 0;  } |

Q556.(10分)编写一个简单的23 跟火柴游戏程序，实现人跟计算机玩这个游戏的程序。

游戏规则是：

1、两个游戏者开始拥有23 跟火柴棒；

2、每个游戏者轮流移走1 根、2 根或3 根火柴；

3、谁取走最后一根火柴为失败者。

程序一次运行示例如下：

这里是23 根火柴游戏！！

注意：最大移动火柴数目为三根

请输入移动的火柴数目：

3

您移动的火柴数目为：b

您移动后剩下的火柴数目为：20

计算机移动的火柴数目为：3

计算机移动后剩下的火柴数目为：17

请输入移动的火柴数目：

2

您移动的火柴数目为：b

您移动后剩下的火柴数目为：15

计算机移动的火柴数目为：2

计算机移动后剩下的火柴数目为：13

请输入移动的火柴数目：

3

您移动的火柴数目为：b

您移动后剩下的火柴数目为：10

计算机移动的火柴数目为：1

计算机移动后剩下的火柴数目为：9

请输入移动的火柴数目：

1

您移动的火柴数目为：b

您移动后剩下的火柴数目为：8

计算机移动的火柴数目为：3

计算机移动后剩下的火柴数目为：5

请输入移动的火柴数目：

2

您移动的火柴数目为：b

您移动后剩下的火柴数目为：3

计算机移动的火柴数目为：2

计算机移动后剩下的火柴数目为：1

请输入移动的火柴数目：

3

您移动的火柴数目为：b

您移动后剩下的火柴数目为：-2

对不起！您输了！

### 参考答案

|  |
| --- |
| #include<stdio.h>  int main()  {      int g = 23;      int k = 3;      int b, c;      printf("这里是23 根火柴游戏！！\n");      printf("注意：最大移动火柴数目为三根\n");      do      {          printf("请输入移动的火柴数目：\n");          scanf("%d", &b);            if (b < 1 || b > 3)          {              printf("对不起！您输入的了不合适的数目，请点击任意键重新输入！\n");              printf("您输入移动火柴数目：\n");              scanf("%d", &b);          }          else          {              g = g - b;              printf("您移动的火柴数目为：%b\n", b);              printf("您移动后剩下的火柴数目为：%d\n", g);          }            if (!g || g < 0)          {              printf("对不起！您输了！\n");              break;          }          else          {              c = (g - 1) % (k + 1);              if (!c)c = 1;              g = g - c;              printf("计算机移动的火柴数目为：%d\n", c);              printf("计算机移动后剩下的火柴数目为：%d\n", g);              if (!g || g < 0)              {                  printf("恭喜您！您赢了！ \n");                  break;              }          }      }      while (g > 0);      return 0;  } |

Q487.(20分)求1到100以内能被3整除但不能被7整除的数之和。

\*\*输出格式要求："sum=%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int k;      int sum;        k = 1;      sum = 0;        while (k <= 100)      {          if (k % 3 == 0 && k % 7 != 0)              sum = sum + k;          k++;      }        printf("sum=%d\n", sum);        return 0;  } |

Q1305.(10分)一球从100米高度自由落下，每次落地后反弹回原高度的一半，再落下，求它在第10次落地时，共经过多少米？第10次反弹多高？

\*\*输入格式要求："%f"

\*\*输出格式要求："%f"

程序运行示例如下：

第10次落地时共经过299.609375米。

第10次反弹0.097656米。

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      float sn = 100.0, hn = sn / 2;      int n;        for (n = 2; n <= 10; n++)      {          sn = sn + 2 \* hn;          hn = hn / 2;      }      printf("第10次落地时共经过%f米。\n", sn);      printf("第10次反弹%f米。\n", hn);        return 0;  } |

Q1295.(10分)下面程序的功能是求解爱因斯坦数学题。爱因斯坦曾出过这样一道数学题：有一条长阶梯，若每步跨2阶，最后剩下1阶；若每步跨3阶，最后剩下2阶；若每步跨5阶，最后剩下4阶；若每步跨6阶，最后剩下5阶；只有每步跨7阶，最后才正好1阶不剩。编程打印这条阶梯共有多少阶。找出其中错误，并改正之。

|  |
| --- |
| #include <stdio.h>  main()  {      int  x,find;        x=0;      do{          ++x;          find=x%2=1&&x%3=2&&x%5=4&&x%6=5&&x%7=0;      }while (find);      printf("x=%d\n",x);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int  x,find=0;      x=0;      do{          ++x;          find=x%2==1&&x%3==2&&x%5==4&&x%6==5&&x%7==0;      }while (!find);      printf("x=%d\n",x);  } |

Q398.(10分)梯形法。用梯形法编程求函数f(a)=x 2 +2x+1

的定积分，∫ 2 0 f(x)dx

的值。

\*\*输入格式要求："%lf,%lf,%lf" 提示信息："Enter n,a,b:"

\*\*输出格式要求："Sum=%lf\n"

程序运行示例如下：

Enter n,a,b:10,0,2

Sum=10.604000

### 参考答案

|  |
| --- |
| #include "stdio.h"  main()  { int i;    double s=0,h,f0,f1,n,a,b;    printf("Enter n,a,b:");    scanf("%lf,%lf,%lf",&n,&a,&b);    h=(b-a)/n;    f0=a\*a+2\*a+1;    for(i=0;i<=n;i++)  { a=a+h;    f1=a\*a+2\*a+1;      s+=(f0+f1)\*h/2;      f0=f1;    }    printf("Sum=%lf\n",s);  } |

Q426.(10分)常胜将军。现有21根火柴，两人轮流取，每人每次可以取1至4根，不可多取，也不能不取，谁取最后一根火柴谁输。请编写一个程序进行人机对弈，要求人先取，计算机后取；计算机一方为常胜将军。

\*\*输入格式要求："%d" 提示信息："Game begin:\n" "How many sticks do you wish to take (1~%d)?"

\*\*输出格式要求：" %d sticks left in the pile.\n" " You have taken the last sticks.\n"

" \*\*\*You lose!\nGame Over.\n" "Computer take %d sticks.\n"

程序运行示例如下：

Game begin:

How many sticks do you wish to take (1~4)?3

18 sticks left in the pile.

Computer take 2 sticks.

16 sticks left in the pile.

How many sticks do you wish to take (1~4)?3

13 sticks left in the pile.

Computer take 2 sticks.

11 sticks left in the pile.

How many sticks do you wish to take (1~4)?3

8 sticks left in the pile.

Computer take 2 sticks.

6 sticks left in the pile.

How many sticks do you wish to take (1~4)?3

3 sticks left in the pile.

Computer take 2 sticks.

1 sticks left in the pile.

How many sticks do you wish to take (1~1)?1

You have taken the last sticks.

\*\*\*You lose!

Game Over.

### 参考答案

|  |
| --- |
| #include<stdio.h>  main()  {int a=21,i;   printf("Game begin:\n");  while(a>0)      {do          {printf("How many sticks do you wish to take (1~%d)?",a>4?4:a);           scanf("%d",&i);          }while(i>4||i<1||i>a);            if(a-i>0)              printf(" %d sticks left in the pile.\n",a-i);            if((a-i)<=0)              {               printf(" You have taken the last sticks.\n");               printf(" \*\*\*You lose!\nGame Over.\n");               break;               }              else               printf("Computer take %d sticks.\n",5-i);            a-=5;            printf(" %d sticks left in the pile.\n",a);      }  } |

Q589.(10分)小明今年12岁，他母亲比他大24岁。编写一个程序计算小明的母亲在几年后比小明的年龄大一倍，那时他们两人的年龄各是多少？

\*\*输出格式要求："year=%d\n" "mingAge=%d\n" "motherAge=%d\n"

程序运行示例如下：

year=12

mingAge=24

motherAge=48

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      int mingAge, motherAge, year;      mingAge = 12;      motherAge = mingAge + 24;      year = 1;      while ((motherAge + year) != 2 \* (mingAge + year))      {          year = year + 1;      }      printf("year=%d\n", year);      printf("mingAge=%d\n", mingAge + year);      printf("motherAge=%d\n", motherAge + year);      return 0;  } |

Q1704.(10分)编程计算1×2×3+3×4×5+…+99×100×101的值。

输入格式：无

输出格式："sum = %ld"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      long i;  //注意这里i必须定义为long类型      long term, sum = 0;      for (i=1; i<=99; i=i+2)      {          term = i \* (i + 1) \* (i + 2);          sum = sum + term;      }      printf("sum = %ld", sum);  } |

Q604.(10分)用双重循环编程打印以下图形：

ABCDEF

BCDEF

CDEF

DEF

EF

F

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      char i, ch;      for (i = 0; i < 6; i++)      {          for (ch = 'A' + i; ch <= 'F'; ch++)          {              printf("%c", ch);          }          printf("\n");      }      return 0;  } |

Q466.(10分)程序：计算整数中数字的位数。

\*\*输入格式要求："%d" 提示信息："Enter a nonnegative integer:"

\*\*输出格式要求："The number has %d digit(s).\n"

程序运行示例如下：

Enter a nonnegative integer: 60

The number has 2 digit(s).

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int digits = 0, n;        printf("Enter a nonnegative integer:");      scanf("%d", &n);        do      {          n /= 10;          digits++;      }      while (n > 0);        printf("The number has %d digit(s).\n", digits);        return 0;  } |

Q433.(10分)乘式还原

A代表数字0到9中的前五个数字，Z代表后五个数字，请还原下列乘式

AZA

\* AAZ

-------

AAAA

AAZZ

ZAA

-------

ZAZAA

\*\*输出格式要求："\n %ld\n" "\* %ld\n" "--------------\n" " %ld\n %ld\n %ld\n" " %ld\n"

程序运行示例如下：

372

\* 246

--------------

2232

1488

744

--------------

91512

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>    int main()  {      long i,j,k,l,m,n,term,t1,t2,t3;      int flag;      for(i=0;i<=4;++i)          for(j=5;j<=9;++j)              for(k=0;k<=4;++k)              {                  term=100\*i+10\*j+k;                  for(flag=0,n=0;n<4&&!flag;)                      flag=jud((t3=++n\*100\*term)/100,"001");                  if(flag)                  {                      for(flag=0,m=0;m<4&&!flag;)                          flag=jud((t2=++m\*10\*term)/10,"1100");                      if(flag)                      {                          for(flag=0,l=5;l<9&&!flag;)                              flag=jud(t1=++l\*term,"0000");                          if(flag&&jud(t1+t2+t3,"00101"))                              print(term,n\*100+m\*10+l,t1,t2,t3);                      }                    }      }  }  print(a,b,s1,s2,s3)     long a,b,s1,s2,s3;  {      printf("\n   %ld\n",a);      printf("\*  %ld\n",b);      printf("--------------\n");      printf("  %ld\n %ld\n %ld\n",s1,s2/10,s3/100);      printf("--------------\n");      printf(" %ld\n",a\*b);  }  jud(q,pflag)     long q;     char \*pflag;  {      while(q!=0&&pflag!=NULL)          if(\*pflag-'0'!=(q%10>=5?1:0) )              return (0);          else{q/=10;++pflag;}      if(q==0&&\*pflag==0)          return(1);      else return(0);  } |

Q1318.(10分)用牛顿迭代法求根。方程为ax^3+bx^2+cx+d=0，系数a、b、c、d由主函数输入，求x在1附近的一个实根。求出根后，由主函数输出。

程序的运行示例如下：

输入方程的系数a、b、c、d：

1 2 3 4

方程是： 1.00x^3+ 0.00x^2+ 0.00x+ 0.00=0

X= 0.0000176

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    float solut(float a, float b, float c, float d)  {      float x = 1, x0, f, f1;        do      {          x0 = x;          f = ((a \* x0 + b) \* x0 + c) \* x0 + d;          f1 = (3 \* a \* x0 + 2 \* b) \* x0 + c;          x = x0 - f / f1;      }      while (fabs(x - x0) >= 1e-5);        return x;  }    int main()  {      float a, b, c, d;        printf("\n输入方程的系数a、b、c、d：\n");      scanf("%f,%f,%f,%f", &a, &b, &c, &d);      printf("\n方程是：%5.2fx^3+%5.2fx^2+%5.2fx+%5.2f=0", a, b, c, d);      printf("\nX=%10.7f\n", solut(a, b, c, d));        return 0;  } |

Q1342.(10分)编程验证哥德巴赫猜想：任意一个充分大的偶数，可以用两个素数之和表示。如：

4 = 2 + 2 6 = 3 + 3。

\*\*输入格式要求："%d" 提示信息："Input a number:\n" 输入奇数时报错： "Input error!\n"

\*\*输出格式要求：输入偶数时显示"%d=%d+%d\n"

程序的运行的输入输出样例1：

屏幕先显示提示信息：

Input a number:

然后用户键盘输入：

9

最后屏幕输出：

Input error!

程序的运行的输入输出样例2：

屏幕先显示提示信息：

Input a number:

然后用户键盘输入：

100

最后屏幕输出：

100=3+97

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      int i, m;      int p, q, flagp, flagq;        printf("Input a number:\n");      scanf("%d", &m);        if (m % 2 == 0)      {          for (p = 2; p < m; p++)          {              q = m - p;                /\*以下判断p是否为素数\*/              flagp = 1;              for (i = 2; i <= sqrt(p) && flagp; i++)              {                  if (p % i == 0)                  {                      flagp = 0;                  }              }                /\*以下判断q是否为素数\*/              flagq = 1;              for (i = 2; i <= sqrt(q) && flagq; i++)              {                  if (q % i == 0)                  {                      flagq = 0;                  }              }                /\*若是素数将其打印，并退出循环\*/              if (flagp && flagq)              {                  printf("%d=%d+%d\n", m, p, q);                  break;              }          }      }      else          printf("Input error!\n");        return 0;  } |

Q553.(10分)A、B、C、D、E五人在某天夜里合伙去捕鱼，到第二天凌晨时都疲惫不堪，于是各自找地方睡觉。日上三竿，A第一个醒来，他将鱼分为五份，把多余的一条鱼扔掉，拿走自己的一份，B第二个醒来，也将鱼分为五份，把多余的一条鱼扔掉，拿走自己的一份。C、D、E依次醒来，也按同样的方法拿鱼。问他们合伙至少补了多少条鱼？

\*\*输出格式要求："%d"

### 参考答案

|  |
| --- |
| #include<stdio.h>  int main()  {      int n, i, x, flag = 1;      for (n = 6; flag; n++)      {          for (x = n, i = 1; i <= 5 && flag; i++)          {              if ((x - 1) % 5 == 0)                  x = 4 \* (x - 1) / 5;              else                  flag = 0;          }          if (flag)              break;          else              flag = 1;      }      printf("%d\n", n);      return 0;  } |

Q1582.(10分)爱因斯坦数学题。爱因斯坦曾出过这样一道数学题：有一条长阶梯，若每步跨2阶，则最后剩下1阶，若每步跨3阶，则最后剩下2阶，若每步跨5阶，则最后剩下4阶，若每步跨6阶，则最后剩下5阶，只有每步跨7阶，最后才正好1阶不剩。请问，这条阶梯共有多少阶？

|  |
| --- |
| #include <stdio.h>  main()  {      int x,find=0;        while (!find)      {              if (x%2=1 && x%3=2 && x%5=4 && x%6=5 && x%7=0)                    {                      printf("x=%d\n,x);                          find=1;           }                  x++ ;      }  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  void main()  {      int x=1,find=0;        while (!find)      {          if( x%2==1 && x%3==2 && x%5==4 && x%6==5 && x%7==0 )          {              printf("x=%d\n",x);              find=1;          }          x++ ;      }  } |

Q210.(30分)所谓素数是指这个数只能被1和自身整除。要求在主函数输入一个数，

调用函数Fun()判断该数是否是素数。打印信息在主函数中进行。例如：

从键盘输入5，5是素数则打印如下信息："5 is a prime number".

又如:从键盘输入4, 4不是素数则打印如下信息:"4 is not a prime number"

负数、0和1均不是素数。对输入的数据要考虑数据的合法性，不满足条件的数

要重新输入直到满足条件为止。不能使用全局变量,不按给定的函数原型编写程序

不给分。Fun()函数原型如下：

int Fun(int m);

\*\*输入数据提示信息："Please input a number:\n"

注：该提示信息请放在循环体外

\*\*输入数据格式为："%d"

\*\*输出格式要求：

若是素数输出数据格式为："%d is a prime number\n"

若不是素数输出数据格式为： "%d is not a prime number\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  int Fun(int m);  int main()  {      int m, flag;      printf("Please input a number:\n");      do      {          scanf("%d", &m); //1      }      while (m < 2);  //e1      flag = Fun(m); //3      if (flag) //1          printf("%d is a prime number\n", m);      else          printf("%d is not a prime number\n", m); //1      return 0;  }    int Fun(int m)//2  {      int i, flag = 1; //1      for (i = 2; i < m; i++) //1      {          if (m % i == 0) //1          {              flag = 0; //1              break;          }      }      return flag;//2  } |

Q1303.(10分)企业发放的奖金根据利润提成。利润（I）低于或等于10万元时，奖金可提10%；利润高于10万元，低于20万（100000<I<=200000）元时，低于10万元的部分按10%提成，高于10万元的部分，可提成7.5%；200000<I<=400000时，高于20万元的部分按5%提成；400000<I<=600000时，高于40万元的部分按3%提成；600000<I<=1000000时，高于60万的部分按1.5%提成；I>1000000时，超过100万的部分按1%提成。从键盘输入当月利润I，求应发放奖金总数。要求：用switch语句编程序实现。

\*\*输入格式要求："%ld" 提示信息："请输入利润："

\*\*输出格式要求："奖金是%10.2f"

程序的运行示例如下：

请输入利润：234000

奖金是 59500.00

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      long i;      int branch;      float temp, bonus, bon1, bon2, bon4, bon6, bon10;        /\* 初始化变量 \*/      bon1 = 100000 \* 0.1;      bon2 = bon1 + 100000 \* 0.075;      bon4 = bon2 + 200000 \* 0.05;      bon6 = bon4 + 200000 \* 0.03;      bon10 = bon6 + 400000 \* 0.015;      printf("请输入利润：");      scanf("%ld", &i);      temp = i / 100000;      if (i <= 100000)          branch = 0;      else          branch = temp;      if (branch > 10)      {          branch = 10;      }      /\* 计算 \*/      switch (branch)      {      case 0:          bonus = i \* 0.1;          break;      case 1:          bonus = bon1 + (i - 100000) \* 0.075;          break;      case 2:      case 3:          bonus = bon2 + (i - 200000) \* 0.05;          break;      case 4:      case 5:          bonus = bon4 + (i - 400000) \* 0.03;          break;      case 6:      case 7:      case 8:      case 9:          bonus = bon6 + (i - 600000) \* 0.015;          break;      case 10:          bonus = bon10 + (i - 1000000) \* 0.01;      }      printf("奖金是%10.2f", bonus);        return 0;  } |

Q1731.(20分)

编程计算下面的分段函数，根据从键盘输入的整数x的值，在屏幕上输出y值。

\*\*输入格式要求："%d" 提示信息："Input x: "  
\*\*输出格式要求："y=%f\n"

程序运行示例如下：

Inputx:4↙

y=0.018316

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    main()  {      int x;      double y;        printf("Input x: ");      scanf("%d", &x);        // 输入一个整数      if (x > 0)      {          y = exp(-x);        //如果大于0，计算y=exp(-x)的值      }      else if (x == 0)      {          y = 1;              //x=0，则y=1      }      else      {          y = -exp(x);        //x<0，则y=-exp(x)      }      printf("y=%f\n", y);  } |

Q452.(10分)编写一个程序完成输入一个整数，输出它的符号。

\*\*输入格式要求："%d" 提示信息："Enter n:"

\*\*输出格式要求："+\n" "-\n" "this number has no sign\n"

程序运行示例1如下：

Enter n:90

+

程序运行示例2如下：

Enter n:0

this number has no sign

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int n;      printf("Enter n:");      scanf("%d", &n);      if (n > 0)      {          printf("+\n");      }      else if (n < 0)      {          printf("-\n");      }      else      {          printf("this number has no sign\n");      }  } |

Q1723.(10分)编程输入三角形的三条边a、b、c，判断它们能否构成三角形。若能构成三角形，指出是何种三角形：等腰三角形、等边三角形、直角三角形，还是一般三角形？

输入提示信息："Input a,b,c:"

输入格式："%f,%f,%f"

运行示例：

① Input a,b,c:3,4,5↙

直角三角形

② Input a,b,c:4,4,5↙

等腰三角形

③ Input a,b,c:3,4,6↙

一般三角形

④ Input a,b,c:3,4,9↙

不是三角形

⑤ Input a,b,c:10,10,14.14↙

等腰直角三角形

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  #define EPS  1e-1  main()  {      float a, b, c;      int flag = 1;                            /\* 置标志变量flag为非0值 \*/      printf("Input a,b,c:");      scanf("%f,%f,%f", &a, &b, &c);      if (a + b > c && b + c > a && a + c > b)    /\* 如果满足三角形的基本条件 \*/      {          if (fabs(a - b) <= EPS || fabs(b - c) <= EPS || fabs(c - a) <= EPS)          {              printf("等腰");           /\* 等腰 \*/              flag = 0;                  /\* 置标志变量flag为0值 \*/          }          if (fabs(a \* a + b \* b - c \* c) <= EPS   || fabs(a \* a + c \* c - b \* b) <= EPS                  || fabs(c \* c + b \* b - a \* a) <= EPS)          {              printf("直角");           /\* 直角 \*/              flag = 0;                  /\* 置标志变量flag为0值 \*/          }          if (flag)                       /\* 若标志变量flag非0，则是一般三角形 \*/          {              printf("一般");          }          printf("三角形\n");      }      else                                  /\* 如果不满足三角形的基本条件 \*/      {          printf("不是三角形\n");      }  } |

Q1300.(10分)企业发放的奖金根据利润提成。利润（I）低于或等于10万元时，奖金可提10%；利润高于10万元，低于20万（100000<I<=200000）元时，低于10万元的部分按10%提成，高于10万元的部分，可提成7.5%；200000<I<=400000时，高于20万元的部分按5%提成；400000<I<=600000时，高于40万元的部分按3%提成；600000<I<=1000000时，高于60万的部分按1.5%提成；I>1000000时，超过100万的部分按1%提成。从键盘输入当月利润I，求应发放奖金总数。要求：用if语句编程序实现。

\*\*输入格式要求："%ld" 提示信息："请输入利润："

\*\*输出格式要求："奖金是%10.2f"

程序的运行示例如下：

请输入利润：234000

奖金是 59500.00

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      long i;      float bonus, bon1, bon2, bon4, bon6, bon10;        /\* 初始化变量 \*/      bon1 = 100000 \* 0.1;      bon2 = bon1 + 100000 \* 0.075;      bon4 = bon2 + 200000 \* 0.05;      bon6 = bon4 + 200000 \* 0.03;      bon10 = bon6 + 400000 \* 0.015;      printf("请输入利润：");      scanf("%ld", &i);      /\* 计算 \*/      if (i <= 1e5)          bonus = i \* 0.1;      else if (i <= 2e5)          bonus = bon1 + (i - 100000) \* 0.075;      else if (i <= 4e5)          bonus = bon2 + (i - 200000) \* 0.05;      else if (i <= 6e5)          bonus = bon4 + (i - 400000) \* 0.03;      else if (i <= 1e6)          bonus = bon6 + (i - 600000) \* 0.015;      else          bonus = bon10 + (i - 1000000) \* 0.01;      printf("奖金是%10.2f", bonus);        return 0;  } |

Q1134.(10分)下面程序用于输入三角形的三条边a,b,c，判断它们能否构成三角形，若能则指出是何种三角形：等腰三角形、直角三角形、一般三角形。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

#define LIMIT = 1e-1;

main()

{

float a, b, c;

int flag = 1;

scanf("%d, %d, %d", a, b , c);

if (a + b > c && b + c > a && a + c > b)

{

if (fabs(a - b) <= LIMIT | fabs(b - c) <= LIMIT | fabs(c - a) <= LIMIT)

{

printf("等腰");

flag = 0;

}

else if (fabs(a \* a + b \* b - c \* c) <= LIMIT

&& fabs(a \* a + c \* c - b \* b) <= LIMIT

&& fabs(c \* c + b \* b - a \* a) <= LIMIT)

{

printf("直角");

flag = 0;

}

if (!flag)

{

printf("一般");

}

printf("三角形\n");

}

else

{

printf("不是三角形\n");

}

}

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    #define LIMIT 1e-1    main()  {      float   a, b, c;      int     flag = 1;        scanf("%f, %f, %f", &a, &b , &c);      if (a + b > c && b + c > a && a + c > b)      {          if (fabs(a - b) <= LIMIT || fabs(b - c) <= LIMIT || fabs(c - a) <= LIMIT)          {              printf("等腰");              flag = 0;          }          if (fabs(a \* a + b \* b - c \* c) <= LIMIT                  || fabs(a \* a + c \* c - b \* b) <= LIMIT                  || fabs(c \* c + b \* b - a \* a) <= LIMIT)          {              printf("直角");              flag = 0;          }          if (flag)          {              printf("一般");          }          printf("三角形\n");      }      else      {          printf("不是三角形\n");      }  } |

Q344.(10分)编程判断输入整数x的正负性和奇偶性。

\*\*输入格式要求："%d" 提示信息："Enter a integer:"

\*\*输出格式要求："The integer is a plus even number\n" "This integer is a plus odd number\n"

"This integer is a minus even number\n" "This integer is a minus odd number\n"

"This integer is zero.\n"

程序运行示例如下：

Enter a integer:-9

This integer is a minus odd number

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int x;      printf("Enter a integer:");      scanf("%d",&x);      if(x!=0)      {          if(x>0)          {              x=x%2;              if(x==0) printf("The integer is a plus even number\n");              else printf("This integer is a plus odd number\n");          }          else          {              x=x%2;              if(x==0) printf("This integer is a minus even number\n");              else printf("This integer is a minus odd number\n");          }      }      else printf("This integer is zero.\n");  } |

Q330.(10分)输入年份year和月month，求该月有多少天。判断是否为闰年，可用如下C语言表达式：year%4==0&&year%100!=0||year%4==0。若表达式成立，则year为闰年；否则，表达式不成立，为平年。

\*\*输入格式要求："%d%d" 提示信息："Enter year and month:"

\*\*输出格式要求："%d,%d is %d days\n"

程序运行示例如下：

Enter year and month:2012 11

2012,11 is 30 days

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int year,month,days;      printf("Enter year and month:");      scanf("%d%d",&year,&month);      switch(month)      {          case 1: case 3: case 5: case 7:          case 8: case 10: case 12:              days=31;break;          case 4: case 6: case 9: case 11:              days=30;break;          case 2:              if(year%4==0&&year%100!=0||year%4==0)                  days=29;              else days=28;              break;          default:              printf("Input error\n");              days=0;      }      if(days!=0)          printf("%d,%d is %d days\n",year,month,days);  } |

Q579.(10分)输入某年某月某日，判断这一天是这一年的第几天？程序分析：以3月5日为例，应该先把前两个月的加起来，然后再加上5天即本年的第几天，特殊情况，闰年且输入月份大于3时需考虑多加一天。

\*\*输入格式要求："%d,%d,%d" 提示信息："\nplease input year,month,day\n"

\*\*输出格式要求："It is the %dth day." "data error"

程序运行示例如下：

please input year,month,day

2011,11,22

It is the 327th day.

### 参考答案

|  |
| --- |
| int main()  {      int day, month, year, sum, leap;      printf("\nplease input year,month,day\n");      scanf("%d,%d,%d", &year, &month, &day);      switch (month)/\*先计算某月以前月份的总天数\*/      {      case 1:          sum = 0;          break;      case 2:          sum = 31;          break;      case 3:          sum = 59;          break;      case 4:          sum = 90;          break;      case 5:          sum = 120;          break;      case 6:          sum = 151;          break;      case 7:          sum = 181;          break;      case 8:          sum = 212;          break;      case 9:          sum = 243;          break;      case 10:          sum = 273;          break;      case 11:          sum = 304;          break;      case 12:          sum = 334;          break;      default:          printf("data error");          break;      }      sum = sum + day;      /\*再加上某天的天数\*/      if (year % 400 == 0 || (year % 4 == 0 && year % 100 != 0))          /\*判断是不是闰年\*/          leap = 1;      else          leap = 0;      if (leap == 1 && month > 2)/\*如果是闰年且月份大于2,总天数应该加一天\*/ sum++;      printf("It is the %dth day.", sum);  } |

Q462.(10分)程序：计算股票经纪人的佣金

当股票通过经纪人进行买卖时，经纪人的佣金往往根据股票交易额采用某种变化的比例进行计算。下面的表格显示了实际支付给经纪人的费用的数量：

交易额范围 佣金费用

低于2500美元 30美元+1.7%

2500~6250美元 56美元+0.66%

6250~20000美元 76美元+0.34%

20000~50000美元 100美元+0.22%

50000~500000美元 155美元+0.11%

超过500000美元 255美元+0.09%

最低收费是39美元。下面的程序要求用户输入交易额，然后显示出佣金的数额：

Enter value of trade: 30000

Commission: 166.00∗∗输入格式要求："∗∗输出格式要求："Commission:

%.2f\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      float commission, value;        printf("Enter value of trade:");      scanf("%f", &value);        if (value < 2500.00)          commission = 30.00 + .017 \* value;      else if (value < 6250.00)          commission = 56.00 + .0066 \* value;      else if (value < 20000.00)          commission = 76.00 + .0034 \* value;      else if (value < 50000.00)          commission = 100.00 + .0022 \* value;      else if (value < 500000.00)          commission = 155.00 + .0011 \* value;      else          commission = 255.00 + .0009 \* value;        if (commission < 39.00)          commission = 39.00;        printf("Commission: $%.2f\n", commission);        return 0;  } |

Q1347.(10分)计算两个正整数的最大公约数。请按以下给定的函数原型编程：

int MaxCommonFactor(int a, int b);

返回值：返回的是最大公约数；若输入的数据有任意一个不满足条件，返回值是-1。

程序的运行示例1如下：

45,155 （输入，用,隔开）

5 （输出）

程序的运行示例2如下：

-9,20

-1

### 参考答案

|  |
| --- |
| #include <stdio.h>    int MaxCommonFactor(int a, int b);    int main()  {      int x, y;      scanf("%d,%d", &x, &y);      printf("%d", MaxCommonFactor(x, y));      return 0;  }    int MaxCommonFactor(int a, int b)  {      int min = a, i;      if (a <= 0 || b <= 0)          return -1;      else      {          if (b < a)          {              min = b;          }          for (i = min; i >= 1; i--)          {              if (a % i == 0 && b % i == 0)                  return i;          }      }  } |

Q1246.(10分)

|  |
| --- |
| 下面程序用于输入三角形的三条边a,b,c，判断它们能否构成三角形，若能则指出是何种三角形：等腰三角形、直角三角形、一般三角形。（找出其中的3处错误并改正之）  #include <stdio.h>  #include <math.h>    main()  {      float   a, b, c;      int flag = 1;        scanf("%d,%d,%d", &a, &b , &c);      if (a+b>c && b+c>a && a+c>b)      {      if (fabs(a-b)<=0.1||fabs(b-c)<=0.1||fabs(c-a)<=0.1)      {          printf("等腰三角形\n");          flag = 0;      }      else if (fabs(a\*a+b\*b-c\*c)<=0.1                  ||fabs(a\*a+c\*c-b\*b)<=0.1                  ||fabs(c\*c+b\*b-a\*a)<=0.1)      {          printf("直角三角形\n");          flag = 0;      }      if (!flag)      {          printf("一般三角形\n");      }      }      else      {      printf("不是三角形\n");      }  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    main()  {      float   a, b, c;      int flag = 1;        scanf("%f, %f, %f", &a, &b , &c);   //原来%d错      if (a+b>c && b+c>a && a+c>b)      {      if (fabs(a-b)<=0.1||fabs(b-c)<=0.1||fabs(c-a)<=0.1)      {          printf("等腰三角形\n");          flag = 0;      }      if (fabs(a\*a+b\*b-c\*c)<=0.1                  ||fabs(a\*a+c\*c-b\*b)<=0.1                  ||fabs(c\*c+b\*b-a\*a)<=0.1) //原来多了else      {          printf("直角三角形\n");          flag = 0;      }      if (flag)  //原来flag前面多了!      {          printf("一般三角形\n");      }      }      else      {      printf("不是三角形\n");      }  } |

Q328.(10分)在屏幕上显示一张时间表，操作人员根据提示进行选择，程序根据输入的时间序号显示相应的问候信息。

程序运行示例1如下：

\*\*\*\*TIME\*\*\*\*

1.Morinig

2.afternoon

3.night

Enter your choice:2

Good afternoon

程序运行示例2如下：

\*\*\*\*TIME\*\*\*\*

1.Morinig

2.afternoon

3.night

Enter your choice:4

Selection wrong

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char ch;      printf("\*\*\*\*TIME\*\*\*\*\n");      printf("1.Morinig\n");      printf("2.afternoon\n");      printf("3.night\n");      printf("Enter your choice:");      ch=getchar();      switch(ch)      {          case '1': printf("\nGood morning\n");break;          case '2': printf("\nGood afternoon\n");break;          case '3': printf("\nGood night\n");break;          default: printf("Selection wrong\n");      }  } |

Q586.(10分)编写一个程序，输入年份和月份，判断该年是否是闰年，并根据给出的月份判断是什么季节和该月有多少天？（闰年的条件是年份能被4整除但不能被100整除，或者能被400整除；规定3~5月为春季，6~8月为夏季，9~11月为秋季，1、2和12月为冬季）。

\*\*输入格式要求："%d,%d" 提示信息："Please enter year,month:"

\*\*输出格式要求："%d is leap year\n" "%d is not leap year\n" "The season is spring/summer/autumn/winter" "The number of days of this month is %d\n"

程序运行示例如下：

Please enter year,month:2012,11

2012 is leap year

The season is autumn

The number of days of this month is 30

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      int y, m, leap, season, days;      printf("Please enter year,month:");      scanf("%d,%d", &y, &m);      if ((y % 4 == 0 && y % 100 != 0) || (y % 400 == 0))          leap = 1;      else          leap = 0;      if (m >= 3 && m <= 5)          season = 1;      else if (m >= 6 && m <= 8)          season = 2;      else if (m >= 9 && m <= 11)          season = 3;      else          season = 4;      switch (m)      {      case 1:      case 3:      case 5:      case 7:      case 8:      case 10:      case 12:          days = 31;          break;      case 4:      case 6:      case 9:      case 11:          days = 30;          break;      case 2:          if (leap == 1)              days = 29;          else              days = 28;      }      printf("%d %s leap year\n", y, (leap == 1 ? "is" : "is not"));      printf("The season is ");      switch (season)      {      case 1:          printf("spring\n");          break;      case 2:          printf("summer\n");          break;      case 3:          printf("autumn\n");          break;      case 4:          printf("winter\n");      }      printf("The number of days of this month is %d\n", days);      return 0;  } |

Q1130.(10分)比较两个字符的大小，按由小到大输出。运行时输入数据：34↙

请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

char t, c1, c2;

getchar(c1), getchar(c2);

if (c1 > c2)

t = c1;

c1 = c2;

c2 = t;

printf("%c,%c", c1, c2);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      char t, c1, c2;        c1 = getchar();      c2 = getchar();      if (c1 > c2)      {          t = c1;          c1 = c2;          c2 = t;      }      printf("%c,%c", c1, c2);  } |

Q1

328.(10分)定义一个结构体变量（包括年、月、日）。计算该日在本年中是第几天？注意闰年问题。

程序的运行示例如下：

请输入日期（年，月，日）

1990,2,14

2月14日是1990年的第45天。

### 参考答案

|  |
| --- |
| #include <stdio.h>    struct  {      int year;      int month;      int day;  } date;    int main(void)  {      int days;        printf("请输入日期（年，月，日）\n");      scanf("%d,%d,%d", &date.year, &date.month, &date.day);      switch (date.month)      {      case 1:          days = date.day;          break;      case 2:          days = date.day + 31;          break;      case 3:          days = date.day + 59;          break;      case 4:          days = date.day + 90;          break;      case 5:          days = date.day + 120;          break;      case 6:          days = date.day + 151;          break;      case 7:          days = date.day + 181;          break;      case 8:          days = date.day + 212;          break;      case 9:          days = date.day + 243;          break;      case 10:          days = date.day + 273;          break;      case 11:          days = date.day + 304;          break;      case 12:          days = date.day + 334;          break;      }      if ((date.year % 4 == 0 && date.year % 100 != 0              || date.year % 400 == 0) && date.month >= 3)      {          days += 1;      }      printf("\n%d月%d日是%d年的第%d天。", date.month, date.day, date.year, days);        return 0;  } |

Q483.(10分)将百分制成绩转换为ABCDE五个等级（假设成绩都为整数值）。

\*\*输入格式要求："%d" 提示信息："input the score :"

\*\*输出格式要求："score=%d is error!\n" "score=%d-->A\n"

程序示例运行1如下：

input the score :90

score=90-->A

程序示例运行2如下：

input the score :50

score=50-->E

程序示例运行3如下：

input the score :76

score=76-->C

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int score;        printf("input the score :");      scanf("%d", &score);        if (score < 0 || score > 100)          printf("score=%d is error!\n", score);      else if (score >= 90)          printf("score=%d-->A\n", score);      else if (score >= 80)          printf("score=%d-->B\n", score);      else if (score >= 70)          printf("score=%d-->C\n", score);      else if (score >= 60)          printf("score=%d-->D\n", score);      else          printf("score=%d-->E\n", score);        return 0;  } |

Q1202.(10分)编程判断输入整数的正负性和奇偶性。

\*\*输入格式要求："%d" 提示信息："Input m:"

\*\*输出格式要求：

"%d is a negative even\n"

"%d is a negative odd\n"

"%d is zero. It is an even\n"

程序的5次运行示例如下：

① Input m:6↙

6 is a positive even

② Input m:-7↙

6 is a negative odd

③ Input m:-6↙

-6 is a negative even

④ Input m:7↙

7 is a positive odd

⑤ Input m:0↙

0 is zero. It is an even

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int m;      printf("Input m: ");      scanf("%d", &m);        //输入一个整数      if (m > 0)               //是否为正数      {          if (m % 2 == 0)     //是正数，且能被2整除，则是正偶数          {              printf("%d is a positive even\n", m);          }          else                    //不能被2整除，则是正奇数          {              printf("%d is a positive odd\n", m);          }      }      else if (m < 0)          //判断是否为负数      {          if (m % 2 == 0)          {              printf("%d is a negative even\n", m);   //是负偶数          }          else          {              printf("%d is a negative odd\n", m);    //是负奇数          }      }      else      {          printf("%d is zero. It is an even\n", m);      }  } |

Q482.(10分)输入年份，判断该年是否为闰年。

\*\*输入格式要求："%d" 提示信息："输入年份:"

\*\*输出格式要求："%d是闰年!\n" "%d不是闰年!\n"

程序示例运行1如下：

输入年份: 2012

2012是闰年!

程序示例运行2如下：

输入年份: 2011

2011不是闰年!

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int year;        printf("输入年份:");      scanf("%d", &year);        if ((year % 4 == 0 && year % 100 != 0)              || year % 400 == 0)          printf("%d是闰年!\n", year);      else          printf("%d不是闰年!\n", year);        return 0;  } |

Q121.(30分)编程计算分段：

y= x x<1

y= 2x-1 1<=x<10

y= 3x-11 x>=10

从键盘输入一个单精度实数x，打印出y值。

\*\*输入提示信息格式要求为："Please input x:"；

\*\*输出格式要求为"y = %.2f\n"。

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {     float x,y;     printf("Please input x:");     scanf("%f",&x);  //1     if(x<1)   //1        y=x;     else if(x>=1 && x<10)  //1        y=2\*x-1;     else if(x>=10)    //1        y=3\*x-11;     printf("y = %.2f\n",y);  //1  } |

Q329.(10分)简单计算器。请编写一个程序计算表达式：data1 op data2的值。其中op为运算符+-\*/。

\*\*输入格式要求："%f%c%f" 提示信息："Enter your expression:"

\*\*输出格式要求："%.2f+%.2f=%.2f\n" "%.2f-%.2f=%.2f\n" "%.2f\*%.2f=%.2f\n" "%.2f/%.2f=%.2f\n"

"Division by zero\n" "Unknown operator.\n"

程序运行示例如下：

Enter your expression:6\*8

6.00\*8.00=48.00

程序运行示例如下：

Enter your expression:9/0

Division by zero

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      float data1,data2;      char op;      printf("Enter your expression:");      scanf("%f%c%f",&data1,&op,&data2);      switch(op)      {          case '+':              printf("%.2f+%.2f=%.2f\n",data1,data2,data1+data2); break;          case '-':              printf("%.2f-%.2f=%.2f\n",data1,data2,data1-data2); break;          case '\*':              printf("%.2f\*%.2f=%.2f\n",data1,data2,data1\*data2); break;          case '/':              if(data2==0)                  printf("Division by zero\n");              else                  printf("%.2f/%.2f=%.2f\n",data1,data2,data1/data2);              break;          default:              printf("Unknown operator.\n");      }  } |

Q481.(10分)编程求两个整数中的最大值。

\*\*输入格式要求："%d%d" 提示信息："input the value of x and y:"

\*\*输出格式要求："The max of %d and %d is %d\n"

程序示例运行如下：

input the value of x and y:3 4

The max of 3 and 4 is 4

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int x, y;        printf("input the value of x and y:");      scanf("%d%d", &x, &y);        if (x >= y)          printf("The max of %d and %d  is %d\n", x, y, x);      else          printf("The max of %d and %d  is %d\n", x, y, y);        return 0;  } |

Q484.(10分)输入三角形的三条边a，b，c，判断它们能否构成三角形。若能构成三角形，指出是何种三角形（等腰三角形、直角三角形、一般三角形）。

\*\*输入格式要求："%f,%f,%f" 提示信息："Input the three edge length: "

\*\*输出格式要求："等腰三角形" "直角三角形" "一般三角形" "不是三角形"

程序运行示例如下：

① Input the three edge:3,4,5↙

直角三角形

② Input the three edge:4,4,5↙

等腰三角形

③ Input the three edge:10,10,14.14↙

等腰直角三角形

④ Input the three edge:3,4,9↙

不是三角形

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    #define EPS  1e-1    main()  {      float a, b, c;      int flag = 1;      printf("Input the three edge length: ");      scanf("%f,%f,%f", &a, &b, &c);      //输入三角形的三条边      if (a + b > c && b + c > a && a + c > b) //三角形的基本条件      {          if (fabs(a - b) <= EPS || fabs(b - c) <= EPS || fabs(c - a) <= EPS)          {              printf("等腰");              flag = 0;          }          if (fabs(a \* a + b \* b - c \* c) <= EPS || fabs(a \* a + c \* c - b \* b) <= EPS || fabs(c \* c + b \* b - a \* a) <= EPS)          {              printf("直角");              flag = 0;          }          if (flag)          {              printf("一般");          }          printf("三角形\n");      }      else      {          printf("不是三角形\n");      }  } |

Q463.(10分)

程序：显示法定格式的日期合同和其他法律文档中经常使用下列日期格式：

Dated this    day of    ,19  .

编写程序用来显示这种格式的日期。用户以月/日的格式录入日期，然后计算机显示出“法定”格式的日期：

Enter date (mm/dd/yy) : 7/19/96

Dated this 19th day of July, 1996.

可以使用printf函数实现主要的格式化。然而，还有两个问题：如何为日添加“th”（或者“st”、“nd”、“rd”），以及如何使用单词代替数字显示月份。幸运的是，switch语句可以很好地解决这两种问题：用一个switch语句负责显示日的后缀，再用另一个switch语句显示出月份名。

\*\*输入格式要求："%d/%d/%d" 提示信息："Enter date (mm/dd/yy):" "Dated this %d" "st/nd/rd/th" " day of " "January/February/.../December"  
\*\*输出格式要求："%10d%10d\n" ", 19%.2d.\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int month, day, year;        printf("Enter date (mm/dd/yy):");      scanf("%d/%d/%d", &month, &day, &year);        printf("Dated this %d", day);      switch (day)      {      case 1:      case 21:      case 31:          printf("st");          break;      case 2:      case 22:          printf("nd");          break;      case 3:      case 23:          printf("rd");          break;      default:          printf("th");          break;      }        printf(" day of ");        switch (month)      {      case 1:          printf("January");          break;      case 2:          printf("February");          break;      case 3:          printf("March");          break;      case 4:          printf("April");          break;      case 5:          printf("May");          break;      case 6:          printf("June");          break;      case 7:          printf("July");          break;      case 8:          printf("August");          break;      case 9:          printf("September");          break;      case 10:          printf("October");          break;      case 11:          printf("November");          break;      case 12:          printf("December");          break;      }        printf(", 19%.2d.\n", year);        return 0;  } |

Q1729.(10分)从键盘任意输入一个年号，判断它是否是闰年。

若是闰年，输出“Yes”，否则输出“No”。

已知符合下列条件之一者是闰年：

1）能被4整除，但不能被100整除;

2）能被400整除。

\*\*提示信息格式要求: "Input year:\n"

\*\*输入数据格式要求: 无

\*\*输出格式数据要求:

如果是闰年输出: "Yes\n"

否 则 输 出: "No\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int year;      printf("Input year:\n");      scanf("%d",&year);                  //1        if((year%4 == 0 && year%100 != 0) || year % 400 == 0)  //1          printf("Yes\n");                //1      else                              //1          printf("No\n");                //1  } |

Q345.(10分)求两个整数中的较大值。

\*\*输入格式要求："%d%d" 提示信息："input a and b="

\*\*输出格式要求："The max of a and b :%d\n"

程序运行示例如下：

input a and b=5 6

The max of a and b :6

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a,b,max;      printf("input a and b=");      scanf("%d%d",&a,&b);      if(a>b) max=a;      else max=b;      printf("The max of a and b :%d\n",max);  } |

Q327.(10分)通过键盘输入字符，将输入的字符分为控制、数字、大写字母、小写字母和其他字符等五类。

\*\*输入格式要求："%c" 提示信息："Enter a charactor:"

\*\*输出格式要求："\nThe charactor is a control charactor.\n"

"\nThe charactor is a digit charactor.\n" "\nThe charactor is a capital charactor.\n"

"\nThe charactor is a lower charactor.\n" "\nThe charactor is a other charactor.\n"

程序运行示例1如下：

Enter a charactor:1

The charactor is a digit charactor.

程序运行示例2如下：

Enter a charactor:a

The charactor is a lower charactor.

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char c;      printf("Enter a charactor:");      c=getchar();      if(c<0x20) printf("\nThe charactor is a control charactor.\n");      else if(c>='0'&&c<='9')          printf("\nThe charactor is a digit charactor.\n");      else if(c>='A'&&c<='Z')          printf("\nThe charactor is a capital charactor.\n");      else if(c>='a'&&c<='z')          printf("\nThe charactor is a lower charactor.\n");      else          printf("\nThe charactor is a other charactor.\n");  } |

Q1401.(10分)请编程计算居民应交水费，并提供各种测试数据。居民应交水费y（元）与月用水量x（吨）的函数关系式如下：

y=f(x)=⎧ ⎩ ⎨ 0, 4x 3  , 2.5x−10.5,  x<0 0≤x≤15 x>15

\*\*输入格式要求：信息提示："Enter x:" 输入格式："%lf"

\*\*输出格式要求："f(%.2f)=%.2f\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      double x, y;           /\*定义2个双精度浮点型变量\*/      printf("Enter x:");       /\*输入提示\*/      scanf("%lf", &x);       /\*输入double型数据用%lf\*/      if (x < 0)      {          y = 0;      /\*满足x<0\*/      }      else if (x <= 15)      {          y = 4 \* x / 3; /\*不满足x<0，但满足 ,即满足 \*/      }      else      {          y = 2.5 \* x - 10.5; /\*即不满足x<0,也不满足 ,即满足x>15\*/      }      printf("f(%.2f)=%.2f\n", x, y);      return 0;  } |

Q195.(30分)编程计算: 1!+3!+5!+...+(2n-1)!，要求阶乘计算调用fun函数实现，

数据输入及打印结果在主函数实现。阶乘计算fun函数原型为:

long fun(int m);

参数说明:

参 数: m是要进行阶乘计算的数；

返回值: 数m的阶乘

\*\*\*\*输入提示信息格式: "Input n:\n"

\*\*\*\*输入数据格式要求: %d

\*\*\*\*输出数据格式要求: "sum=%ld\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

### 参考答案

|  |
| --- |
| #include<stdio.h>  long fun(int m);  main()  {      int i,n;      long sum=0,p;           //1        printf("Input n:\n");      scanf("%d",&n);         //1        for(i=1;i<=2\*n-1;i+=2)       //2      {          p=fun(i);       //2          sum=sum+p;      //1      }      printf("sum=%ld\n",sum);    //1    }  long fun(int m)  {      long p=1;           //1      int i;        for(i=1;i<=m;i++)        //2      {          p=p\*i;          //1      }        return p;           //2  } |

Q571.(10分)请按如下函数原型编写计算1～n之间的所有素数之和的程序。

int IsPrime(int x)

在主函数中调用该函数计算，并输出1～n之间的所有素数之和

\*\*要求输入提示信息为:无

\*\*要求输入格式为: "%d"

\*\*要求输出格式为："%d"

程序运行示例如下：

10

17

### 参考答案

|  |
| --- |
| #include<stdio.h>  #include<math.h>  int IsPrime(int x);  int main()  {      int m = 3;      int sum = 2;      int n;      scanf("%d", &n);      for (m = 3; m <= n; m++)      {          if (IsPrime(m))          {              sum += m;          }      }      printf("%d", sum);      return 0;  }  int IsPrime(int x)  {      int i, flag = 1;      int squareRoot = (int)sqrt(x);      if (x <= 1)      {          flag = 0;      }      for (i = 2; i <= squareRoot && flag; i++)      {          if (x % i == 0)          {              flag = 0;          }      }      return flag;  } |

Q1114.(10分)下列给定程序中，函数fun的功能是：判断m是否为素数，若是返回1，否则返回0。主函数的功能是：按每行5个输出1-100之间的全部素数。请改正程序中的错误，使它能得出正确的结果。

#include <math.h>

#include <stdio.h>

main()

{

int m, k = 0;

for (m = 1; m <= 100; m++)

if (fun(m) == 1)

{

printf("%4d", m);

k++;

if (k % 5 == 0)

printf("\n");

}

}

void fun(int n)

{

int i, k = 1;

if (m <= 1) k = 0;

for (i = 1; i < m; i++)

if (m % i = 0) k = 0;

return m;

}

### 参考答案

|  |
| --- |
| #include <math.h>  #include <stdio.h>    main()  {      int m, k = 0;      for (m = 1; m <= 100; m++)          if (fun(m) == 1)          {              printf("%4d", m);              k++;              if (k % 5 == 0)                  printf("\n");          }  }    int fun(int n)  {      int i, k = 1;      if (n <= 1) k = 0;      for (i = 2; i <= sqrt(n); i++)          if (n % i == 0) k = 0;      return k;  } |

Q380.(10分)将任一整数转换为二进制形式。

\*\*输入格式要求："%d" 提示信息："Input number:"

\*\*输出格式要求："number of decimal form:%d\n" " it\'s binary form: "

程序运行示例如下：

Input number:876

number of decimal form:876

it's binary form: 00000000000000000000001101101100

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int x;      printf("Input number:");      scanf("%d",&x);      printf("number of decimal form:%d\n",x);      printf("    it\'s binary form: ");      printb(x,sizeof(int)\*8);      putchar('\n');  }  printb(int x,int n)  {      if(n>0)      {          putchar('0'+((unsigned)(x&(1<<(n-1)))>>(n-1)));          printb(x,n-1);      }  } |

Q447.(10分)求1898

现將不超过2000的所有素数从到大排成第一行，第二行上的每个数都等于它“右肩”上的素数与“左肩”上的素数之差。请编程求出：第二行数中是否存在这样的若干个连续的整数，它们的和恰好是1898？假如存在的话，又有几种这样的情况？

\*\*输入格式要求：提示信息："There are follwing primes in first row :\n"

\*\*输出格式要求："(%d).%3d,......,%d\n"

程序运行示例如下：

There are follwing primes in first row :

(1).101,......,1999

(2). 89,......,1987

(3). 53,......,1951

(4). 3,......,1901

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  #define NUM 320  int number[NUM];  main()  {      int i,j,count=0;      printf("There are follwing primes in first row :\n");      for(j=0,i=3;i<=1999;i+=2)          if (fflag(i)) number[j++]=i;      for(j--;number[j]>1898;j--)      {          for(i=0;number[j]-number[i]>1898;i++);          if(number[j]-number[i]==1898)              printf ("(%d).%3d,......,%d\n",++count,number[i],number[j]);      }  }  fflag(i)  int i;  {      int j;      if (i<=1) return(0);      if (i==2)  return(1);      if(!(i%2)) return(0);      for(j=3;j<=(int)( sqrt((double)i)+1);j+=2)          if (!(i%j))  return(0);      return(1);  } |

Q351.(10分)计算2和-3的0到5次幂。

\*\*输出格式要求："2 power %d is %d, -3 power %d is %d\n"

程序运行示例如下：

2 power 0 is 1, -3 power 0 is 1

2 power 1 is 2, -3 power 1 is -3

2 power 2 is 4, -3 power 2 is 9

2 power 3 is 8, -3 power 3 is -27

2 power 4 is 16, -3 power 4 is 81

2 power 5 is 32, -3 power 5 is -243

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i;      for(i=0;i<=5;i++)          printf("2 power %d is %d, -3 power %d is %d\n",i,power(2,i),i,power(-3,i));  }  power(int x,int n)  {      int p;      for(p=1;n>0;n--)          p\*=x;      return p;  } |

Q379.(10分)\*求杨辉三角形。在屏幕上显示如下的杨辉三角形：

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

…… …… …… …… ……

\*\*输入格式要求："%d" 提示信息："N="

\*\*输出格式要求："%6d"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,j,n=13;      printf("N=");      while(n>12)          scanf("%d",&n);      for(i=0;i<=n;i++)      {          for(j=0;j<12-i;j++)              printf(" ");          for(j=1;j<i+2;j++)              printf("%6d",c(i,j));          printf("\n");      }  }  int c(int x,int y)  {      int z;      if(y==1||y==x+1)          return 1;      else          z=c(x-1,y-1)+c(x-1,y);      return z;  } |

Q186.(15分)两个正整数的最大公约数是能够整除这两个整数的最大整数。

请按如下函数原型编写计算最大公约数的函数Gcd()。

int Gcd(int a, int b);

在主函数中调用该函数计算，并输出从键盘任意输入的两整数的最大公约数。

\*\*要求输入提示信息为: "Input a,b:\n"

\*\*要求输入格式为: "%d,%d"

\*\*要求输出格式为

如果a或b不是正整数，输出: "Input number should be positive!\n"

否则输出: "Greatest Common Divisor of %d and %d is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

### 参考答案

|  |
| --- |
| #include <stdio.h>  int Gcd(int a, int b);  main()  {      int a, b, c;      printf("Input a,b:\n");      scanf("%d,%d", &a, &b); //1      c = Gcd(a, b);  //1      if (c != -1) //1      {          printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c); //1      }      else      {          printf("Input number should be positive!\n"); //1      }  }  int Gcd(int a, int b) //1  {      int i, t;      if (a <= 0 || b <= 0) //1      {          return -1; //1      }      t = a < b ? a : b;  //2      for (i = t; i > 0; i--)//2      {          if (a % i == 0 && b % i == 0)//1             return i;  //1      }      return 1; //1  } |

Q155.(15分)从键盘任意输入一个整数n（如果n<0，则要求提示"Input Error!\n"）,用函数编程计算这个整数的阶乘值。

要求按如下函数原型编程。

long fac(int n);

在主函数中调用fac计算n!。

要求必须用函数编程，否则不给分。

\*\*输入提示信息格式要求为："Input an integer:\n"；

\*\*输出格式要求为"%d! = %ld\n"，其中第一个整数为输入数值，第二个整数位其阶乘。

### 参考答案

|  |
| --- |
| #include <stdio.h>  long fac(int n);  main()  {      int n;      long res;      printf("Input an integer:\n");      scanf("%d",&n);                       //1      if (n < 0) printf("Input Error!\n");      else      {      res = fac(n);                      //1      printf("%d! = %ld\n", n, res);       }  }  long fac(int n)  {      int i;      long res = 1;             //1      for(i=1; i<=n; i++)              //1      {          res \*= i;                      //1      }      return res;  } |

Q1131.(10分)求1+2+...+n之和。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

unsigned long fun(int n);

main()

{

int n;

unsigned long sum = 0;

printf("Input n:");

scanf("%d", &n);

while (n)

{

sum = fun(n--);

}

printf("The sum is :%u", sum);

}

unsigned long fun(int n)

{

unsigned long m\_sum = 0;

m\_sum += n;

return m\_sum;

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    unsigned long fun(int n);    main()  {      int n;      unsigned long  sum = 0;      printf("Input n:");      scanf("%d", &n);      while (n)      {          sum = fun(n--);      }      printf("The sum is :%u", sum);  }    unsigned long fun(int n)  {      static unsigned long  m\_sum = 0;      m\_sum += n;      return m\_sum;  } |

Q306.(15分)编程计算: 1!+3!+5!+...+(2n-1)!，要求阶乘计算调用fun函数实现，

数据输入及打印结果在主函数实现。阶乘计算fun函数原型为:

long fun(int m);

参数说明:

参 数: m是要进行阶乘计算的数；

返回值: 数m的阶乘

\*\*\*\*输入提示信息格式: "Input n:\n"

\*\*\*\*输入数据格式要求: %d

\*\*\*\*输出数据格式要求: "sum=%ld\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

### 参考答案

|  |
| --- |
| #include<stdio.h>  long fun(int m);  main()  {      int i,n;      long sum=0,p;           //1        printf("Input n:\n");      scanf("%d",&n);         //1        for(i=1;i<=2\*n-1;i+=2)       //2      {          p=fun(i);       //2          sum=sum+p;      //1      }      printf("sum=%ld\n",sum);    //1    }  long fun(int m)  {      long p=1;           //1      int i;        for(i=1;i<=m;i++)        //2      {          p=p\*i;          //1      }        return p;           //2  } |

Q1354.(30分)数值151是一个回文素数。因为151既是一个素数也是一个回文数（回文数是从前向后读和从后向前读都一样的数）。写一个程序找出所有在[a,b]区间内的回文素数(5 <= a < b < 1000)。要求判素数调用primer函数实现，判回文数调用huiwen函数实现。

素数primer函数原型为:

int primer(int number)

参数说明:

参 数: number是要进行判断的数；

回文数huiwen函数原型为:

int huiwen(int number)

参数说明:

参 数: number是要进行判断的数；

输入

第一行：两个整数，a和b

输出

符合条件的回文素数，一个数一行。

输入样例

5 500

输出样例

5

7

11

101

131

151

181

191

313

353

373

383

### 参考答案

|  |
| --- |
| #include <math.h>  #include <stdio.h>    // Judge if the palindromes that have been found are primes  int primer(int number)  {      int i;      int sqrt\_number;      sqrt\_number = sqrt(number);      for (i = 2; i <= sqrt\_number; i++)      {          if ((number % i) == 0)              return 0; //The number is not prime      }      return 1; // The number is prime  }    int huiwen(int number)  {      int g,s,b;      g=number%10;      s=number/10%10;      b=number/100;      if (number>=10&&number<100)      {   if(g!=s)  return 0; }//The number is not huiwen      else          if (number>=100&&number<1000)          {   if(g!=b)  return 0; }//The number is not huiwen      return 1; // The number is huiwen  }  int main()  {      int a, b; // Enter two numbers as the range of prime palindromes      int i;        scanf("%d %d", &a, &b);        for (i = a; i <= b; i++)          {             if (primer(i) && huiwen(i))                  printf("%d\n", i);          }      return 0;  } |

Q1566.(30分)从键盘输入一个年份，判断是否为闰年。

题目要求：

输入提示： Input a year:

输出要求：

如果是闰年： %d is a leap year!

如果不是闰年： %d is not a leap year!

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int year;      printf("Input a year:");      scanf("%d", &year);      if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))      {          printf("%d is a leap year!\n", year);      }      else printf("%d is not a leap year!\n", year);        return 0;  } |

Q131.(20分)编写一个找出大于给定整数m的最小素数的函数，要求在主函数中输入一个整数m，调用子函数IsPrime()找出其最小素数，并在主函数中输出结果。

函数原型：int IsPrime(int x)

\*\*\*\*要求输入提示信息为：Please input n:

\*\*\*\*输出格式要求为："%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int IsPrime(int x)  {      int i,flag=1;      int squareRoot=sqrt(x);      if(x<=1) //1      {          flag=0;      }      for(i=2; i<=squareRoot&&flag; i++)   //2      {          if(x%i==0)  //1          {              flag=0;          }      }      return flag;  }  main()  {      int n,i;      printf("Please input n:");      scanf("%d",&n);      i=n+1;      while(i>=n)  //1      {          if(IsPrime(i))  //1          {              printf("%d\n",i);   //1              break;          }          else          {              i++;    //1          }      }  } |

Q580.(10分)将一个正整数分解质因数。例如：输入90,打印出90=2\*3\*3\*5。

\*\*输入格式要求："%d" 提示信息："\nplease input a number:\n"

\*\*输出格式要求："%d=%d\*%d"

程序运行示例如下：

please input a number:

90

90=2\*3\*3\*5

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int n, i;        printf("\nplease input a number:\n");      scanf("%d", &n);      printf("%d=", n);      for (i = 2; i <= n; i++)      {          while (n != i)          {              if (n % i == 0)              {                  printf("%d\*", i);                  n = n / i;              }              else              {                  break;              }          }      }      printf("%d", n);  } |

Q1709.(10分)国王的许诺。相传国际象棋是古印度舍罕王的宰相达依尔发明的。舍罕王十分喜欢象棋，决定让宰相自己选择何种赏赐。这位聪明的宰相指着8×8共64格的象棋盘说：陛下，请您赏给我一些麦子吧，就在棋盘的第1个格子中放1粒，第2格中放2粒，第3格中放4粒，以后每一格都比前一格增加一倍，依此放完棋盘上的64个格子，我就感恩不尽了。舍罕王让人扛来一袋麦子，他要兑现他的许诺。请问：国王能兑现他的许诺吗？试编程计算舍罕王共要多少麦子赏赐他的宰相，这些麦子合多少立方米（已知1立方米麦子约1.42e8粒）？

输入格式：无

输出格式：

"sum = %e\n"

"volum = %e\n"

### 参考答案

|  |
| --- |
| #include  <math.h>  #include  <stdio.h>  #define   CONST  1.42e8                    //定义符号常量CONST值为1.42e8  main()  {      int      n;      double  term;      double  sum = 0;                    //累加和变量sum初始化为0      for (n = 1; n <= 64; n++)      {          term = pow(2, n - 1);             //根据累加项的规律计算累加项          sum = sum + term;                   //执行累加运算      }      printf("sum = %e\n", sum);          //打印总麦粒数      printf("volum = %e\n", sum / CONST);    //打印折合的总麦粒体积数  } |

Q570.(10分)有1、2、3、4四个数字，能组成多少个互不相同且无重复数字的三位数（比如：123,124,213...）？都是多少？

\*\*输出格式要求："counter=%d\n" "%d%d%d "

程序运行示例如下：

counter=24

123 124 132 134 142 143 213 214 231 234 241 243 312 314 321 324 341 342 412 413 421 423 431 432

### 参考答案

|  |
| --- |
| #include<stdio.h>  int main()  {      int a, b, c, counter;      counter = 0;      for (a = 1; a < 5; a++)          for (b = 1; b < 5; b++)              for (c = 1; c < 5; c++)              {                  if (a != b && b != c && a != c)                      counter++;              }      printf("counter=%d\n", counter);        for (a = 1; a < 5; a++)          for (b = 1; b < 5; b++)              for (c = 1; c < 5; c++)              {                  if (a != b && b != c && a != c)                      printf("%d%d%d ", a, b, c);              }      return 0;  } |

Q130.(15分)编写程序求出用数字0至9可以组成多少个各位上没有重复数字的三位偶数。

\*\*\*\*要求输入提示信息为：无

\*\*\*\*输出格式要求为：%d\n

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a,b,c,count=0;      for(a=1; a<=9; a++)  //2      {          for(b=0; b<=9; b++)  //2          {              for(c=0; c<=9; c++)  //2              {                  if(a!=b&&a!=c&&b!=c&&c%2==0)    //2                  {                      count++;  //1                  }              }          }      }      printf("%d\n",count);    //1    } |

Q166.(25分)编程实现找出字符串中最大字符元素并输出该元素及其对应的ASCII值.

\*\*\*\*要求输入提示信息为：

"Input a string:\n"

\*\*\*\*输出格式要求为：

"The largest character of \"%s\" is \'%c\' ,The ASCII is %d."

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  main()  {      char string[81];      int i=0,max;      printf("Input a string:");  //1      printf("\n");      gets(string);      max=string[i];      //1      while(string[i]!='\0')      //2      {          i=i+1;          if(string[i]>max)        //2             max=string[i];       //1      }      printf("The largest character of \"%s\" is \'%c\' ,The ASCII is %d.",string,max,max);   //2  } |

Q397.(10分)牛顿迭代法。用牛顿迭代法求x=a  √

，迭代公式为X n+1 =1 2  (X n +a X n   )

，要求迭代的精度满足|X n+1 −X n |<0.00001

。如果迭代20次之后仍未能达到精度要求，也停止计算。

\*\*输入格式要求："%f" 提示信息："Input a=? "

\*\*输出格式要求："\n a=%.6f,x=%.6f,i=%d"

### 参考答案

|  |
| --- |
| #include "stdio.h"  main()  {  float a,x,x0;     int i=0;     do     {  printf("Input a=? ");        scanf("%f",&a);     } while(a<0);     x =a/4;     do     {  i++;        x0=x;        x=(x0+a/x0)/2;     } while(i<20&&(x-x0>=1.e-5||x-x0<=-1.e-5));     printf("\n a=%.6f,x=%.6f,i=%d",a,x,i);  } |

Q346.(10分)输入被除数和除数，求商。

\*\*输入格式要求："%d%d" 提示信息："Enter two numbers:"

\*\*输出格式要求："cannot divide by zero.\n"（被0除） "%d"(输出a/b)

程序运行示例如下：

Enter two numbers:8 3

2

程序运行示例如下：

Enter two numbers:6 3

2

程序运行示例如下：

Enter two numbers:9 0

cannot divide by zero.

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a,b;      printf("Enter two numbers:");      scanf("%d%d",&a,&b);      if(b) printf("%d\n",a/b);      else printf("cannot divide by zero.\n");  } |

Q1354.(30分)数值151是一个回文素数。因为151既是一个素数也是一个回文数（回文数是从前向后读和从后向前读都一样的数）。写一个程序找出所有在[a,b]区间内的回文素数(5 <= a < b < 1000)。要求判素数调用primer函数实现，判回文数调用huiwen函数实现。

素数primer函数原型为:

int primer(int number)

参数说明:

参 数: number是要进行判断的数；

回文数huiwen函数原型为:

int huiwen(int number)

参数说明:

参 数: number是要进行判断的数；

输入

第一行：两个整数，a和b

输出

符合条件的回文素数，一个数一行。

输入样例

5 500

输出样例

5

7

11

101

131

151

181

191

313

353

373

383

### 参考答案

|  |
| --- |
| #include <math.h>  #include <stdio.h>    // Judge if the palindromes that have been found are primes  int primer(int number)  {      int i;      int sqrt\_number;      sqrt\_number = sqrt(number);      for (i = 2; i <= sqrt\_number; i++)      {          if ((number % i) == 0)              return 0; //The number is not prime      }      return 1; // The number is prime  }    int huiwen(int number)  {      int g,s,b;      g=number%10;      s=number/10%10;      b=number/100;      if (number>=10&&number<100)      {   if(g!=s)  return 0; }//The number is not huiwen      else          if (number>=100&&number<1000)          {   if(g!=b)  return 0; }//The number is not huiwen      return 1; // The number is huiwen  }  int main()  {      int a, b; // Enter two numbers as the range of prime palindromes      int i;        scanf("%d %d", &a, &b);        for (i = a; i <= b; i++)          {             if (primer(i) && huiwen(i))                  printf("%d\n", i);          }      return 0;  } |

Q560.(10分)两个乒乓球队进行比赛，各出三人。甲队为 a,b,c 三人，乙队为 x,y,z 三人。已抽签决定比赛名单。有人向队员打听比赛的名单。 a 说他不和 x 比， c 说他不和 x,z 比，请编程序找出三队赛手的名单。

\*\*输出格式要求："order is a--%c\tb--%c\tc--%c\n"

程序运行示例如下：

order is a--z b--x c—y

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      char i, j, k; /\*i 是 a 的对手， j 是 b 的对手， k 是 c 的对手 \*/        for (i = 'x'; i <= 'z'; i++)            for (j = 'x'; j <= 'z'; j++)            {                if (i != j)                    for (k = 'x'; k <= 'z'; k++)                    {                      if (i != k && j != k)                        {                          if (i != 'x' && k != 'x' && k != 'z')                                printf("order is a--%c\tb--%c\tc--%c\n", i, j, k);                        }                    }            }          return 0;  } |

Q454.(10分)已知xyz+yzz=532，其中x，y，z都是数字（0-9），编写一个程序求出x，y，z分别代表什么数字。

\*\*输出格式要求："x=%d,y=%d,z=%d"

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int x, y, z;        for (x = 1; x < 6; x++)      {          for (y = 0; y < 4; y++)          {              for (z = 0; z < 4; z++)              {                  if ((x \* 100 + y \* 10 + z) +                          (y \* 100 + z \* 10 + z) == 532)                  {                      printf("x=%d,y=%d,z=%d", x, y, z);                  }              }          }      }  } |

Q410.(10分)亲密数。如果整数A的全部因子（包括1，但不包括其自身）之和等于B，且整数B的全部因子（包括1，但不包括B本身）之和等于A，则将整数A和B称为亲密数。3000以内的全部亲密数。

\*\*输出格式要求："There are following friendly-numbers pair smaller than 3000:\n" " %4d..%4d"

程序运行示例如下：

There are following friendly-numbers pair smaller than 3000:

220.. 284 1184..1210 2620..2924

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int a,i,m,n;   printf("There are following friendly-numbers pair smaller than 3000:\n");   for(a=1;a<3000;a++)   { for(m=0,i=1;i<=a/2;i++)        if(!(a%i))         m+=i;     for(n=0,i=1;i<=m/2;i++)         if(!(m%i))            n+=i;              if(n==a&&a<m)                 printf("   %4d..%4d",a,m);         }  } |

Q611.(10分)每个合数都可以写成几个质数相乘的形式。将一个正整数分解质因数。例如90=2\* 3\* 3\* 5,而质因数分解只针对合数，质数没有质因数。当程序输入质数时，输出"Invalid input."，否则输出其质因数序列。

\*\*输入格式要求："%d," "%d"

\*\*输出格式要求："%d" "Invalid input.\n"

程序运行示例1如下：

89

Invalid input.

程序运行示例2如下：

98

2,7,7

### 参考答案

|  |
| --- |
| #include <math.h>  int main()  {      int num, i, apnum = 0;      int copy;      int first = 0;      scanf("%d", &num);      copy = num;      for (i = 2; i < copy; i++)      {          if (!(num % i))          {              if (first++ != 0)                  printf(",");              apnum = i;              num = num / i;              i--;              printf("%d", apnum);          }      }      if (apnum == 0)          printf("Invalid input.\n");      return 0;  } |

Q569.(10分)编写程序，把一维数组a 输入任意6个整数，假设a 为： 7 4 5 9 1 2

输出：

7 4 5 9 1 2

2 7 4 5 9 1

1 2 7 4 5 9

9 1 2 7 4 5

5 9 1 2 7 4

4 5 9 1 2 7

\*\*输入格式要求："%d"

\*\*输出格式要求："%d "

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a[6], i, j, k, m;      for (i = 0; i < 6; i++)          scanf("%d", &a[i]);      for (i = 5; i >= 0; i--)      {          printf("\n");          for (m = 0; m < 6; m++)              printf("%d ", a[m]);          k = a[5];          for (j = 5; j >= 1; j--)              a[j] = a[j - 1];          a[0] = k;      }  } |

Q1333.(10分)从键盘为3\*3的矩阵输入数据，找出主对角线上最大的元素，以及所在的行号。

\*\*输入格式要求："%d"

\*\*输出格式要求："max=%d ,row=%d"

程序的运行示例如下：

1 2 3

4 5 6

7 8 9

max=9 ,row=2

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int i, j, row = 0, max;      int a[3][3];      for (i = 0; i < 3; i++)      {          for (j = 0; j < 3; j++)          {              scanf("%d", &a[i][j]);          }      }      max = a[0][0];      for (i = 0; i < 3; i++)      {          if (max < a[i][i])          {              max = a[i][i];              row = i;          }      }      printf("max=%d ,row=%d", max, row);  } |

Q1307.(10分)

 用迭代法求x=sqrt(a)。求平方根的迭代公式为：xn+1= (1/2)(xn+ a/xn)，要求前后两次求出的x的差的绝对值小于10-5。

\*\*输入格式要求："%f" 提示信息："请输入一个整数："  
\*\*输出格式要求："%5.2f的平方根=%8.5f\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      float a, xn0, xn1;        printf("请输入一个整数：");      scanf("%f", &a);        xn0 = a / 2;      xn1 = (xn0 + a / xn0) / 2;      do      {          xn0 = xn1;          xn1 = (xn0 + a / xn0) / 2;      }      while (fabs(xn0 - xn1) >= 1e-5);      printf("%5.2f的平方根=%8.5f\n", a, xn1);        return 0;  } |

Q334.(10分)\*对输入的行和字符进行计数。在计算机中，一行是以一个回车符\n作为行结束标记的，这样在程序中可以通过搜索\n对行进行计数。在UNIX操作系统中，一般有ctrl+d作为文件结束标记，其字符码为-1.当输入ctrl+d时表示文件输入结束，停止计数。

\*\*输出格式要求："chars=%d, lines=%d\n"

### 参考答案

|  |
| --- |
| #define EOF -1  #include <stdio.h>  main()  {      int c,nl,nc;      nl=nc=0;      while((c=getchar())!=EOF)      {          ++nc;          if(c=='\n')              ++nl;      }      printf("chars=%d, lines=%d\n",nc,nl);  } |

Q597.(10分)编写程序：有1,2,3,4共四个数字,能组成多少个互不相同的且无重复数字的三位数,都是多少。

\*\*输出格式要求：" % d % d % d\n" "共有%d种组合！"

程序运行示例如下：

1 2 3

1 2 4

1 3 2

1 3 4

1 4 2

1 4 3

2 1 3

2 1 4

2 3 1

2 3 4

2 4 1

2 4 3

3 1 2

3 1 4

3 2 1

3 2 4

3 4 1

3 4 2

4 1 2

4 1 3

4 2 1

4 2 3

4 3 1

4 3 2

共有24种组合！

返回 查看参考答案

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, j, k, count;        count = 0;      for (i = 1; i < 5; i++)          for (j = 1; j < 5; j++)              for (k = 1; k < 5; k++)                  if (i != k && i != j && k != j)                  {                      count++;                      printf(" % d % d % d\n", i, j, k);                  }      printf("共有%d种组合！", count);  } |

Q555.(10分)魔术师利用一副牌中的13张红桃，预先将它们排好后迭在一起，牌面朝下。对观众说：“我不看牌”，只数数就可以猜到每张牌是什么，我大声数，你们听，不信？你们就看。魔术师将最上面的那张牌数为1，把它翻过来正好是红桃A，将红桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌的下面，将第二张牌翻过来，正好是红桃2，也将它放在桌子上。第三次数1、2、3，将前面两张依次放在这迭牌的下面，再翻第三张牌正好是红桃3.这样依次将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

\*\*输出格式要求："%d "

### 参考答案

|  |
| --- |
| #include<stdio.h>    int main()  {      int cards[14] = {0};      int i, j = 1, n;      for (i = 1 ; i <= 13 ; i++)      {          n = 1;          do          {              if (j > 13)                  j = 1;              if (cards[j] > 0)                  j++;              else              {                  if (n == i)                      cards[j] = i;                  j++;                  n++;              }          }          while (n <= i);      }      for (i = 1; i <= 13; i++)          printf("%d ", cards[i]);      printf("\n");        return 0;  } |

Q443.(10分)小明买书

小明假期同爸爸一起去书店，他选中了六本书，每本书的单价分别：3.1，1.7，2，5.3，0.9和7.2元。不巧的是，爸爸兜里只带了十几块钱，为了让小明过一个愉快的假期，爸爸仍然同意买书，但提出了一个要求，要小明从六本书中选出若干本，使得单价相加所得的和同10最接近。你能够帮助小明解决这个问题吗？

\*\*输出格式要求：" 10(+ -)%.2f=" "%.2f+"

程序运行示例如下：

3.1 1.7 2 5.3 0.9 7.2

10(+ -)0.10=2.00+0.90+7.20

10(+ -)0.10=1.70+2.00+5.30+0.90

10(+ -)0.10=3.10+1.70+5.30

### 参考答案

|  |
| --- |
| #include <math.h>  main()  {    int d[6],m,i,j;    long b[63],flag;    float c[6]={3.1,1.7,2,5.3,0.9,7.2},min,x;    /\*printf("Please enter the prices of 6 books:");    for(i=0;i<6;i++)      scanf("%f",&c[i]);\*/    for(i=0,min=-1,d[0]=0;d[0]<2;d[0]++)      for(d[1]=0;d[1]<2;d[1]++)        for(d[2]=0;d[2]<2;d[2]++)          for(d[3]=0;d[3]<2;d[3]++)            for(d[4]=0;d[4]<2;d[4]++)             for(d[5]=0;d[5]<2;d[5]++)             {               for(flag=0,x=0.,j=5;j>=0;j--)               {                 x+=c[j]\*d[j];flag=flag\*10+d[j];               }               x=((x-10>0)?x-10:10-x);               if(min<0)               {                 min=x;                 b[i++]=flag;               }               else if(min-x>1.e-6)               {                   min=x;b[0]=flag;i=1;               }               else if(fabs((double)x-min)<1.e-6)                   b[i++]=flag;             }      for(m=0;m<i;m++)      {        printf("  10(+ -)%.2f=",min);        for(flag=b[m],j=0;flag>0;j++,flag/=10)              if(flag%10)                  if(flag>1) printf("%.2f+",c[j]);                  else     printf("%.2f\n",c[j]);      }  } |

Q152.(1分)华氏和摄氏温度的转换公式为C=5/9×（F-32）。

其中，C表示摄氏温度，F表示华氏温度。

要求：华氏0℉～300℉，每隔20℉输出一个华氏温度对应的摄氏温度值。

\*\*输出格式要求："%4.0f%10.1f\n"，第一个浮点数为华氏温度，第二个浮点数为摄氏温度。

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int    upper,step;      float  fahr = 0,celsius;      upper = 300;            //1      step = 20;              //1      while (fahr <= upper)   //1      {          celsius = 5.0 / 9 \* (fahr - 32) ;  //1              printf("%4.0f%10.1f\n", fahr, celsius);              fahr = fahr + step;              //1      }  } |

Q1211.(10分)

|  |
| --- |
| 找出以下程序的错误，并改正之。  韩信点兵。韩信有一队兵，他想知道有多少人，便让士兵排队报数：按从1至5报数，最末一个士兵报的数为1；按从1至6报数，最末一个士兵报的数为5；按从1至7报数，最末一个士兵报的数为4；最后再按从1至11报数，最末一个士兵报的数为10。你知道韩信至少有多少兵吗？  #include <stdio.h>  main()  {      int  x;        while (1)      {          if(x%5=1&&x%6=5&&x%7=4&&x%11=10)          {              break;          }          x++      }      printf(" x = %d\n", x);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int  x=1;      while (1)      {          if(x%5==1&&x%6==5&&x%7==4&&x%11==10)          {              break;          }          x++;      }      printf(" x = %d\n", x);  } |

Q464.(10分)程序：显示平方值的表格

现在要编写一个程序来显示平方值的表格。首先程序提示用户输入数n。然后显示出n行的输出，其中每一行包含两个数：一个是1~n的数，另一个则是此数的平方值。

\*\*输入格式要求："%d" 提示信息："This program prints a table of squares.\n" "Enter number of entries in table:\n"

\*\*输出格式要求："%10d%10d\n"

This program prints a table of squares.

Enter number of entries in table: 5

1 1

2 4

3 9

4 16

5 25

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, n;        printf("This program prints a table of squares.\n");      printf("Enter number of entries in table:\n");      scanf("%d", &n);        i = 1;      while (i <= n)      {          printf("%10d%10d\n", i, i \* i);          i++;      }        return 0;  } |

Q422.(10分)最大公约数和最小公倍数。求任意两个正整数的最大公约数GCD和最小公倍数LCM。

\*\*输入格式要求："%d%d" 提示信息："Input a & b:"

\*\*输出格式要求："The GCD of %d and %d is:%d\n" "The LCM of them is:%d\n"

程序运行示例如下：

Input a & b:34 78

The GCD of 34 and 78 is:2

The LCM of them is:1326

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int a,b,num1,num2,temp;   printf("Input a & b:");   scanf("%d%d",&num1,&num2);   if(num1>num2)   { temp=num1;   num1=num2;   num2=temp;   }   a=num1;b=num2;   while(b!=0)    {  temp=a%b;       a=b;       b=temp;    }    printf("The GCD of %d and %d is:%d\n",num1,num2,a);    printf("The LCM of them is:%d\n",num1\*num2/a);  } |

Q550.(20分)在歌星大奖赛中，有10个评委为参赛的选手打分，分数为1到100分的整数。选手最后得分为：去掉一个最高分和一个最低分后其余8个分数的平均值。请编写代码实现该计分程序。

\*\*输入格式要求："%d" 提示信息："Input score %d\n"

\*\*输出格式要求："Canceled max score: %d\nCanceled min score: %d\n" "Average score: %d\n"

程序运行示例如下：

Input score 1

98

Input score 2

87

Input score 3

89

Input score 4

84

Input score 5

83

Input score 6

87

Input score 7

89

Input score 8

92

Input score 9

93

Input score 10

95

Canceled max score: 98

Canceled min score: 83

Average score: 89

注：不能使用数组进行编程。

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int score, max, min, i, sum;      max =0;      min = 101;      sum = 0;      for (i = 1; i <= 10; i++)      {          printf("Input score %d\n", i);          scanf("%d", &score);          sum += score;          if (score > max)              max = score;          if (score < min)              min = score;      }      printf("Canceled max score: %d\nCanceled min score: %d\n", max, min);      printf("Average score: %d\n", (sum - max - min) / 8);        return 0;  } |

Q1311.(10分)两个乒乓球队进行比赛，各出三个人。甲队为A、B、C三人，乙队为x、y、z三人。已抽签决定比赛名单。有人向队员打听比赛的名单。A说他不和x比，C说他不和x、z比，请编程序找出三对赛手的名单。

\*\*输出格式要求："顺序为：\na--%c\tb--%c\tc--%c\n"

程序的运行示例如下：

顺序为：

a--z b--x c—y

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      char i, j, k; /\* i是a的对手，j是b的对手，k是c的对手 \*/        for (i = 'x'; i <= 'z'; i++)      {          for (j = 'x'; j <= 'z'; j++)          {              if (i != j)              {                  for (k = 'x'; k <= 'z'; k++)                  {                      if (i != k && j != k)                      {                          if (i != 'x' && k != 'x' && k != 'z')                          {                              printf("顺序为：\na--%c\tb--%c\tc--%c\n", i, j, k);                          }                      }                  }              }          }      }        return 0;  } |

Q342.(10分)输入10个整数，求其中正数的个数及平均值，精确到小数点后两位。

\*\*输入格式要求："%d" 提示信息："Input integer:"

\*\*输出格式要求："Plus number:%d,average value:%.2f" "Plus number:0,average value: 0"

程序运行示例如下：

Input integer:1

Input integer:2

Input integer:-9

Input integer:8

Input integer:-3

Input integer:3

Input integer:81

Input integer:3

Input integer:0

Input integer:5

Plus number:7,average value:14.71

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,count,j,sum;      sum=count=0;      for(i=1;i<=10;i++)      {          printf("Input integer:");          scanf("%d",&j);          if(j<=0) continue;          count++;          sum+=j;      }      if(count)          printf("Plus number:%d,average value:%.2f",count,1.0\*sum/count);      else printf("Plus number:0,average value: 0");  } |

Q1340.(10分)用双重循环编程打印以下图形：

ABCDEF

BCDEF

CDEF

DEF

EF

F

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      char i, ch;      for (i = 0; i < 6; i++)      {          for (ch = 'A' + i; ch <= 'F'; ch++)          {              printf("%c", ch);          }          printf("\n");      }      return 0;  } |

Q122.(25分)编程实现输出1—100之间能被7整除但不能被3整除的所有偶数，

并求满足该条件的所有数的累加和。

\*\*要求输入提示信息为：无输入提示信息和输入数据

\*\*要求输出格式为： （1）"%5d"

（2）"\nsum=%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,sum=0;      for(i=1;i<=100;i++) //2      {          if(i/2\*2==i && i/7\*7==i && i/3\*3!=i) //2          {              printf("%5d",i);  //2              sum=sum+i;  //2          }      }      printf("\nsum=%d\n",sum);  //2  } |

Q154.(1分)附加题：改错题：

我国古代的《张丘建算经》中有这样一道著名的百鸡问题：

“鸡翁一，值钱五；鸡母一，值钱三；鸡雏三，值钱一。百钱买百鸡，问鸡翁、母、雏各几何？”

其意为：公鸡每只5元，母鸡每只3元，小鸡3只1元。用100元买100只鸡，问公鸡、母鸡和小鸡各能买多少只？

已知本题的解有4个。

下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

注意：将修改后的完整的源程序写在答题区内。对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分，如果只改正了部分错误，则不加分。

\*\*输出格式要求：见以下代码。

#include <stdio.h>

main()

{

int x, y, z;

for (x = o; x <= 20; x++);

{

for (y = o; y <= 33; y++);

{

z = 100 – x - y;

if (5x + 3y + z/3 = 100)

{

printf("x=%d, y=%d, z=%d\n", x, y, z);

}

}

}

}

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int x, y, z;      for (x = 0; x <= 20; x++)      {          for (y = 0; y <= 33; y++)          {                  z = 100 - x - y;                  if (5\*x + 3\*y + z/3.0 == 100)                  {                      printf("x=%d, y=%d, z=%d\n", x, y, z);                  }          }      }  } |

Q475.(10分)下一代因特网IPv6的地址占128位（二进制位，也称为比特），假设以每秒100万个地址的速度分配，请问分配完需要花费多少年？

\*\*输出格式要求："%f"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      printf("%f", (ldexp(1.0, 128) / 1000000 / (365 \* 24 \* 60 \* 60)));        return 0;  } |

Q1710.(10分)任意输入一个整数m，若m不是素数，则输出其所有不包括1和自身的因子；否则输出“没有因子，是素数”的相关提示信息。

输入提示信息："Please enter a number:"

输入格式："%d"

输出格式：

有因子时："%d\n"

无因子时："It is a prime number.No divisor!\n"

### 参考答案

|  |
| --- |
| #include  <math.h>  #include  <stdio.h>  main()  {      int  m, i;      int  flag = 1;               //置标志变量flag初值为真（1）      printf("Please enter a number:");      scanf("%d", &m);      if (m <= 1)                  //负数,0和1都不是素数      {          flag = 0;                //将标志变量flag置为假（0）          m = (int)fabs(m);       //对m取绝对值，以便对负数输出因子但不输出正负号      }      for (i = 2; i <= m - 1; i++)      {          if (m % i == 0)          //若m能被i整除，表明i是m的一个因子          {              flag = 0;             //则将标志变量flag置为假（0）              printf("%d\n", i);  //输出m的因子i          }      }      if (flag)   //若标志变量flag为真，则表明全部检验完毕，未发现能整除m的数      {          printf("It is a prime number.No divisor!\n");//m是素数，没有因子      }  } |

Q284.(20分)编程实现输出1—100之间能被7整除但不能被3整除的所有偶数，

并求满足该条件的所有数的累加和。

\*\*要求输入提示信息为：无输入提示信息和输入数据

\*\*要求输出格式为：（1）"%5d"

（2）"\nsum=%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,sum=0;      for(i=1;i<=100;i++) //2      {          if(i/2\*2==i && i/7\*7==i && i/3\*3!=i) //2          {              printf("%5d",i);  //2              sum=sum+i;  //2          }      }      printf("\nsum=%d\n",sum);  //2  } |

Q1705.(10分)编程计算1!+2!+3!+4!+…+10!的值。

输入格式：无

输出格式："1!+2!+...+10! = %ld\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      long term = 1, sum = 0;      int i;      for (i=1; i<=10; i++)      {          term = term \* i;          sum = sum + term;      }      printf("1!+2!+...+10! = %ld\n", sum);  } |

Q333.(10分)从键盘中读入一系列字符，原样输出到屏幕上，直到输入字母a时才停止。

程序运行示例1如下：

dbca （输入）

dbc （输出）

程序运行示例2如下：

first （输入）

first （输出）

second （输入）

second （输出）

a （输入）

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char ch;        while ((ch = getchar()) != 'a')      {          putchar(ch);      }  } |

Q1272.(10分)打印100~200之间的所有素数。改正错误。

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  {      int m,i;        for(m = 100; m <= 200; m++)      {                for(i = 2; i < sqrt(m) && flag; i++)              {              if(m%i == 0)                  flag = 0;          }          if(flag)          {              printf("%d ", m);          }      }  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  {      int m,i,flag;        for(m = 100; m <= 200; m++)      {          flag = 1;              for(i = 2; i < sqrt(m) && flag; i++)              {              if(m%i == 0)                  flag = 0;          }          if(flag)          {              printf("%d ", m);          }      }  } |

Q407.(10分)一个奇异的三位数。一个自然数的七进制表达式是一个三位数，而这个自然数的九进制表示也是一个三位数，且这两个三位数的数码顺序正好相反，求这个三位数。

\*\*输出格式要求："The special number with 3 digits is:" "%d%d%d(7)=%d%d%d(9)=%d(10)\n"

程序运行示例如下：

The special number with 3 digits is:503(7)=305(9)=248(10)

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {int i,j,k;   for(i=1;i<7;i++)     for(j=0;j<7;j++)       for(k=1;k<7;k++)         if(i\*9\*9+j\*9+k==i+j\*7+k\*7\*7)         {          printf("The special number with 3 digits is:");          printf("%d%d%d(7)=%d%d%d(9)=%d(10)\n",                                       k,j,i,i,j,k,i\*9\*9+j\*9+k);      }  } |

Q561.(10分)某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下：每位数字都加上 5, 然后用和除以 10 的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

\*\*输入格式要求："%d" 提示信息："Input a 4 digits number\n"

\*\*输出格式要求："%d"

程序运行示例如下：

Input a 4 digits number

1234

9876

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      int a, i, aa[4], t;        printf("Input a 4 digits number\n");      scanf("%d", &a);        aa[0] = a % 10;      aa[1] = a % 100 / 10;      aa[2] = a % 1000 / 100;      aa[3] = a / 1000;        for (i = 0; i <= 3; i++)      {          aa[i] += 5;          aa[i] %= 10;      }        for (i = 0; i <= 3 / 2; i++)      {          t = aa[i];          aa[i] = aa[3 - i];          aa[3 - i] = t;      }        for (i = 3; i >= 0; i--)          printf("%d", aa[i]);      printf("\n");        return 0;  } |

Q613.(10分)有5个人坐在一起，问第五个人多少岁？他说比第4个人大2岁。问第4个人岁数，他说比第3个人大2岁。问第三个人，又说比第2人大两岁。问第2个人，说比第一个人大两岁。最后问第一个人，他说是10岁。请问第五个人多大？

\*\*输出格式要求："%d"

### 参考答案

|  |
| --- |
| age(int n)  {      int c;      if (n == 1) c = 10;      else c = age(n - 1) + 2;      return(c);  }  main()  {      printf("%d", age(5));  } |

Q377.(10分)输入一个正整数，要求以相反的顺序输出该数。用递归方法实现。

\*\*输入格式要求："%d" 提示信息："Enter number:"

\*\*输出格式要求："%d"

程序运行示例如下：

Enter number:35567899

99876553

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int num;      printf("Enter number:");      scanf("%d",&num);      printn(num);      printf("\n");  }  printn(int n)  {      if(0<=n&&n<=9)          printf("%d",n);      else      {          printf("%d",n%10);          printn(n/10);      }  } |

Q1725.(10分)计算游戏人员的年龄.有5个人围坐在一起，问第5个人多大年纪，他说比第4个人大2岁；问第4个人，他说比第3个人大2岁；问第3个人，他说比第2个人大2岁；问第2个人，他说比第1个人大2岁。第1个人说自己10岁，问第5个人多大年纪。

输入格式：无

输出格式："The 5th person's age is %d\n"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  unsigned int ComputeAge(unsigned int n);  int main(void)  {      unsigned int n = 5;      printf("The 5th person's age is %d\n", ComputeAge(n));      return 0;  }  // 函数功能：用递归算法计算第n个人的年龄  unsigned int ComputeAge(unsigned int n)  {      unsigned int age;      if (n == 1)      {          age = 10;      }      else      {          age = ComputeAge(n - 1) + 2;      }      return age;  } |

Q1121.(10分)计算n!算法如下，请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

long fact(int n);

main()

{

int n, result = 0;

printf("Input n:");

scanf("%d", &n);

result = fact(n);

printf("%d != %d", n, result);

}

long fact(int n)

{

int result;

if (n < 0)

printf("n<0,data error!\n");

else

{

result = n \* fact(n - 1);

return result;

}

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    long fact(int n);    main()  {      int n;      unsigned long result = 0;        printf("Input n:");      scanf("%d", &n);        result = fact(n);      if (result != -1)          printf("%d != %u", n, result);  }  long fact(int n)  {      unsigned long result;      if (n < 0)      {          printf("n<0,data error!\n");          return -1;      }      else if (n == 0 || n == 1)          return 1;      else      {          result = n \* fact(n - 1);          return result;      }  } |

Q375.(10分)请用递归的方法计算下列函数的值：px(x,n)=x-x^2+x^3-x^4+…((-1)^n-1)(x^n) n>0

\*\*输入格式要求："%lf%d" 提示信息："Enter X and N:"

\*\*输出格式要求："px=%lf\n"

程序运行示例如下：

Enter X and N:4 6

px=-3276.000000

### 参考答案

|  |
| --- |
| #include <stdio.h>  double px(double x,int n)  {      if(n==1)          return x;      else          return (x\*(1-px(x,n-1)));  }  main()  {      double x; int n;      printf("Enter X and N:");      scanf("%lf%d",&x,&n);      printf("px=%lf\n",px(x,n));  } |

Q533.(10分)用递归方法编程计算Fibonacci数列的前N项。其中fib(0)=0,fib(1)=1,fib(n)=fib(n-1)+fib(n-2)。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："Fib(%d)=%d\n"

程序运行示例如下：

Input n:10

Fib(1)=1

Fib(2)=1

Fib(3)=2

Fib(4)=3

Fib(5)=5

Fib(6)=8

Fib(7)=13

Fib(8)=21

Fib(9)=34

Fib(10)=55

### 参考答案

|  |
| --- |
| #include <stdio.h>  long Fib(int a);  int main()  {      int n, i, x;      printf("Input n:");      scanf("%d", &n);      for (i = 1; i <= n; i++)      {          x = Fib(i);    /\* 调用递归函数Fib()计算Fibonacci数列的第n项 \*/          printf("Fib(%d)=%d\n", i, x);      }      return 0;  }  /\* 函数功能：用递归法计算Fibonacci数列中的第n项的值 \*/  long Fib(int n)  {      long f;      if (n == 0)   f = 0;              /\* 基线情况 \*/      else if (n == 1)   f = 1;        /\* 基线情况 \*/      else   f = Fib(n - 1) + Fib(n - 2); /\* 一般情况 \*/      return f;  } |

Q368.(15分)编写函数，通过指针连接两个字符串。

\*\*输入格式要求："%s" 提示信息："Enter string 1:" "Enter string 2:"

\*\*输出格式要求："a+b=%s\n"

程序运行示例如下：

Enter string 1:happy

Enter string 2:hollween

a+b=happyhollween

### 参考答案

|  |
| --- |
| #include <stdio.h>  char \*strcat(char \*str1,char \*str2)  {      char \*p=str1;      while(\*p!='\0') p++;      while(\*p++=\*str2++);      return str1;  }  main()  {      char a[50],b[30];      printf("Enter string 1:");      scanf("%s",a);      printf("Enter string 2:");      scanf("%s",b);      printf("a+b=%s\n",strcat(a,b));  } |

Q1626.(10分)用字符指针作函数参数编程实现字符串逆序存放功能。

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 80  void Inverse(char \*pStr);  int main()  {      char a[N];      printf("Input a string:");      gets(a);      Inverse(a);      printf("Inversed results:%s\n", a);      return 0;  }  /\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/  void Inverse(char \*pStr)  {      int  len;      char temp;      char \*pStart;       /\* 指针变量pStart指向字符串的第一个字符 \*/      char \*pEnd;         /\* 指针变量pEnd指向字符串的最后一个字符 \*/      len = strlen(pStr); /\* 求出字符串长度 \*/      for (pStart = pStr, pEnd = pStr + len - 1; pStart < pEnd; pStart++, pEnd--)      {          temp = \*pStart;          \*pStart = \*pEnd;          \*pEnd = temp;      }  } |

Q1638.(35分)按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char \*dstStr, char \*srcStr);

将srcStr字符串连接到dstStr字符串后面，；

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 80  void  MyStrcat(char \*dstStr, char \*srcStr);  int main()  {      char  s[N], t[N];      printf("Input a string:");      gets(s);      printf("Input another string:");      gets(t);      MyStrcat(s, t);      printf("Concatenate results:%s\n", s);      return 0;  }  /\*  函数功能：将字符串srcStr连接到字符串dstStr之后 \*/  void  MyStrcat(char \*dstStr, char \* srcStr)  {      while (\*dstStr != '\0')      {          dstStr++;      }      while (\*srcStr != '\0')/\* 若srcStr所指字符不是字符串结束标志 \*/      {          \*dstStr = \*srcStr;  /\* 将srcStr所指字符复制到dstStr所指内存中\*/          srcStr++;           /\* 使srcStr指向下一个字符 \*/          dstStr++;           /\* 使dstStr指向下一个存储单元 \*/      }      \*dstStr = '\0';     /\* 在字符串dstStr的末尾添加字符串结束标志 \*/  } |

Q369.(10分)编写函数，通过指针将一个字符串反向。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："a=%s\n"

程序运行示例如下：

Enter string:love

a=evol

### 参考答案

|  |
| --- |
| #include <stdio.h>  revstr(char \*s)  {      char \*p=s,c;      while(\*p) p++;      p--;      while(s<p)      {          c=\*s;          \*s++=\*p;          \*p--=c;      }  }  main()  {      char a[50];      printf("Enter string:");      scanf("%s",a);      revstr(a);      printf("a=%s\n",a);  } |

Q1624.(10分)用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*输入格式要求："%s"

输入提示信息：

"Input a string:"

"Input a character:"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 100  void  Squeeze(char \*s, char c);  int main()  {      char  str[20], ch;      printf("Input a string:");      gets(str);      printf("Input a character:");      ch = getchar();      Squeeze(str, ch);      printf("Results:%s\n", str);      return 0;  }  void  Squeeze(char \*s, char c)  {      char str[N];      char \*t = str;      strcpy(t, s);      for (; \*t != '\0'; t++)      {          if (\*t != c)          {              \*s = \*t;              s++;          }      }      \*s = '\0';  /\* 在字符串t2的末尾添加字符串结束标志 \*/  } |

Q545.(10分)键盘任意输入两个整数，利用指针编程实现将其交换后再重新输出。

程序的运行示例如下：

Please enter a,b:15,8

Before swap: a = 15, b = 8

After swap: a = 8, b = 15

### 参考答案

|  |
| --- |
| #include  <stdio.h>  void  Swap(int \*x, int \*y);  int main()  {      int  a, b;      printf("Please enter a,b:");      scanf("%d,%d", &a, &b);      printf("Before swap: a = %d, b = %d\n", a, b);/\* 打印交换前的a,b \*/      Swap(&a, &b);                                  /\* 按地址调用函数Swap() \*/      printf("After swap: a = %d, b = %d\n", a, b);/\* 打印交换后的a,b \*/      return 0;  }  /\* 函数功能：交换两个整型数的值 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  } |

Q1282.(10分)

|  |
| --- |
| 程序改错。在一个3x4矩阵中找出最大数及最大数所在元素的下标。  #include <stdio.h>  #define M 3  #define N 4  int FindMax(int (\*p)[N],int m,int n, int \*pRow, int \*pCol)  main()  {      int score[M][N], i, j, maxScore, row, col;      for ( i=0; i<M; i++ )      {            for ( j=0; j<N; j++ )            {                 scanf("%d", &score[i][j]);            }      }      maxScore = FindMax(\*score, M, N, &row, &col);      printf("%d %d %d\n", maxScore, row+1, col+1);  }    int FindMax( int (\*p)[N], int m, int n,  int \*pRow, int \*pCol )  {      int  i, j, max;      max = \*(p);      pRow = 0;      pCol = 0;      for (i=0; i<m; i++)      {          for (j = 0; j<n; j++)                  {              if ( \*(\*(p+i)+j) > max )                          {                  max = \*(\*(p+i)+j) ;                  \*pRow = i;                  \*pCol = j;                          }          }      }    } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define M 3  #define N 4  int FindMax(int (\*p)[N],int m,int n, int \*pRow, int \*pCol);  main()  {      int score[M][N], i, j, maxScore, row, col;      for ( i=0; i<M; i++ )      {            for ( j=0; j<N; j++ )            {                 scanf("%d", &score[i][j]);            }      }      maxScore = FindMax(score, M, N, &row, &col);      printf("%d %d %d\n", maxScore, row+1, col+1);  }    int FindMax( int (\*p)[N], int m, int n,  int \*pRow, int \*pCol )  {      int  i, j, max;      max = \*(\*(p));      \*pRow = 0;      \*pCol = 0;      for (i=0; i<m; i++)      {          for (j = 0; j<n; j++)                  {              if ( \*(\*(p+i)+j) > max )                          {                  max = \*(\*(p+i)+j) ;                  \*pRow = i;                  \*pCol = j;                          }          }      }      return max;  } |

Q370.(10分)编写函数，通过指针求字符串的长度。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："string length=%d\n"

程序运行示例如下：

Enter string:nihao

string length=5

### 参考答案

|  |
| --- |
| #include <stdio.h>  strlen(char \*s)  {      char \*p=s;      while(\*p)          p++;      return p-s;  }  main()  {      char a[50];      printf("Enter string:");      scanf("%s",a);      printf("string length=%d\n",strlen(a));  } |

Q1210.(10分)

|  |
| --- |
| 编程输入10个数，找出其中的最大值及其所在的数组下标位置。找出其中错误并改正之。  #include <stdio.h>    int FindMax(int num[], int n, int \*pMaxPos);    main()  {      int num[10], maxValue, maxPos, minValue, minPos, i;        printf("Input 10 numbers:\n ");      for (i=0; i<10; i++)      {          scanf("%d", num[i]);      }      maxValue = FindMax(num, 10, \*maxPos);      printf("Max=%d, Position=%d\n",maxValue, maxPos);  }  int FindMax(int num[], int n, int \*pMaxPos)  {      int i, max;        max = num[0];        for (i = 1, i < n, i++)      {          if (num[i] > max)          {              max = num[i];              \*pMaxPos = i;          }      }      return max;  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>    int FindMax(int num[], int n, int \*pMaxPos);    main()  {      int num[10], maxValue, maxPos, minValue, minPos, i;        printf("Input 10 numbers:\n ");      for (i=0; i<10; i++)      {          scanf("%d", &num[i]);      }      maxValue = FindMax(num, 10, &maxPos);      printf("Max=%d, Position=%d\n",maxValue, maxPos);  }  int FindMax(int num[], int n, int \*pMaxPos)  {      int i, max;        max = num[0];      \*pMaxPos = 0;      for (i = 1; i < n; i++)      {          if (num[i] > max)          {              max = num[i];              \*pMaxPos = i;          }      }      return max;  } |

Q362.(10分)编写一个交换变量值的C函数，交换数组a和数组b中的对应元素。

a[5]={1,2,3,4,5}

b[5]={10,20,30,40,50}

\*\*输出格式要求："a[%d]=%2d, " "b[%d]=%2d, "

程序运行示例如下：

a[0]=10, a[1]=20, a[2]=30, a[3]=40, a[4]=50,

b[0]= 1, b[1]= 2, b[2]= 3, b[3]= 4, b[4]= 5,

### 参考答案

|  |
| --- |
| #include <stdio.h>  swap(pa,pb)      int \*pa,\*pb;  {      int temp;      temp=\*pa;\*pa=\*pb;\*pb=temp;  }  main()  {      int a[5]={1,2,3,4,5};      int b[5]={10,20,30,40,50};      int i;      for(i=0;i<5;i++)          swap(&a[i],&b[i]);      for(i=0;i<5;i++)          printf("a[%d]=%2d, ",i,a[i]);      for(printf("\n"),i=0;i<5;i++)          printf("b[%d]=%2d, ",i,b[i]);  } |

Q576.(10分)输入一个整形数，然后按汉语的习惯，将其读出来并输出。如1052，读作：一千零五十二。

输入样例：

1052

输出样例：

一千零五十二

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #include <stdlib.h>  const char shuzi[10][4] = {"零", "一", "二", "三", "四", "五", "六", "七", "八", "九"};  const char weishu[5][4] = {"十", "百", "千", "万", "亿"};    void read(char \*str);    int main()  {      char str[11] = {'0'};      scanf("%s", str + 1);      read(str + 1);      return 0;  }    void read(char \*str)  {      if (strlen(str) == 0)          return;      else      {          switch (str[0])          {          case '0':              if (str[1] != '0' && strlen(str) >= 2)                  printf("%s", shuzi[0]);              break;          case '1':              if (strlen(str) != 2 && strlen(str) != 6)                  printf("%s", shuzi[1]);              else if (\*(str - 1) != '0')                  printf("%s", shuzi[1]);              break;          case '2':              printf("%s", shuzi[2]);              break;          case '3':              printf("%s", shuzi[3]);              break;          case '4':              printf("%s", shuzi[4]);              break;          case '5':              printf("%s", shuzi[5]);              break;          case '6':              printf("%s", shuzi[6]);              break;          case '7':              printf("%s", shuzi[7]);              break;          case '8':              printf("%s", shuzi[8]);              break;          case '9':              printf("%s", shuzi[9]);              break;          default:              return;          }          switch (strlen(str))          {          case 9:              printf("%s", weishu[4]);              break;          case 8:              if (str[0] != '0')                  printf("%s", weishu[2]);              break;          case 7:              if (str[0] != '0')                  printf("%s", weishu[1]);              break;          case 6:              if (str[0] != '0')                  printf("%s", weishu[0]);              break;          case 5:              printf("%s", weishu[3]);              break;          case 4:              if (str[0] != '0')                  printf("%s", weishu[2]);              break;          case 3:              if (str[0] != '0')                  printf("%s", weishu[1]);              break;          case 2:              if (str[0] != '0')                  printf("%s", weishu[0]);              break;          case 1:              break;          default:              return;          }          read(++str);      }  } |

Q366.(10分)编写一个使用指针的C函数，交换数组a和数组b的对应元素。

int a[5]={1,2,3,4,5};

int b[5]={10,20,30,40,50};

\*\*输出格式要求："a[%d]=%2d, " "b[%d]=%2d, "

程序运行示例如下：

a[0]=10, a[1]=20, a[2]=30, a[3]=40, a[4]=50, b[0]= 1, b[1]= 2, b[2]= 3, b[3]= 4, b[4]= 5,

### 参考答案

|  |
| --- |
| #include <stdio.h>  swap(pa,pb)      int \*pa,\*pb;  {      int temp;      temp=\*pa; \*pa=\*pb; \*pb=temp;  }  main()  {      int a[5]={1,2,3,4,5};      int b[5]={10,20,30,40,50};      int i;      for(i=0;i<5;i++)          swap(&a[i],&b[i]);      for(i=0;i<5;i++)          printf("a[%d]=%2d, ",i,a[i]);      for(i=0;i<5;i++)          printf("b[%d]=%2d, ",i,b[i]);  } |

Q409.(10分)由二个平方三位数获得三个平方二位数。已知两个平方三位数abc和xyz，其中未必是不同的；而ax、by、cz是三个平方二位数。请编程求三位数abc和xyz。

\*\*输出格式要求："The possible perfect squares combinations are:\n" " %d and %d\n"

程序运行示例如下：

The possible perfect squares combinations are:

400 and 900

841 and 196

### 参考答案

|  |
| --- |
| #include "math.h"  main()  {void f();   int i,t,a[3],b[3];   printf("The possible perfect squares combinations are:\n");   for(i=11;i<=31;i++)     for(t=11;t<=31;t++)     {f(i\*i,a);      f(t\*t,b);      if(sqrt(a[0]\*10+b[0])==(int)sqrt(a[0]\*10+b[0])           &&sqrt(a[1]\*10+b[1])==(int)sqrt(a[1]\*10+b[1])           &&sqrt(a[2]\*10+b[2])==(int)sqrt(a[2]\*10+b[2]))           printf(" %d and %d\n",i\*i,t\*t);     }  }  void f(n,s)    int n,\*s;   {int k;    for(k=1000;k>=10;s++)    { \*s=(n%k)/(k/10);        k/=10;    }  } |

Q1135.(10分)根据如下性质，设计函数MaxCommonFactor()，计算两个正整数的最大公约数。

性质1：当a>b时，计算a与b的公约数等价于计算a-b与b的公约数。

性质2：当a<b时，计算a与b的公约数等价于计算b-a与b的公约数。

性质3：当a=b时，a与b的公约数等于a或b。

请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

int MaxCommonFactor(int a, int b);

main()

{

int a, b, x;

printf("Input a,b:");

scanf("%d,%d", a, b);

x = MaxCommonFactor(int a, int b);

if (x < 0) printf("Input Error!\n");

printf("%d\n", x);

}

int MaxCommonFactor(int a, int b)

{

if (a <= 0 || b <= 0)

return -1;

while if (a != b);

{

if (a > b)

return a - b;

else if (b > a)

return b - a;

else return a;

}

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    int MaxCommonFactor(int a, int b);    main()  {      int a, b, x;      printf("Input a,b:");      scanf("%d,%d", &a, &b);      x = MaxCommonFactor(a, b);      if (x < 0)          printf("Input Error!\n");      else          printf("%d\n", x);  }  int MaxCommonFactor(int a, int b)  {      if (a <= 0 || b <= 0)          return -1;      while (a != b)      {          if (a > b)              a = a - b;          else if (b > a)              b = b - a;      }      return a;  } |

Q566.(10分)求以下分数序列通项式，并求出前 n项之和。要求：求和的结果通过函数返回数值。

如： n=10， 输出： 16.479905

2/1,3/2,5/3,8/5,13/8,21/24

\*\*输入格式要求："%d" 提示信息："请输入n的值：\n"

\*\*输出格式要求："n项之和为：%lf\n"

程序运行示例如下：

请输入n的值：

20

n项之和为：32.660261

### 参考答案

|  |
| --- |
| #include <stdio.h>  double fun(int i)  {      int a, b, c, k;      double s = 0.0;      a = 2;      b = 1;        for (k = 0; k < i; k++)      {          s = s + (double)a / b;          c = a;          a = a + b;          b = c;      }      return s;  }    main()  {      int n;      double sum;      printf("请输入n的值：\n");      scanf("%d", &n);      sum = fun(n);      printf("n项之和为：%lf\n", sum);  } |

Q314.(15分)编程打印200~300之间所有素数。

要求判断一个数是否是素数用函数实现，打印结果请在主函数实现。

判断一个数是否是素数函数原型为:

int fun(int m);

参数说明：

参 数：m是要进行判断的数；

返回值：若此数是素数返回值为1；否则返回值为0

\*\*\*\*输入提示信息格式: 无

\*\*\*\*输入数据格式要求: 无

\*\*\*\*输出数据格式要求: "%d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

### 参考答案

|  |
| --- |
| #include<stdio.h>  int fun(int m);  main()  {      int m, flag;        for (m = 200; m <= 300; m++)  //2      {            flag = fun(m);       //3          if (flag)            //1          {              printf("%d\n", m);    //1          }      }  }  int fun(int m)  {      int i, flag = 1;             //1        for (i = 2; i < m; i++)      //2      {          if (m % i == 0)      //2          {              flag = 0;    //1              break;          }      }      return flag;                 //1  } |

Q309.(20分)如果一个正整数m的所有小于n的不同因子（包括1）加起来正好等于n本身，那么就被称它为完全数。它是指这样的一些特殊的自然数，它所有的真因子（即除了自身以外的约数）的和，恰好等于它本身。

注意：1没有真因子，所以不是完全数。例如，6就是一个完全数，是因为6 = 1 + 2 + 3。

请编写一个程序，找出整数m 以内的所有完全数。m由键盘输入。

要求：输入、输出由主函数实现，判断完全数用函数实现。

判断完全数的函数原型如下，若函数返回0，则代表不是完全数，若返回1，则代表是完全数。

int IsPerfect(int x);

\*\*要求输入提示信息为: "Input m:\n"

\*\*要求输入格式为: "%d"

\*\*要求输出格式为

"%d is a perfect number\n"

"%d is not a perfect number\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int IsPerfect(int x);  main()  {      int m;      printf("Input m:\n");      scanf("%d", &m);//1      if (IsPerfect(m))//2      {          printf("%d is a perfect number\n", m);//1      }      else      {          printf("%d is not a perfect number\n", m);//1      }  }  int IsPerfect(int x)//1  {      int i;      int total = 0;//1      for (i=1; i<x; i++)//2      {          if (x%i == 0)//1          {              total = total + i;//1          }      }      if (total == x)//1      {          return 1;//1      }      else      {          return 0;//1      }  } |

Q1156.(10分)在下面这个程序中，要求其实现的功能是对用户输入的任意整数计算其阶乘，输入小于等于-1的数时退出程序。下面给出的是一个有错误的程序，请找出其中的错误，并改正之。

#include <stdio.h>

#define END -1;

long Factorial(int x);

main()

{

int x;

while (1);

{

printf("input x:);

scanf("%d", x);

if (x <= END)

break

else

printf("%d! = %d\n", x, Factorial(x));

}

}

long Factorial(int x)

{

int i;

int result;

for (i=1, i<=x, i++)

result \*= i;

}

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define END  -1  long Factorial(int x);  main()  {      int x;      while (1)      {          printf("input x:");          scanf("%d", &x);          if (x <= END)              break;          else              printf("%d! = %ld\n", x, Factorial(x));      }  }    long Factorial(int x)  {      int i;      long result = 1;        for (i=1; i<=x; i++)          result \*= i;      return result;  } |

Q605.(10分)编程输出1000以内所有的完数，并输出其所有因子。所谓完数，即一个数的所有因子（除其自身）之和恰好等于其自身。如6就是一个完数，6=1+2+3。

\*\*输出格式要求："%5d"

程序运行示例如下：

6

1 2 3

28

1 2 4 7 14

496

1 2 4 8 16 31 62 124 248

### 参考答案

|  |
| --- |
| #include <stdio.h>  int fun1(int m);  void fun2(int m, int flag);  int main()  {      int m, flag;      for (m = 2; m <= 1000; m++)      {          flag = fun1(m);          fun2(m, flag);      }      return 0;  }  int fun1(int m)  {      int i, sum, flag;      for (i = 1, sum = 0; i < m; i++)      {          if (m % i == 0)          {              sum = sum + i;          }      }      if (sum == m)      {          flag = 1;      }      else      {          flag = 0;      }      return flag;  }  void fun2(int m, int flag)  {      int i, sum;      if (flag == 1)      {          printf("\n%5d\n", m);          for (i = 1; i < m; i++)          {              if (m % i == 0)              {                  printf("%5d", i);              }          }      }  } |

Q558.(10分)从一个字符串删除元音字母，例如，如果输入的字符串是“C programming”，那么输出将是“C prgrmmng”。

\*\*输入格式要求：提示信息："Enter a string to delete vowels\n"

\*\*输出格式要求："String after deleting vowels: %s\n"

程序运行示例如下：

Enter a string to delete vowels

c programming

String after deleting vowels: c prgrmmng

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>    int check\_vowel(char);    int main()  {      char s[100], t[100];      int i, j = 0;        printf("Enter a string to delete vowels\n");      gets(s);        for (i = 0; s[i] != '\0'; i++)      {          if (check\_vowel(s[i]) == 0)        //not a vowel          {              t[j] = s[i];              j++;          }      }        t[j] = '\0';        strcpy(s, t);    //We are changing initial string        printf("String after deleting vowels: %s\n", s);        return 0;  }      int check\_vowel(char c)  {      switch (c)      {      case 'a':      case 'A':      case 'e':      case 'E':      case 'i':      case 'I':      case 'o':      case 'O':      case 'u':      case 'U':          return 1;      default:          return 0;      }  } |

Q1322.(10分)下面程序用于计算两个正整数的最小公倍数。

|  |
| --- |
| #include <stdio.h>    int MinCommonMultiple(int a, int b)    main()  {      int a, b, x;        printf("Input a,b:");      scanf("%d,%d",&a,&b);        x = MinCommonMultiple(int a,int b);      printf("MinCommonMultiple = %d\n", x);  }    int MinCommonMultiple(int a, int b);  {      int i;        for (i=1; i<b; i++)      {          if ((i\*a) % b = 0)              return i\*a;      }  return 0;  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>    int MinCommonMultiple(int a, int b) ;    main()  {      int a, b, x;        printf("Input a,b:");      scanf("%d,%d",&a,&b);        x = MinCommonMultiple(a,b);      printf("MinCommonMultiple = %d\n", x);  }  int MinCommonMultiple(int a, int b)  {      int i;        for (i=1; i<b; i++)      {          if ((i\*a) % b == 0)              return i\*a;      }      return b\*a;  } |

Q1117.(10分)函数fun功能：由小到大输出两实数。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

fun();

}

fun(float a, float b)

{

float t;

scanf("%f%f", &a, &b);

if (a < b)

{

t = a;

a = b;

b = t;

}

printf("%5.2,%5.2f\n", &a, &b);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    fun()  {      float t;      float a, b;        scanf("%f%f", &a, &b);      if (a < b)      {          t = a;          a = b;          b = t;      }      printf("%5.2f,%5.2f\n", a, b);  }    main()  {      fun();  } |

Q291.(20分)编写一个找出大于给定整数m的最小素数的函数。所谓素数是指这个整数只能被1和自身整除。要求在主函数中输入一个整数m，调用子函数IsPrime()找出其最小素数，并输出结果。

函数原型：int IsPrime(int x)

\*\*\*\*要求输入提示信息为：Please input n:

\*\*\*\*输出格式要求为："%d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int IsPrime(int x)  {      int i,flag=1;      int squareRoot=sqrt(x);      if(x<=1) //1      {          flag=0;      }      for(i=2; i<=squareRoot&&flag; i++)   //2      {          if(x%i==0)  //1          {              flag=0;          }      }      return flag;  }  main()  {      int n,i;      printf("Please input n:");      scanf("%d",&n);      i=n+1;      while(i>=n)  //1      {          if(IsPrime(i))  //1          {              printf("%d\n",i);   //1              break;          }          else          {              i++;    //1          }      }  } |

Q204.(15分)编程打印200~300之间所有素数。

要求判断一个数是否是素数用函数实现，打印结果请在主函数实现。

判断一个数是否是素数函数原型为:

int fun(int m);

参数说明：

参 数：m是要进行判断的数；

返回值：若此数是素数返回值为1；否则返回值为0

\*\*\*\*输入提示信息格式: 无

\*\*\*\*输入数据格式要求: 无

\*\*\*\*输出数据格式要求: "%d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

### 参考答案

|  |
| --- |
| #include<stdio.h>  int fun(int m);  main()  {      int m,flag;        for(m=200;m<=300;m++)         //2      {            flag=fun(m);         //3          if(flag)             //1          {              printf("%d\n",m);     //1          }      }  }  int fun(int m)  {      int i,flag=1;                //1        for(i=2;i<m;i++)             //2      {          if(m%i==0)           //2          {              flag=0;      //1              break;          }      }      return flag;                 //1  } |

Q324.(10分)求三角形面积。输入三角形的三个边长，计算三角形的面积。设三角形的三个边长分别为a、b、c，为简单起见，我们认为输入的三个边长数据是正确的，可以组成一个三角形。

\*\*输入格式要求："%f,%f,%f" 提示信息："Enter 3 floats:"

\*\*输出格式要求："area=%f\n"

程序运行示例如下：

Enter 3 floats:2,2,3

area=1.984313

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  {      float a,b,c,s,area;      printf("Enter 3 floats:");      scanf("%f,%f,%f",&a,&b,&c);      s=(a+b+c)/2;      area=sqrt(s\*(s-a)\*(s-b)\*(s-c));      printf("area=%f\n",area);  } |

Q1124.(10分)求用户输入的两个数的商，程序运行时，以如下格式输入数据：

Input two integers:4 2↙

请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

int a, b, c;

printf("Input two integers:");

scanf("%d,%d", &a, &b);

c = a\b;

printf("The quotient of a and b is :%d", c);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int a, b, c;        printf("Input two integers:");      scanf("%d%d", &a, &b);      c = a / b;      printf("The quotient of a and b is :%d", c);  } |

Q1014.(10分)此程序是输入一个华氏温度与摄氏温度的转化问题。阅读程序，找出其中的错误，并改正之。

#include <stdio.h>

main()

{

double F,c;

scanf("%f",F);

c=5/9(F-32);

printf('F=%2,2f\n',F,'c=%2,2\n',C);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      double F, c;      scanf("%lf", &F);      c = 5.0 / 9.0 \* (F - 32);      printf("F = %2.2f\nc = %2.2f\n", F, c);  } |

Q1351.(10分)一只大象口渴了，要喝20升水才能解渴，但现在只有一个深h厘米，底面半径为r厘米的小圆桶(h和r都是整数)。问大象至少要喝多少桶水才会解渴。(设PAI=3.14159)(输入输出样例中的◊代表一个空格)

输入一行：小圆桶的深h，和底面半径r，单位都是厘米。

输出两行：第一行提示输入语句；第二行为大象至少要喝水的桶数。

输入输出样例：

输入：

height◊=◊23,◊radius◊=◊11

输出：

please◊input◊the◊height◊and◊the◊radius:

3

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define PI  3.14159  #define RATE    0.001  #define NEED    20    int main()  {      int h, r;      double volum;        printf("please input the height and the radius:\n");      scanf("height = %d, radius = %d", &h, &r);        volum = PI \* r \* r \* h \* RATE;      printf("%d", (int)(20 / volum) + 1);        return 0;  } |

Q479.(10分)输入一个整数，截取它对应的二进制位中从右到左的第8-11位（最右边为第0位）。

\*\*输入格式要求："%d" 提示信息："请输入一个整数:"

\*\*输出格式要求："它的从右开始的第11-8位为:\n" "%d "

程序运行示例如下：

请输入一个整数:2997

它的从右开始的第11-8位为:

1 0 1 1

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define MASK 0xf  //MASK=(00001111)    int main()  {      int intNum, i, t;        printf("请输入一个整数:");      scanf("%d", &intNum);      intNum >>= 8;      intNum = intNum & MASK;      printf("它的从右开始的第11-8位为:\n");      for (i = 1; i <= 4; i++)      {          t = intNum;          t >>= (4 - i);          t = t & 1;          printf("%d ", t > 0 ? 1 : 0);      }        return 0;  } |

Q1122.(10分)判断输入的三个数是否相等。算法如下，请改正程序中的错误，使它能得出正确的结果。

|  |
| --- |
| #include <stdio.h>    main()  {      int a, b, c;        scanf("%d,%d,%d", a, b, c);      if (a = = b = = c)          printf("The three number is equal!!!");      else          printf("The three number isn't equal!!!");  } |

参考答案

#include <stdio.h>

main()

{

    int a, b, c;

    scanf("%d,%d,%d", &a, &b, &c);

    if (a == b && b == c)

        printf("The three number is equal!!!");

    else

        printf("The three number isn't equal!!!");

}

Q585.(10分)编写一个程序，输入一个3位正整数，要求逆序输出对应的数。例如：输入123，则输出321。

\*\*输入格式要求："%d" 提示信息："Please enter a positive integer: "

\*\*输出格式要求："%d-->%d\n"

程序运行示例如下：

Please enter a positive integer: 345

345-->543

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>    int main()  {      int n, i, j, k, m;      printf("Please enter a positive integer: ");      scanf("%d", &n);      i = n / 100;      m = n - 100 \* i;      j = m / 10;      m = m - 10 \* j;      k = m;      m = 100 \* k + 10 \* j + i;      printf("%d-->%d\n", n, m);      return 0;  } |

Q325.(10分)求前驱字符和后继字符。输入一个字符，找出它的前驱和后继字符，并按ASCII码值，按从小到大顺序输出这三个字符及其对应的ASCII码值。（注：一个字符的前驱字符是指在ASCII码表中，排列在该字符前面的一个字符，即比该字符的ASCII码值小1的字符。一个字符的后继字符是指在ASCII码表中，排列在该字符后面的一个字符。）

\*\*输入格式要求："%c" 提示信息："Enter a character:"

\*\*输出格式要求："%c %c %c\n" "%d %d %d\n"

程序运行示例如下：

Enter a character:G

F G H

70 71 72

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char c,c1,c2;      printf("Enter a character:");      c=getchar();      c1=c-1;      c2=c+1;      printf("%c %c %c\n",c1,c,c2);      printf("%d %d %d\n",c1,c,c2);  } |

Q473.(10分)

编程输出如下菜单界面：

=======大学信息管理系统=======

------------------------------

   办公室管理      财务管理

   教务管理        图书管理

   科研管理        设备管理

   人事管理        后勤管理

   退出系统

------------------------------

请您在上述功能中选择（0——8）：

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      printf("=======大学信息管理系统======= \n");      printf("------------------------------\n");      printf("1. 办公室管理     5. 财务管理 \n");      printf("2. 教务管理       6. 图书管理 \n");      printf("3. 科研管理       7. 设备管理 \n");      printf("4. 人事管理       8. 后勤管理 \n");      printf("0. 退出系统\n");      printf("------------------------------\n");      printf("请您在上述功能中选择（0——8）:\n");        return 0;  } |

Q1116.(10分)求输入两个数的和、差、积、商和余数。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

float a, b;

float sum, minus, product, quotient;

int remainder;

printf("\n请输入两个数:\n");

scanf("%f\n%f", a, &b);

sum = a + b;

minus = a - b;

product = a \* b;

quotient = a / b;

remainder = a % b;

printf("和为:%.2f\n", sum);

printf("差为:%.2f\n", minus);

printf("积为:%.2f\n", product);

printf("商为:%.2f\n", quotient);

printf("余数为:%d\n", remainder);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      float a, b;      float sum, minus, product, quotient;      int remainder;        printf("\n请输入两个数:\n");      scanf("%f%f", &a, &b);      sum = a + b;      minus = a - b;      product = a \* b;      quotient = a / b;      remainder = (int)a % (int)b;      printf("和为:%.2f\n", sum);      printf("差为:%.2f\n", minus);      printf("积为:%.2f\n", product);      printf("商为:%.2f\n", quotient);      printf("余数为:%d\n", remainder);  } |

Q323.(10分)编写程序，实现从键盘输入学生的三门课成绩，计算并输出其总成绩sum，平均成绩ave和总成绩除3的余数rem。

\*\*输入格式要求："%d%d%d" 提示信息："Enter three integer: "

\*\*输出格式要求："SUM = %4d\nAVERAGE = %f REMAINDER = %3d\n"

程序运行示例如下：

Enter three integer: 30 90 100

SUM = 270

AVERAGE = 90.000000 REMAINDER = 0

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a, b, c, sum, rem;      float ave;      printf("Enter three integer: ");      scanf("%d%d%d", &a, &b, &c);      sum = a + b + c;      ave = sum / 3.0;      rem = sum % 3;      printf("SUM = %4d\nAVERAGE = %f  REMAINDER = %3d\n", sum, ave, rem);  } |

Q477.(10分)编写程序计算圆的面积和周长。

\*\*输入格式要求："%f" 提示信息："请输入半径的值："

\*\*输出格式要求："半径为%5.2f的圆的面积为%5.2f,圆的周长为%5.2f"

示例运行如下：

请输入圆的半径：3.5

半径为 3.50的圆的面积为38.47,圆的周长为21.98

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define  PI  3.14    int main()  {      float area;      float radius, circleArea, circlePerimeter;        printf("请输入半径的值：");      scanf("%f", &radius);        circleArea = PI \* radius \* radius;      circlePerimeter = 2 \* PI \* radius;        printf("半径为%5.2f的圆的面积为%5.2f,圆的周长为%5.2f", radius, circleArea, circlePerimeter);        return 0;  } |

Q511.(10分)从键盘输入一个大写英文字母，将其转换为小写英文字母后，将转换后的小写英文字母及其十进制的ASCII码值显示到屏幕上。

\*\*输入格式要求：提示信息："Press a key and then press Enter:"

\*\*输出格式要求："%c, %d\n"

程序运行示例如下：

Press a key and then press Enter:B

b, 98

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char  ch;      printf("Press a key and then press Enter:");      ch = getchar();      ch = ch + 32;      printf("%c, %d\n", ch, ch); /\* 分别输出变量ch中的字符及其ASCII码值 \*/  } |

Q1128.(10分)由用户输入三个数据，算法如下，请改正程序中的错误，使它能得出正确的结果。

用户输入：12a4.2，程序输出：

The input integer is : 12

The input character is : a

The input float is : 4.200000

#include <stdio.h>

main()

{

int i;

char ch;

float f;

printf("Please input:\n");

scanf("%d %c%f", &i, ch, &f);

printf("The input integer is : %d \nThe input character is : %c\n", i, ch);

printf("The input float is : %f", f);

}

### 参考答案

|  |
| --- |
| #include <stdio.h>    main()  {      int i;      char ch;      float f;        printf("Please input:\n");      scanf("%d%c%f", &i, &ch, &f);      printf("The input integer is : %d \nThe input character is : %c\n", i, ch);      printf("The input float is : %f", f);  } |

Q1129.(10分)将字符串“Hello World”读入字符数组中去。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

#include <string.h>

main()

{

char ch[10];

scanf("%s", ch);

if (!strcmp(ch, "Hello World"))

printf("OK!");

else

printf("Error!");

}

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>    main()  {      char ch[10];        gets(ch);      if (!strcmp(ch, "Hello World"))          printf("OK!");      else          printf("Error!");  } |

Q375.(10分)请用递归的方法计算下列函数的值：px(x,n)=x-x^2+x^3-x^4+…((-1)^n-1)(x^n) n>0

\*\*输入格式要求："%lf%d" 提示信息："Enter X and N:"

\*\*输出格式要求："px=%lf\n"

程序运行示例如下：

Enter X and N:4 6

px=-3276.000000

### 参考答案

|  |
| --- |
| #include <stdio.h>  double px(double x,int n)  {      if(n==1)          return x;      else          return (x\*(1-px(x,n-1)));  }  main()  {      double x; int n;      printf("Enter X and N:");      scanf("%lf%d",&x,&n);      printf("px=%lf\n",px(x,n));  } |

Q1728.(10分)编写一个程序，打印所有的“水仙花数”。

所谓“水仙花数”，是指一个三位数，

其各位数字的立方和等于该数本身。

例如，153是“水仙花数”，因为153=13+33+53。

\*\*要求输入提示信息为：无输入提示信息和输入数据

\*\*要求输出格式为："%d\n"

### 参考答案

|  |
| --- |
| #include<stdio.h>  main()  {    int i,j,k,n;    for(n=100;n<1000;n++)    {      i=n/100;//1      j=(n-i\*100)/10;//1      k=n-i\*100-j\*10;//1      if(i\*i\*i+j\*j\*j+k\*k\*k==n)        printf("%d\n",n);      }    } |

Q1123.(10分)对候选人得票的统计程度。设由3个候选人，每次输入一个得票的候选人的名字，要求最后输入各人得票结果。（总票数为n）请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

struct person

{

char name[20];

int count;

}

person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

int i, j;

char leader\_name[20];

scanf("%d\n", &n);

for (i = 0; i < n; i++)

{

scanf("%s", leader\_name);

for (j = 0; j < 3; j++)

if (leader\_name == leader[j].name)

leader[j].count++;

}

printf("\n");

for (i = 0; i < 3; i++)

printf("%5s:%d\n", leader[i].name, leader[i].count);

}

### 参考答案

|  |
| --- |
| #include <string.h>    struct person  {      char name[20];      int count;  };    struct person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};    main()  {      int i, j, n;      char leader\_name[20];        scanf("%d\n", &n);      for (i = 0; i < n; i++)      {          scanf("%s", leader\_name);          for (j = 0; j < 3; j++)              if (!strcmp(leader\_name, leader[j].name))                  leader[j].count++;      }      printf("\n");      for (i = 0; i < 3; i++)          printf("%5s:%d\n", leader[i].name, leader[i].count);  } |

Q398.(10分)梯形法。用梯形法编程求函数f(a)=x 2 +2x+1

的定积分，∫ 2 0 f(x)dx

的值。

\*\*输入格式要求："%lf,%lf,%lf" 提示信息："Enter n,a,b:"

\*\*输出格式要求："Sum=%lf\n"

程序运行示例如下：

Enter n,a,b:10,0,2

Sum=10.604000

### 参考答案

|  |
| --- |
| #include "stdio.h"  main()  { int i;    double s=0,h,f0,f1,n,a,b;    printf("Enter n,a,b:");    scanf("%lf,%lf,%lf",&n,&a,&b);    h=(b-a)/n;    f0=a\*a+2\*a+1;    for(i=0;i<=n;i++)  { a=a+h;    f1=a\*a+2\*a+1;      s+=(f0+f1)\*h/2;      f0=f1;    }    printf("Sum=%lf\n",s);  } |

Q1685.(10分)用二维数组的列指针作为函数实参，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int \*a, int \*at, int m, int n);

void InputMatrix(int \*a, int m, int n);

void PrintMatrix(int \*at, int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输出提示信息和格式："The transposed matrix is:\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define M 10  #define N 10  void Transpose(int \*a, int \*at, int m, int n);  void InputMatrix(int \*a, int m, int n);  void PrintMatrix(int \*at, int n, int m);  int main()  {      int s[M][N], st[N][M], m, n;      printf("Input m, n:");      scanf("%d,%d", &m, &n);      InputMatrix(\*s, m, n);      Transpose(\*s, \*st, m, n);      printf("The transposed matrix is:\n");      PrintMatrix(\*st, n,  m);      return 0;  }  /\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/  void Transpose(int \*a, int \*at, int m, int n)  {      int i, j;      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              at[j \* m + i] = a[i \* n + j];          }      }  }  /\* 函数功能：输入m\*n矩阵a的值 \*/  void InputMatrix(int \*a, int m, int n)  {      int i, j;      printf("Input %d\*%d matrix:\n", m, n);      for (i = 0; i < m; i++)      {          for (j = 0; j < n; j++)          {              scanf("%d", &a[i \* n + j]);          }      }  }  /\* 函数功能：输出n\*m矩阵at的值 \*/  void PrintMatrix(int \*at, int n, int m)  {      int i, j;      for (i = 0; i < n; i++)      {          for (j = 0; j < m; j++)          {              printf("%d\t", at[i \* m + j]);          }          printf("\n");      }  } |

Q1368.(10分)以下程序的功能是计算十个数据的平均值。找出其中的错误并改正之。

|  |
| --- |
| #include <stdio.h>  void main(void)  {      int i, sum;      float  avg;      int    sc[10], \*p = sc;        for (i=0, i<10, i++)      {              scanf("%d", p);              p++;              sum += \*p;      }      avg = sum / 10;      printf("avg=%f\n", avg);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  void main(void)  {      int i, sum=0;  float avg;  int sc[10], \*p=sc;    for (i=0; i<10; i++)  {          scanf("%d", p);          p++;          sum+= \*(p-1);  }  avg = (float)sum/ 10;  printf("avg=%f", avg);  } |

Q1273.(10分)

|  |
| --- |
| 该程序的作用是将一个字符串中的元音字母复制到另一个字符串，然后输出。改正错误。    #include<stdio.h>    main()  {      char str1[80], str2[80];        printf("输入一个字符串\n");      gets(str1);        Cpy(str1[], str2[]);        printf("输出一个字符串\n");      puts(str2);  }  void Cpy(char s[], char c[])  {      int i, j;        for(i=0; s[i] != '\0'; i++)      {          if(s[i]= 'a'||s[i]= 'A'||s[i]= 'e'||s[i]= 'E'||            s[i]= 'i'||s[i]= 'I'||s[i]= 'o'||s[i]= 'O'||s[i]= 'u'||s[i]= 'U')          {              s[i]=c[j];              j++;          }      }    } |

### 参考答案

|  |
| --- |
| #include<stdio.h>  void Cpy(char s[], char c[]);  main()  {      char str1[80], str2[80];        printf("输入一个字符串\n");      gets(str1);        Cpy(str1, str2); //改为：Cpy(str1, str2)        printf("输出一个字符串\n");      puts(str2);  }  void Cpy(char s[], char c[])  {      int i, j = 0; //改为：j=0;        for (i = 0; s[i] != '\0'; i++)      {          if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||              s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||              s[i] == 'u' || s[i] == 'U')              //将=改为：==          {              c[j] = s[i]; //改为:c[j] = s[i];              j++;          }      }      c[j] = '\0';  } |

Q110.(30分)用一个整型数组feedback保存调查的20个反馈意见，

其中反馈意见是1-10范围中的一个整数。

用函数编程计算反馈意见的众数。

众数是数组中出现次数最多的那个数。

（假设不会发生两个或两个以上的反馈意见出现次数相同的情况）

要求：

（1）任意从键盘输入20个值在1-10范围中的整数，

编写函数Mode，求众数，

主函数调用函数Mode，并输出众数。

（2）计算n个数的众数的函数原型：

int Mode(int answer[], int n)

（3）\*\*输入提示信息格式要求："Input 20 feedbacks:\n"

\*\*输出格式要求："Mode value=%d\n"

（4）不要使用结构体

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  M   20  #define  N   11  int Mode(int answer[], int n);  main()  {      int  i, feedback[M];//1      int modeval;      printf("Input 20 feedbacks:\n");      for (i=0; i<M; i++)//2      {          scanf("%d", &feedback[i]);//1      }      modeval=Mode(feedback, M);//2        printf("Mode value=%d\n", modeval);//1  }    int Mode(int answer[], int n) //1  {      int  i, grade, max = 0, modeValue = 0, count[N] = {0};//1      for (i=0; i<n; i++)//2      {          count[answer[i]]++;//2      }         for (grade=1; grade<=N-1; grade++)//2      {          if (count[grade] > max)//2          {              max = count[grade];//1                          modeValue = grade;//1          }      }        return modeValue;//1  } |

Q546.(10分)请编程实现按奥运会参赛国国名在字典中的顺序对其入场次序进行排序。假设参赛国不超过150个。要求通过二维字符数组传参调用函数实现排序,输入和输出在主函数中实现。

\*\*输入格式要求："%d" 提示信息："How many countries?" "Input their names:\n"

\*\*输出格式要求："Sorted results:\n"

程序的运行示例如下：

How many countries? 5

Input their names:

America

England

Australia

Sweden

Finland

Sorted results:

America

Australia

England

Finland

Sweden

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #include  <string.h>  #define   MAX\_LEN  10                       /\* 字符串最大长度 \*/  #define   N         150                     /\* 字符串个数 \*/  void SortString(char str[][MAX\_LEN], int n);  int main()  {      int    i, n;      char   name[N][MAX\_LEN];                /\* 定义二维字符数组 \*/      printf("How many countries?");      scanf("%d", &n);      getchar();                                /\* 读走输入缓冲区中的回车符 \*/      printf("Input their names:\n");      for (i = 0; i < n; i++)      {          gets(name[i]);                          /\* 输入n个字符串 \*/      }      SortString(name, n);                    /\* 字符串按字典顺序排序 \*/      printf("Sorted results:\n");      for (i = 0; i < n; i++)      {          puts(name[i]);                      /\* 输出排序后的n个字符串 \*/      }      return 0;  }  /\* 函数功能：交换法实现字符串按字典顺序排序 \*/  void SortString(char str[][MAX\_LEN], int n)  {      int    i, j;      char  temp[MAX\_LEN];      for (i = 0; i < n - 1; i++)      {          for (j = i + 1; j < n; j++)          {              if (strcmp(str[j], str[i]) < 0)              {                  strcpy(temp, str[i]);                  strcpy(str[i], str[j]);                  strcpy(str[j], temp);              }          }      }  } |

Q565.(30分)程序中函数 transform()的功能: 将一个由二进制数字字符组成的字符串转换为与其值相等的十进制整数。规定输入的字符串最多只能包含10位二进制数字字符。

\*\*输入格式要求：

要求对输入的二进制数字字符组成的字符串进行长度和内容确为二进制的判断。

提示信息："输入一个二进制字符串（10位）:" "错误：字符串太长\n" "错误:%c 不是一个二进制字符\n"

\*\*输出格式要求："输入的二进制字符串为" "\n%s 转换成十进制整数为：%d\n"

程序运行示例如下：

输入一个二进制字符串（10位）:10010

输入的二进制字符串为10010

10010 转换成十进制整数为：18

提示：exit(0)函数在stdlib.h库中定义。

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #include <stdlib.h>    int transform(char ch[])  {      int n=0,i=0;      while (ch[i] != '\0')      {          n = n \* 2 + ch[i] - '0';          i++;      }      return n;  }    void main()  {      char s[10];      int i;      int n;      printf("输入一个二进制字符串（10位）:");      gets(s);      if (strlen(s) > 10)      {          printf("错误：字符串太长\n");          exit(0);      }      for (i=0;s[i]!='\0';i++)      {          if (s[i] != '0' && s[i] != '1')          {              printf("错误:%c 不是一个二进制字符\n", s[i]);              exit(0);          }      }      printf("输入的二进制字符串为");      puts(s);      n = transform(s);      printf("\n%s 转换成十进制整数为：%d\n", s, n);  } |

Q1718.(10分)身高预测。每个做父母的都关心自己孩子成人后的身高，据有关生理卫生知识与数理统计分析表明，影响小孩成人后的身高的因素包括遗传、饮食习惯与体育锻炼等。小孩成人后的身高与其父母的身高和自身的性别密切相关。

设faHeight为其父身高，moHeight为其母身高，身高预测公式为

男性成人时身高 = (faHeight + moHeight) × 0.54 cm

女性成人时身高 = (faHeight × 0.923 + moHeight) / 2 cm

此外，如果喜爱体育锻炼，那么可增加身高2%；如果有良好的卫生饮食习惯，那么可增加身高1.5%。

请编程从键盘输入用户的性别（用字符型变量sex存储，输入字符F表示女性，输入字符M表示男性）、父母身高（用实型变量存储，faHeight为其父身高，moHeight为其母身高）、是否喜爱体育锻炼（用字符型变量sports存储，输入字符Y表示喜爱，输入字符N表示不喜爱）、是否有良好的饮食习惯等条件（用字符型变量diet存储，输入字符Y表示良好，输入字符N表示不好），利用给定公式和身高预测方法对身高进行预测。

运行示例：

Are you a boy(M) or a girl(F)?F↙

Please input your father's height(cm):182↙

Please input your mother's height(cm):162↙

Do you like sports(Y/N)?N↙

Do you have a good habit of diet(Y/N)?Y↙

Your future height will be 167(cm)

### 参考答案

|  |
| --- |
| #include  <stdio.h>  main()  {      char sex;                    /\* 孩子的性别 \*/      char sports;            /\* 是否喜欢体育运动 \*/      char diet;          /\* 是否有良好的饮食习惯 \*/      float myHeight;     /\* 孩子身高 \*/      float faHeight;     /\* 父亲身高 \*/      float moHeight;     /\* 母亲身高 \*/      printf("Are you a boy(M) or a girl(F)?");      scanf(" %c", &sex);   /\* 在%c前加一个空格，将存于缓冲区中的回车符读走 \*/      printf("Please input your father's height(cm):");      scanf("%f", &faHeight);      printf("Please input your mother's height(cm):");      scanf("%f", &moHeight);      printf("Do you like sports(Y/N)?");      scanf(" %c", &sports);/\* %c前加一空格，读走缓冲区中的回车符 \*/      printf("Do you have a good habit of diet(Y/N)?");      scanf(" %c", &diet); /\* %c前加一空格，读走缓冲区中的回车符 \*/      if (sex == 'M' || sex == 'm')          myHeight = (faHeight + moHeight) \* 0.54;      else          myHeight = (faHeight \* 0.923 + moHeight) / 2.0;      if (sports == 'Y' || sports == 'y')          myHeight = myHeight \* (1 + 0.02);      if (diet == 'Y' || diet == 'y')          myHeight = myHeight \* (1 + 0.015);      printf("Your future height will be %.0f(cm)\n", myHeight);  } |

Q1717.(10分)编程从键盘输入一个小写英文字母，将其转换为大写英文字母后，将转换后的大写英文字母及其十进制的ASCII码值显示到屏幕上。

输入提示信息："Press a key and then press Enter:"

输入字符用getchar()

输出提示信息和格式："%c, %d\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char  ch;      printf("Press a key and then press Enter:");      ch = getchar();      ch = ch - 32;      printf("%c, %d\n", ch, ch); /\* 分别输出变量ch中的字符及其ASCII码值 \*/  } |

Q1716.(10分)编程计算并输出球的体积和表面积，球的半径r的值由用户从键盘输入。圆周率取3.14159

输入提示信息："Input r:"

输入格式："%lf"

输出格式：

"surface = %lf\n"

"volume = %lf\n"

### 参考答案

|  |
| --- |
| #include  <math.h>  #include  <stdio.h>  #define  PI  3.14159       /\* 定义宏常量PI \*/  main()  {      double r, surface, volume;      printf("Input r:");      scanf("%lf", &r);      surface = 4 \* PI \* pow(r, 2);      volume = 4.0 / 3.0 \* PI \* pow(r, 3);      printf("surface = %lf\n", surface);      printf("volume = %lf\n", volume);  } |

Q308.(10分)体型判断。医务工作者经广泛的调查和统计分析，根据身高与体重因素给出了以下按“体指数”进行体型判断的方法：

体指数t = 体重w /(身高h\*h) ，其中w单位为千克，h单位为米

当 t<18 时，为低体重；

当 18≤t<25 时，为正常体重；

当 25≤t<27 时，为超重体重；

当 t≥27 时，为肥胖。

根据上述给定的公式计算体指数t，然后判断你的体重属于何种类型。

\*\*输入提示信息格式："Please enter h,w:\n"

\*\*输入数据格式要求："%f,%f"

\*\*输出数据格式要求:

当 t<18 时，输出："Lower weight!\n"

当 18≤t<25 时，输出："Standard weight!\n"

当 25≤t<27 时，输出："Higher weight!\n"

当 t≥27 时， 输出："Too fat!\n"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  int main()  {      float  h, w, t;      printf("Please enter h,w:\n");      scanf("%f,%f", &h, &w);          //1      t = w / (h \* h);                 //1      if (t < 18)                      //1      {          printf("Lower weight!\n");      }      if (t >= 18 && t < 25)           //1      {          printf("Standard weight!\n");      }      if (t >= 25 && t < 27)           //1      {          printf("Higher weight!\n");      }      if (t >= 27)                     //1      {          printf("Too fat!\n");      }      return 0;  } |

Q198.(20分)如果一个正整数m的所有小于n的不同因子（包括1）加起来正好等于n本身，那么就被称它为完全数。它是指这样的一些特殊的自然数，它所有的真因子（即除了自身以外的约数）的和，恰好等于它本身。

注意：1没有真因子，所以不是完全数。例如，6就是一个完全数，是因为6 = 1 + 2 + 3。

请编写一个程序，找出整数m 以内的所有完全数。m由键盘输入。

要求：输入、输出由主函数实现，判断完全数用函数实现。

判断完全数的函数原型如下，若函数返回0，则代表不是完全数，若返回1，则代表是完全数。

int IsPerfect(int x);

\*\*要求输入提示信息为: "Input m:\n"

\*\*要求输入格式为: "%d"

\*\*要求输出格式为

"%d is a perfect number\n"

"%d is not a perfect number\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int IsPerfect(int x);  main()  {      int m;      printf("Input m:\n");      scanf("%d", &m);//1      if (IsPerfect(m))//2      {          printf("%d is a perfect number\n", m);//1      }      else      {          printf("%d is not a perfect number\n", m);//1      }  }  int IsPerfect(int x)//1  {      int i;      int total = 0;//1      for (i=1; i<x; i++)//2      {          if (x%i == 0)//1          {              total = total + i;//1          }      }      if (total == x)//1      {          return 1;//1      }      else      {          return 0;//1      }  } |

Q1720.(10分)从键盘任意输入一个整数m，若m不是素数，则对m进行质因数分解，并将m表示为质因数从小到大顺序排列的乘积形式输出，否则输出"It is a prime number"。例如，用户输入90时，程序输出90 = 2 \* 3 \* 3 \* 5；用户输入17时，程序输出"It is a prime number"。

输入提示信息："Input m:"

输入格式："%d"

输出格式：

是素数时输出"It is a prime number\n"

否则输出用"%d = "，"%d \* "

运行示例1：

Input m:90↙

90 = 2 \* 3 \* 3 \* 5

运行示例2：

Input m:13↙

It is a prime number

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  int IsPrime(int x);  void OutputPrimeFactor(int x);  int main()  {      int m;      printf("Input m:");      scanf("%d", &m);      if (IsPrime(m))  /\* 素数判定 \*/      {          printf("It is a prime number\n");      }      else      {          printf("%d = ", m);          OutputPrimeFactor(m); /\* 输出x的质因数连乘 \*/      }      return 0;  }  /\* 函数功能：判断x是否是素数，若函数返回0，则表示不是素数，若返回1，则代表是素数 \*/  int IsPrime(int x)  {      int i, flag = 1;      int squareRoot = (int)sqrt(x);      if (x <= 1)   flag = 0;          /\* 负数、0和1都不是素数 \*/      for (i = 2; i <= squareRoot && flag; i++)      {          if (x % i == 0) flag = 0;   /\* 若能被整除，则不是素数 \*/      }      return flag;  }  /\* 函数功能：输出x的质因数连乘 \*/  void OutputPrimeFactor(int x)  {      int i;      for (i = 2; i < x; i++)      {          if (x % i == 0)          {              printf("%d \* ", i);              OutputPrimeFactor(x / i); /\* 递归调用该函数 \*/              return;                    /\* 不可以使用break \*/          }      }      printf("%d", x);                 /\* 输出最后一个因子（质因数，不能再分解）\*/  } |

Q594.(10分)编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #define N 100  int getline(char \*s, int len);  int main()  {      char line[N], b[20], \*ptr1, \*ptrb;      int n = 0, j, a[50];      printf("Please enter a string:");      getline(line, N);      ptr1 = line;      while (\*ptr1 != '\0')      {          ptrb = b;          for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)          {              \*ptrb = \*ptr1;          }          if (j)          {              \*ptrb = '\0';              \*(a + n) = atoi(b);              n++;          }          if (\*ptr1 != '\0')ptr1++;      }      printf("the result of output:\n");      for (j = 0; j < n; j++)      {          printf("%10d\n", \*(a + j));      }      return 0;  }    int getline(char \*s, int len)  {      int c;      char \*p = s;      while (--len > 0 && (c = getchar()) != '\n')      {          \*s++ = c;      }      \*s = '\0';      return s - p;  } |

Addcdc

Q307.(15分)利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

参考答案

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

Q133.(5分)设计一个函数MinCommonMultiple,其功能是计算两个正整数的最小公倍数。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include <stdio.h>

int MinCommonMultiple(int a,int b)

main()

{

int a,b,x;

printf("Input a,b:");

scanf("%d,%d",a,b);

x = MinCommonMultiple(a,b);

printf("MinCommonMultiple = %d\n", x);

}

int MinCommonMultiple(int a,int b);

{

int i;

for(i=1; i<a\*b; i++)

{

if(i%a == 0 || i%b == 0)

return i;

}

}

参考答案

int MinCommonMultiple(int a,int b);

main()

{

int a,b,x;

printf(&quot;Input a,b:&quot;);

scanf(&quot;%d,%d&quot;,&amp;a,&amp;b);

x = MinCommonMultiple(a,b);

printf(&quot;MinCommonMultiple = %d\n&quot;, x);

}

int MinCommonMultiple(int a,int b)

{

int i;

for(i=1; i&lt;a\*b; i++)

{

if(i%a == 0 &amp;&amp; i%b == 0)

return i;

}

}

Q173.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

参考答案

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q191.(15分)用二维数组编程计算并输出n(n<20)行杨辉三角形。

\*\*输入提示信息要求："Input n(n<20):\n"

\*\*输入格式："%d"

\*\*输出要求："%4d"

每输出一行后就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include<stdio.h>

#define N 20

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

main() {

int a[N][N] = {0}, n;//1

printf("Input n(n<20):\n");

scanf("%d", &n);//1

CaculateYH(a, n);//1

PrintYH(a, n); //1

}

void CaculateYH(int a[][N], int n) {

int i, j;

for (i = 0; i < n; i++) //1

{

a[i][0] = 1;//1

a[i][i] = 1;//1

}

for (i = 2; i < n; i++)//1

{

for (j = 1; j <= i - 1; j++) //1

{

a[i][j] = a[i - 1][j - 1] + a[i - 1][j];//1

}

}

}

void PrintYH(int a[][N], int n) {

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j <= i; j++) //1

{

printf("%4d", a[i][j]);//1

}

printf("\n");//1

}

}

Q235.(10分)用函数编程计算两整数的最大值，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大值。

\*\*输入格式要求："%d%d" 提示信息："Input a,b:"

\*\*输出格式要求："max = %d\n"

程序运行示例如下：

Input a,b:5 8

max = 8

参考答案

#include <stdio.h>

int Max(int a, int b);

int main()

{

int a, b;

printf("Input a,b:");

scanf("%d%d", &a, &b);

printf("max = %d\n", Max(a, b));

return 0;

}

/\* 函数功能：计算a和b的最大值 \*/

int Max(int a, int b)

{

return a > b ? a : b;

}

Q602.(10分)一辆卡车违反了交通规则，撞人后逃逸。现场有三人目击该事件，但都没有记住车号，只记住车号的一些特征。甲说：车号的前两位数字是相同的；乙说：车号的后两位数字是相同的，但与前两位不同；丙是位数学家，他说：4位的车号正好是一个整数的平方。现在请根据以上线索帮助警方找出车号以便尽快破案。

\*\*输出格式要求："The number is:%d\n"

参考答案

#include<stdio.h>

int main()

{

int a, b, n;

for (a = 1; a <= 9; a++)

{

for (b = 1; b <= 9; b++)

{

for (n = 1; n <= 100; n++)

{

if (1000 \* a + 100 \* a + 10 \* b + b == n \* n)

printf("The number is:%d\n", n \* n);

}

}

}

return 0;

}

Q282.(10分)打印所有的水仙花数。

所谓“水仙花数”是指一个三位数，其各位数字立方和等于该数本身。

例如153是“水仙花数”，因为153 = 1\*1\*1+3\*3\*3+5\*5\*5。

\*\*输出格式要求为"%6d"。

参考答案

#include <stdio.h>

int main()

{

int i,j,k,n;

for(n=100;n<1000;n++) //1

{

i=n/100; //1

j=(n-i\*100)/10; //1

k=n-i\*100-j\*10; //1

if(i\*i\*i+j\*j\*j+k\*k\*k==n) //1

printf("%6d",n);

}

printf("\n");

return 0;

}

Q413.(10分)马克思手稿中的数学问题。马克思手稿中有一道趣味数学题：有30个人，其中有男人、女人和小孩，在一家饭馆吃饭共花了50先令：每个男人花3先令，每个女人花2先令，每个小孩花1先令，问男人、女人和小孩各几人？

\*\*输入格式要求：无

\*\*输出格式要求："\tMEN\tWOMEN\tCHILDREN\n"

"-----------------------------------------\n"

"%2d:\t%d\t%d\t%d\n"

程序运行示例如下：

MEN WOMEN CHILDREN

-----------------------------------------

1: 0 20 10

2: 1 18 11

3: 2 16 12

4: 3 14 13

5: 4 12 14

6: 5 10 15

7: 6 8 16

8: 7 6 17

9: 8 4 18

10: 9 2 19

11: 10 0 20

12: 11 -2 21

13: 12 -4 22

14: 13 -6 23

15: 14 -8 24

16: 15 -10 25

17: 16 -12 26

参考答案

#include <stdio.h>

main()

{

int x, y, z, count = 0;

printf("\tMEN\tWOMEN\tCHILDREN\n");

printf("-----------------------------------------\n");

for (x = 0; x <= 16; x++)

{

y = 20 - 2 \* x;

z = 30 - x - y;

if (3 \* x + 2 \* y + z == 50)

printf("%2d:\t%d\t%d\t%d\n", ++count, x, y, z);

}

}

Q319.(10分)从键盘输入一个字符，如果它是大写的英文字母，则将其转换成小写

的英文字母，然后将转换的字符及其ASCII码值显示在屏幕上；如果它

是小写的英文字母，则将其转换成大写的英文字母，然后将转换的字符

及其ASCII码值显示在屏幕上；如果输入的是其它字符则显示输入错误。

\*\*输入提示信息格式: 无

\*\*输入数据格式要求: %c

\*\*输出数据格式要求:

若输入的是大小写英文字母，则输出: "%c,%d"

若输入的是其它字符，则输出: "Input error!"

注：已知大写英文字母A的ASCII码值是65

小写英文字母a的ASCII码值是97

参考答案

#include <stdio.h>

int main()

{

char ch;

scanf("%c", &ch); //1

if (ch >= 'A' && ch <= 'Z') //1

{

ch = ch + 32; //1

printf("%c,%d", ch, ch);

}

else if (ch >= 'a' && ch <= 'z') //1

{

ch = ch - 32; //1

printf("%c,%d", ch, ch);

}

else //1

{

printf("Input error!");

}

return 0;

}

Q408.(10分)四位反序数。设N是一个四位数，它的9倍恰好是其反序数，求N。反序数就是将整数的数字倒过来形成的整数，例如：1234的反序数是4321。

\*\*输入格式要求：无

\*\*输出格式要求："The number satisfied states condition is : %d\n"

参考答案

#include <stdio.h>

main()

{

int i;

for(i=1002;i<1111;i++)

if(i%10\*1000+i/10%10\*100+i/100%10\*10+i/1000==i\*9)

printf("The number satisfied states condition is : %d\n",i);

}

Q209.(10分)已知三角形的三边长为a，b，c，计算三角形面积的公式为：

area =sqrt(s(s-a)(s-b)(s-c)),s=1/2(a+b+c)

其中，a，b，c为浮点数，sqrt为开平方计算。

要求编写程序，从键盘输入a，b，c的值，计算并输出三角形的面积。

注意：不用判断输入的边长值是否满足三角形要求，

只需进行简单计算即可。

\*\*要求输入提示信息格式为："Input a,b,c:\n"；

\*\*输出格式要求："area=%.2f\n"

参考答案

#include <stdio.h>

#include <math.h>

main()

{

float a, b, c;

float s, area;

printf("Input a,b,c:");

scanf("%f,%f,%f",&a,&b,&c); //1

s = 1.0 / 2 \* (a + b + c); //1

area = sqrt(s \* (s - a) \* (s - b) \* (s - c)); //1

printf("area=%.2f\n", area); //1

}

Q491.(10分)求有10个整型元素的数组中最大元素及其下标。

\*\*输入格式要求："%d" 提示信息："请输入10个元素的值：\n"

\*\*输出格式要求："值最大的元素值为：%d 下标为%d\n"

程序运行示例1如下：

请输入10个元素的值：

23 45 36 11 20 90 31 66 30 33

值最大的元素值为：90 下标为5

|  |
| --- |
| #include <stdio.h>    int main()  {      int a[10];      int i, index;//i表示循环变量，index用于存放值最大的元素下标        //输入数组元素      printf("请输入10个元素的值：\n");      for (i = 0; i < 10; i++)      {          scanf("%d", &a[i]);      }      //求值最大的元素及其对应下标      index = 0;      for (i = 1; i < 10; i++)      {          if (a[i] > a[index])              index = i;      }      //输出最大的元素值及其下标      printf("值最大的元素值为：%d  下标为%d\n", a[index], index);        return 0;  } |

Q469.(10分)程序：检查数中重复出现的数字。

用户输入数后，程序显示信息Repeated digit或No Repeated digit：

Enter a number: 28212

Repeated digit

数28212有一个重复的数字（2），而数9357则没有。

\*\*输入格式要求："%ld" 提示信息："Enter a number :"

\*\*输出格式要求："Repeated digit\n\n" "No Repeated digit\n\n"

|  |
| --- |
| #include <stdio.h>    #define TRUE    1  #define FALSE   0    typedef int Bool;    int main()  {      Bool digit\_seen[10] = {0};      int digit;      long int n;        printf("Enter a number :");      scanf("%ld", &n);        while (n > 0)      {          digit = n % 10;          if (digit\_seen[digit])              break;          digit\_seen[digit] = TRUE;          n /= 10;      }        if (n > 0)          printf("Repeated digit\n\n");      else          printf("No Repeated digit\n\n");        return 0;  } |

Q594.(10分)编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #define N 100  int getline(char \*s, int len);  int main()  {      char line[N], b[20], \*ptr1, \*ptrb;      int n = 0, j, a[50];      printf("Please enter a string:");      getline(line, N);      ptr1 = line;      while (\*ptr1 != '\0')      {          ptrb = b;          for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)          {              \*ptrb = \*ptr1;          }          if (j)          {              \*ptrb = '\0';              \*(a + n) = atoi(b);              n++;          }          if (\*ptr1 != '\0')ptr1++;      }      printf("the result of output:\n");      for (j = 0; j < n; j++)      {          printf("%10d\n", \*(a + j));      }      return 0;  }    int getline(char \*s, int len)  {      int c;      char \*p = s;      while (--len > 0 && (c = getchar()) != '\n')      {          \*s++ = c;      }      \*s = '\0';      return s - p;  } |

Q153.(5分)附加题：改错题：输入一行字符，统计其中的英文字符、数字字符、

空格及其它字符的个数。下面程序中存在比较隐蔽的错误，

请通过分析和调试程序，发现并改正程序中的错误。对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

注意：将修改后的完整的源程序写在答题区内。

当且仅当错误全部改正，且程序运行结果调试正确，

才给加5分，如果只改正了部分错误，则不加分。

\*\*输入输出格式要求：输入不要有任何提示信息，直接输入字符串；输出格式见以下代码的输出。

#include <stdio.h>

#define ARR\_SIZE = 80;

main()

{

char str[ARR\_SIZE];

int len, i, letter, digit, space, others;

scanf("%s",&str);

len = strlen(str);

for (i=0; i<len; i++)

{

if ('a' <= str[i] <= 'z' || 'A' <= str[i] <= 'Z');

letter = letter ++;

else if (0 <= str[i] <= 9)

digit = digit ++;

else if (str[i] = ' ' )

space = space ++;

else

others = others ++;

}

printf("English character: %d\n", letter);

printf("digit character: %d\n", digit);

printf("space: %d\n", space);

printf("other character: %d\n", others);

}

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define ARR\_SIZE 80  main()  {      char str[ARR\_SIZE];      int  len, i, letter, digit, space, others;      letter=0;   digit=0;    space=0;    others=0;          gets(str);          len = strlen(str);      for (i=0; i<len; i++)      {        if (str[i] >= 'a' && str[i] <= 'z' || str[i] >= 'A' && str[i] <= 'Z')              letter ++;           else if (str[i] >= '0' && str[i] <= '9')                  digit ++;           else if (str[i] == ' ')                  space ++;           else          others ++;      }      printf("English character:  %d\n", letter);      printf("digit character:  %d\n", digit);      printf("space:  %d\n", space);      printf("other character:  %d\n", others);  } |

Q403.(10分)冒泡排序法。设有N个杂乱无序的整数，要求将这N个整数从小到大排序后输出。

\*\*输入格式要求："%d" 提示信息："Enter No.%2d:"

\*\*输出格式要求："%d"

程序运行示例如下：

Enter No. 1:5

Enter No. 2:7

Enter No. 3:3

Enter No. 4:9

Enter No. 5:8

35789

#include <stdio.h>

#define N 5

main()

{int i,j,t,a[N];

for(i=0;i<N;i++)

{ printf("Enter No.%2d:",i+1);

scanf("%d",&a[i]);

}

for(i=0;i<N-1;i++)

for(j=0;j<N-i-1;j++)

if(a[j]>a[j+1])

{ t=a[j];

a[j]=a[j+1];

a[j+1]=t;

}

for(i=0;i<N;i++)

printf("%d",a[i]);

}

Q1711.(10分)输入一行字符，统计其中的英文字符、数字字符、空格和其他字符的个数。

输入提示信息："Please input a string:"

输入字符串用gets()

输出提示信息和格式：

"English character: %d\n"

"digit character: %d\n"

"space: %d\n"

"other character: %d\n"

#include <stdio.h>

#include <string.h>

#define ARR\_SIZE 80

int main()

{

char str[ARR\_SIZE];

int len, i, letter = 0, digit = 0, space = 0, others = 0;

printf("Please input a string:");

gets(str);

for (i = 0; str[i] != '\0'; i++)

{

if (str[i] >= 'a' && str[i] <= 'z' || str[i] >= 'A' && str[i] <= 'Z')

letter ++; //统计英文字符

else if (str[i] >= '0' && str[i] <= '9' )

digit ++; //统计数字字符

else if (str[i] == ' ' )

space ++; //统计空格

else

others ++; //统计其他字符的个数

}

printf("English character: %d\n", letter);

printf("digit character: %d\n", digit);

printf("space: %d\n", space);

printf("other character: %d\n", others);

return 0;

}

Q365.(10分)如果一个正整数等于其各个数字的立方和，则该数称为阿姆斯特朗数（亦称为自恋性数）。如407=4^3+0^3+7^3就是一个阿姆斯特朗数。试编程求1000以内的所有阿姆斯特朗数。

\*\*输出格式要求："There are following Armstrong number smaller than 1000:\n" " %d "

程序运行示例如下：

There are following Armstrong number smaller than 1000:

153 370 371 407

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,t,k,a[4];      printf("There are following Armstrong number smaller than 1000:\n");      for(i=2;i<1000;i++)      {          for(t=0,k=1000;k>=10;t++)          {              a[t]=(i%k)/(k/10);              k/=10;          }          if(a[0]\*a[0]\*a[0]+a[1]\*a[1]\*a[1]+a[2]\*a[2]\*a[2]==i)              printf(" %d ",i);      }  } |

Q1281.(10分)

|  |
| --- |
| 程序改错。下面程序的功能是从键盘输入一行字符，统计其中有多少单词。假设单词之间以空格分开。[提示：判断是否有新单词出现的方法是——当前被检验的字符不是空格，而前一被检验字符是空格，则表示有新单词出现。]  #include<stdio.h>  #define ARR\_SIZE 80;  main()  {      char str[ARR\_SIZE];      int i, num;        gets(str);      if(str[0] != ' ' && str[0] != '\0')      {          num = 1;      }      else      {          num = 0;      }      for(i = 1 ; str[i] != '\0'; i++)      {          if(str[i]  != ' ' && str[i-1]  = ' ')          {              num++;          }      }      printf("num=%d\n", num);  } |

#include<stdio.h>

#define ARR\_SIZE 80

main()

{

    char str[ARR\_SIZE];

    int i, num;

    gets(str);

    if(str[0] != ' ' && str[0] != '\0')

    {

        num = 1;

    }

    else

    {

        num = 0;

    }

    for(i = 1 ; str[i] != '\0'; i++)

    {

        if(str[i]  != ' ' && str[i-1]  == ' ')

        {

            num++;

        }

    }

    printf("num=%d\n", num);

}

Q430.(10分)乘式还原。有乘法算式如下：

×oooooooooooooooooo

18个O的位置上全部是素数（2、3、5或7），请还原此算式。

程序运行示例如下：

775

\* 33

-----

2325

2325

-----

25575

|  |
| --- |
| #include <stdio.h>  #define NUM 5  #define C\_NUM 4  int a[NUM+1];  int b[]={0,2,3,5,7};  main()  {      int i,not\_finish=1;      i=2;      a[1]=1;      while(not\_finish)      {          while(not\_finish&&i<=NUM)              if(a[i]>=C\_NUM)                  if(i==1&&a[i]==C\_NUM)                      not\_finish=0;                  else a[i--]=0;              else a[i++]++;          if(not\_finish)          {              long int sum1,sum2,sum3,sum4;              sum1=b[a[1]]\*100+b[a[2]]\*10+b[a[3]];              sum2=sum1\*b[a[5]];              sum3=sum1\*b[a[4]];              if(sum2>=2222&&sum2<=7777&&f(sum2)&&sum3>=2222&&sum3<=7777&&f(sum3))                  if((sum4=sum2+sum3\*10)>=22222&&sum4<=77777&&f(sum4))                  {                      printf("  %d\n",sum1);                      printf("\*  %d%d\n",b[a[4]],b[a[5]]);                      printf("-----\n");                      printf(" %d\n",sum2);                      printf("%d\n",sum3);                      printf("-----\n");                      printf("%d\n",sum4);                  }               i=NUM;          }      }  }    f(sum)  long sum;  {   int i,k,flag;      while(sum>0)      {          i=sum%10;          for(flag=0,k=1;!flag&&k<=C\_NUM;k++)          if(b[k]==i)          {              flag=1;              break;          }          if(!flag)   return(0);          else      sum=sum/10;      }      return (1);  } |

Q134.(5分)从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str)==0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

|  |
| --- |
| #include "stdio.h"  #include "string.h"  main()  {      int i=0, findFlag=1;      char x[13];      char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };        printf("Input string:\n");      gets(x);        while( i<6 && findFlag)      {          if(strcmp(x,str[i])==0)          {              findFlag = 0;          }          i++;      }      if ( !findFlag )      {          printf("%s\n",x);      }      else      {          printf("Not find!\n");      }  } |

Q457.(10分)从键盘输入一个长度为N（比如10）的整型数组，而后将数组中小于零的元素移动到数组的前端，大于零的元素移动到数组的后端，等于零的元素留在数组中间。比如原始数组为：2 -5 -89 75 0 -89 0 93 48 0，经过处理后的数组为：-5 -89 -89 0 0 0 75 93 48 2。由于不要求数组有序，所以不允许用排序方法。

\*\*输入格式要求："%d" 提示信息："Please input %d interger number\n"

\*\*输出格式要求："%5d"

程序运行示例如下：

Please input 10 interger number

2 -5 -89 75 0 -89 0 93 48 0

-5 -89 -89 0 0 0 75 93 48 2

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define N 10    int main()  {      int a[N];      int i, p1 = 0, p2 = 9, p, temp;      printf("Please input %d interger number\n", N);      for (i = 0; i <= N - 1; i++)      {          scanf("%d", &a[i]);      }      for (; p1 < p2 ; )      {          if (a[p1] == 0)          {              p = p1;              while (a[p1 + 1] == 0)              {                  p1++;              }              if (p1 + 1 < p2)              {                  temp = a[p1 + 1];                  a[p1 + 1] = a[p];                  a[p] = temp;                  p1 = p;                  continue;              }              else              {                  break;              }          }          if (a[p1] > 0)          {              temp = a[p1];              a[p1] = a[p2];              a[p2] = temp;              p2--;          }          else          {              p1++;          }          if (a[p2] == 0)          {              p = p2;              while (a[p2 - 1] == 0)              {                  p2--;              }              if (p1 < p2 - 1)              {                  temp = a[p2 - 1];                  a[p2 - 1] = a[p];                  a[p] = temp;                  p2 = p;                  continue;              }              else              {                  break;              }          }          if (a[p2] < 0)          {              temp = a[p2];              a[p2] = a[p1];              a[p1] = temp;              p1++;          }          else          {              p2--;          }      }      for (i = 0; i <= N - 1; i++)      {          printf("%5d", a[i]);      }      printf("\n");  } |

Q446.(10分)计算分数的精确值

使用数组精确计算M/N（0<M<N<=100）的值。如果M/N是无限循环小数，则计算并输出它的第一循环节，同时要求输出循环节的起止位置（小数位的序号）。

\*\*输入格式要求："%d/%d" 提示信息："please input a fraction (m/n)) (0< m< n<=100):"

\*\*输出格式要求："%d/%d it's accuracy value is:0." "\n\tand it is a infinite cyclic fraction from %d\n" "\tdigit to %d digit after decimal point.\n"

程序运行示例如下：

please input a fraction (m/n)) (0< m< n<=100):1/6

1/6 it's accuracy value is:0.16

and it is a infinite cyclic fraction from 2

digit to 2 digit after decimal point.

|  |
| --- |
| #include <stdio.h>  int remainder[101],quotient[101];  main()  {      int m,n,i,j;      printf("please input a fraction (m/n))  (0< m< n<=100):");      scanf("%d/%d",&m,&n);      printf("%d/%d it's accuracy value is:0.",m,n);      for (i=1;i<=100;i++)      {          remainder[m]=i;          m\*=10;          quotient[i]=m/n;          m=m%n;          if(m==0)          {              for(j=1;j<=i;j++)                  printf("%d",quotient[j]);              break;          }          if(remainder[m]!=0)          {              for (j=1;j<=i;j++)                  printf("%d",quotient[j]);              printf("\n\tand it is a infinite cyclic fraction from %d\n",remainder[m]);              printf("\tdigit to %d digit after decimal point.\n",i);              break;          }      }  } |

Q468.(10分)程序：数列反向。

要求用户录入一串数（假设为10个整数），然后按反向顺序输出这些数：

Enter 10 numbers: 34 82 49 102 7 94 23 11 50 31

In reverse order: 31 50 11 23 94 7 102 49 82 34

\*\*输入格式要求："%d" 提示信息："Enter %d numbers:"

\*\*输出格式要求："In reverse order:" " %d"

#include <stdio.h>

#define N 10

int main()

{

    int a[N], i;

    printf("Enter %d numbers:", N);

    for (i = 0; i < N; i++)

        scanf("%d", &a[i]);

    printf("In reverse order:");

    for (i = N - 1; i >= 0; i--)

        printf(" %d", a[i]);

    printf("\n");

    return 0;

}

Q577.(10分)编程实现两个数组中对应该元素值的交换（数组的长度定义成5）。

输出函数："%5d"

输入举例:注意输入的顺序

6 7 8 9 10 （输入第一个数组元素）

1 2 3 4 5 （输入第二个数组元素）

输出举例：

1 2 3 4 5 （输入第一个数组元素）

6 7 8 9 10 （输入第二个数组元素）

|  |
| --- |
| #include <stdio.h>  int main()  {      int a[5], b[5];      int i, t;      for (i = 0; i < 5; i++)          scanf("%d", &a[i]);      for (i = 0; i < 5; i++)          scanf("%d", &b[i]);      for (i = 0; i < 5; i++)      {          t = a[i];          a[i] = b[i];          b[i] = t;      }      for (i = 0; i < 5; i++)          printf("%5d", a[i]);      printf("\n");      for (i = 0; i < 5; i++)          printf("%5d", b[i]);      printf("\n");  } |

Q561.(10分)某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下：每位数字都加上 5, 然后用和除以 10 的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

\*\*输入格式要求："%d" 提示信息："Input a 4 digits number\n"

\*\*输出格式要求："%d"

程序运行示例如下：

Input a 4 digits number

1234

9876

### 参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      int a, i, aa[4], t;        printf("Input a 4 digits number\n");      scanf("%d", &a);        aa[0] = a % 10;      aa[1] = a % 100 / 10;      aa[2] = a % 1000 / 100;      aa[3] = a / 1000;        for (i = 0; i <= 3; i++)      {          aa[i] += 5;          aa[i] %= 10;      }        for (i = 0; i <= 3 / 2; i++)      {          t = aa[i];          aa[i] = aa[3 - i];          aa[3 - i] = t;      }        for (i = 3; i >= 0; i--)          printf("%d", aa[i]);      printf("\n");        return 0;  } |

Q1715.(10分)假设今年的工业产值为100万元，产值增长率为每年c%，请编程计算当c分别为6、8、10、12时工业产值分别过多少年可实现翻一番（即增加一倍）。

输入格式：无

输出格式："When grow rate is %d%%, the output can be doubled after %d years.\n"

|  |
| --- |
| #include <stdio.h>  #define  CURRENT  100  #define  N      4  main()  {      int  i;      int  year;                              /\* 产值翻番所需年数 \*/      int   growRate[N] = {6, 8, 10, 12};     /\* 工业产值的增长率 \*/      double  output;                         /\* 工业总产值 \*/      for (i = 0; i < N; i++)      {          output = CURRENT;                /\* 当年产值为100万元 \*/          for (year = 0; output < 2 \* CURRENT; year++)          {              output = output \* (1 + growRate[i] / 100.0);          }          printf("When grow rate is %d%%, the output can be doubled after                 %d years.\n", growRate[i], year);      }  } |

Q492.(10分)输入10个学生的C语言成绩（假设成绩为整数值），求最高成绩、最低成绩和平均值。

\*\*输入格式要求："%d" 提示信息："请输入10个学生的C成绩：\n"

\*\*输出格式要求："%d " "10个学生的C成绩分别为：\n" "最高成绩是%d,是第%d个学生\n" "最低成绩是%d,是第%d个学生\n" "10名学生的C成绩的平均分为:%.2f\n"

程序运行示例如下：

请输入10个学生的C成绩：

80 90 93 78 83 67 79 60 64 45

10个学生的C成绩分别为：

80 90 93 78 83 67 79 60 64 45

最高成绩是93,是第3个学生

最低成绩是45,是第10个学生

10名学生的C成绩的平均分为:73.90

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int cScore[10], cMax, cMin, cMaxIndex, cMinIndex, cSum;      //cMax,cMine,cSum分别存放10个学生中最高成绩及最低成绩、成绩总和      int i;        /\*输入10个学生的成绩\*/      printf("请输入10个学生的C成绩：\n");      for (i = 0; i < 10; i++)          scanf("%d", &cScore[i]);      /\*计算平均值、最高分数和最低分数,假设第一个学生的成绩是最高和最低\*/      cMax = cMin = cScore[0];      cSum = cScore[0];      cMaxIndex = cMinIndex = 0;        for (i = 1; i < 10; i++)      {          cSum = cSum + cScore[i];          if (cScore[i] > cMax)          {              cMax = cScore[i];              cMaxIndex = i;          }          if (cScore[i] < cMin)          {              cMin = cScore[i];              cMinIndex = i;          }      }      /\*打印结果\*/      printf("10个学生的C成绩分别为：\n");      for (i = 0; i < 10; i++)          printf("%d ", cScore[i]);      printf("\n");      printf("最高成绩是%d,是第%d个学生\n", cMax, cMaxIndex + 1);      printf("最低成绩是%d,是第%d个学生\n", cMin, cMinIndex + 1);      printf("10名学生的C成绩的平均分为:%.2f\n", (1.0 \* cSum) / 10);        return 0;  } |

Q447.(10分)求1898

现將不超过2000的所有素数从到大排成第一行，第二行上的每个数都等于它“右肩”上的素数与“左肩”上的素数之差。请编程求出：第二行数中是否存在这样的若干个连续的整数，它们的和恰好是1898？假如存在的话，又有几种这样的情况？

\*\*输入格式要求：提示信息："There are follwing primes in first row :\n"

\*\*输出格式要求："(%d).%3d,......,%d\n"

程序运行示例如下：

There are follwing primes in first row :

(1).101,......,1999

(2). 89,......,1987

(3). 53,......,1951

(4). 3,......,1901

|  |
| --- |
| #include <stdio.h>  #include <math.h>  #define NUM 320  int number[NUM];  main()  {      int i,j,count=0;      printf("There are follwing primes in first row :\n");      for(j=0,i=3;i<=1999;i+=2)          if (fflag(i)) number[j++]=i;      for(j--;number[j]>1898;j--)      {          for(i=0;number[j]-number[i]>1898;i++);          if(number[j]-number[i]==1898)              printf ("(%d).%3d,......,%d\n",++count,number[i],number[j]);      }  }  fflag(i)  int i;  {      int j;      if (i<=1) return(0);      if (i==2)  return(1);      if(!(i%2)) return(0);      for(j=3;j<=(int)( sqrt((double)i)+1);j+=2)          if (!(i%j))  return(0);      return(1);  } |

Q424.(10分)魔术师的猜牌术。魔术师利用一副牌中的十三张黑桃，预先将它们排好后迭在一起，牌面朝下。对观众说：我不看牌，只数数就可以猜到每张牌是什么，我大声数数，你们听，不信？你们看看。魔术师将最上面的那张牌数为1，把它翻过来正好是黑桃A，将黑桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌下面，将第二张牌翻过来，正好是黑桃2，也将它放在桌子上。第三次数1、2、3，将前两张牌依次放在这迭牌下面，将第三张牌翻过来，正好是黑桃3。这样依次进行将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

\*\*输入格式要求：提示信息："The original order of cards is:"

\*\*输出格式要求："%d "

程序运行示例如下：

The original order of cards is:1 8 2 5 10 3 12 11 9 4 7 6 13

|  |
| --- |
| #include <stdio.h>  int a[14];  main()  {int i,n,j=1;   printf("The original order of cards is:");   for(i=1;i<=13;i++)   {n=1;    do    {if(j>13)       j=1;     if(a[j])       j++;     else     {if(n==i)         a[j]=i;         j++;         n++;     }    }while(n<=i);   }   for(i=1;i<=13;i++)       printf("%d ",a[i]);   printf("\n");   } |

Q382.(10分)\*约瑟夫问题。这是十七世纪的法国数学家加斯帕在《数目的游戏问题》中讲的一个故事：15个基督教徒和15个异教徒在海上遇险，必须将一半的人投入海中，其余的人才能幸免于难，于是想了个办法：30个人围成一个圆圈，从第一个人开始依次报数，每数到第9个人就将他扔入大海，如此循环进行直到仅余15个人为止。问怎样排法，才能使每次投入大海的都是异教徒。

\*\*输出格式要求："The original circle is (+:papandom, @:christian);\n" "%c "

程序运行示例如下：

The original circle is (+:papandom, @:christian);

@ @ @ @ + + + + + @ @ + @ @ @ + @ + + @ @ + + + @ + + @ @ +

|  |
| --- |
| #include <stdio.h>  struct node  {    int nextp;    int no\_out;  }link[31];  main()  {    int i,j,k;    printf("The original circle is (+:papandom,  @:christian);\n");    for(i=1;i<=30;i++)    {      link[i].nextp=i+1;      link[i].no\_out=1;    }    link[30].nextp=1;    j=30;    for(i=0;i<15;i++)    {       for(k=0;;)       if(k<9)       {         j=link[j].nextp;         k+=link[j].no\_out;       }       else break;       link[j].no\_out=0;    }    for(i=1;i<=30;i++)          printf("%c ",link[i].no\_out?'@':'+');      printf("\n");    } |

Q1714.(10分)编程实现从键盘任意输入20个整数，统计非负数个数，并计算非负数之和。

输入提示信息："Input 20 Numbers:\n"

输入格式："%d"

输出格式："sum = %d, counter = %d\n"

运行示例：

Input 20 Numbers:

1 2 3 4 5 6 7 8 -9 -10↙

12 -45 56 -99 34 87 90 -23 0 65↙

sum = 380, counter = 15

|  |
| --- |
| #include  <stdio.h>  int main(void)  {      int i, n, sum = 0, counter = 0;      printf("Input 20 Numbers:\n");      for (i = 0; i < 20; i++)      {          scanf("%d", &n);          if (n >= 0)              //判断是否为非负数          {              sum += n;           //非负数求和              counter++;          //非负数个数计算          }      }      printf("sum = %d, counter = %d\n", sum, counter);      return 0;  } |

Q1332.(10分)创建并输出一个一维数组（含20个元素），数组元素的值分别是下标的3倍多2。

\*\*输出格式要求："%3d"

程序的运行示例如下：

2 5 8 11 14 17 20 23 26 29 32 35 38 41 44 47 50 53 56 59

|  |
| --- |
| #include <stdio.h>    main()  {      int a[20], k;      for (k = 0; k < 20; k++)      {          a[k] = 3 \* k + 2;          printf("%3d", a[k]);      }  } |

Q558.(10分)从一个字符串删除元音字母，例如，如果输入的字符串是“C programming”，那么输出将是“C prgrmmng”。

\*\*输入格式要求：提示信息："Enter a string to delete vowels\n"

\*\*输出格式要求："String after deleting vowels: %s\n"

程序运行示例如下：

Enter a string to delete vowels

c programming

String after deleting vowels: c prgrmmng

|  |
| --- |
| #include <stdio.h>  #include <string.h>    int check\_vowel(char);    int main()  {      char s[100], t[100];      int i, j = 0;        printf("Enter a string to delete vowels\n");      gets(s);        for (i = 0; s[i] != '\0'; i++)      {          if (check\_vowel(s[i]) == 0)        //not a vowel          {              t[j] = s[i];              j++;          }      }        t[j] = '\0';        strcpy(s, t);    //We are changing initial string        printf("String after deleting vowels: %s\n", s);        return 0;  }      int check\_vowel(char c)  {      switch (c)      {      case 'a':      case 'A':      case 'e':      case 'E':      case 'i':      case 'I':      case 'o':      case 'O':      case 'u':      case 'U':          return 1;      default:          return 0;      }  } |

Q363.(10分)给定一维数组a，求a中各元素的平均值。

a[10]={1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9}

\*\*输出格式要求："The average of array is %f\n"

程序运行示例如下：

The average of array is 1.450000

|  |
| --- |
| main()  {      float average();      static float a[10]={1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9};      printf("The average of array is %f\n",average(a,10));  }  float average(a,n)      int n;      float a[];  {      int i;      float sum=0;      for(i=0;i<n;i++)          sum+=a[i];      return sum/n;  } |

Q599.(10分)创建并输出一个一维数组（含20个元素），数组元素的值分别是下标的3倍多2。

\*\*输出格式要求：" %5d"

程序运行示例如下：

2 5 8 11 14 17 20 23 26 29 32 35 38 41 44 47 50 53 56 59

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int a[20], k;      for (k = 0; k < 20; k++)      {          a[k] = 3 \* k + 2;          printf(" %5d", a[k]);      }  } |

Q1383.(10分)以下程序的功能是从键盘输入10个整型数据，放入数组a 中，求其最大值、最小值及其所在元素的下标位置并输出。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  main()  {      int a[10], n, max, min, maxPos, minPos;        for (n=0, n<10, n++)      {          scanf("%d", a[n]);      }      max = min = a;      maxPos = minPos = 0;      for (n=1; n<10; n++)      {          if (a[n] > max)          {                  max = a[n];                  maxPos = n;          }          else if (a[n] < min)          {                  min = a[n];                  minPos = n;          }      }      printf("max=%d, pos=%d\n",max, maxPos);      printf("min=%d, pos=%d\n",min, minPos);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {   int a[10], n, max, min, maxPos, minPos;      for (n=0; n<10; n++)      {          scanf("%d", &a[n]);      }      max = min = a[0];   //应初始化为数组中的某一元素      maxPos = minPos = 0;      for (n=1; n<10; n++)      {          if (a[n] > max)          {                  max = a[n];                  maxPos = n;          }          else if (a[n] < min)          {                  min = a[n];                  minPos = n;          }      }        printf("max=%d, pos=%d\n",max, maxPos);      printf("min=%d, pos=%d\n",min, minPos);  } |

Q1123.(10分)对候选人得票的统计程度。设由3个候选人，每次输入一个得票的候选人的名字，要求最后输入各人得票结果。（总票数为n）请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

struct person

{

char name[20];

int count;

}

person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

int i, j;

char leader\_name[20];

scanf("%d\n", &n);

for (i = 0; i < n; i++)

{

scanf("%s", leader\_name);

for (j = 0; j < 3; j++)

if (leader\_name == leader[j].name)

leader[j].count++;

}

printf("\n");

for (i = 0; i < 3; i++)

printf("%5s:%d\n", leader[i].name, leader[i].count);

}

#include <string.h>

struct person

{

    char name[20];

    int count;

};

struct person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

    int i, j, n;

    char leader\_name[20];

    scanf("%d\n", &n);

    for (i = 0; i < n; i++)

    {

        scanf("%s", leader\_name);

        for (j = 0; j < 3; j++)

            if (!strcmp(leader\_name, leader[j].name))

                leader[j].count++;

    }

    printf("\n");

    for (i = 0; i < 3; i++)

        printf("%5s:%d\n", leader[i].name, leader[i].count);

}

Q615.(10分)有n个人围成一圈，顺序排号。从第一个人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

\*\*输入格式要求："%d" 提示信息："please input the total of numbers:"

\*\*输出格式要求："%d is left\n"

程序运行示例如下：

please input the total of numbers:30

2 is left

|  |
| --- |
| #include <stdio.h>  #define nmax 50  int main()  {      int i, k, m, n, num[nmax], \*p;      printf("please input the total of numbers:");      scanf("%d", &n);      p = num;      for (i = 0; i < n; i++ )          \*(p + i) = i + 1;      i = 0;      k = 0;      m = 0;      while (m < n)      {          if (\*(p + i) != 0)              k++;          if (k % 3 == 0)          {              \*(p + i) = 0;              m++;          }          i++;          if (i == n)              i = 0;      }      while (\*p == 0)          p++;      printf("%d is left\n", \*p);  } |

Q450.(10分)数字排列 在较中的九个顶点上，空出中间的点，其余的点上任意填入数字1至8；1的位置固定不动，然后移动其余数字，使1到8顺时针从小到大排列。移动的规则是：只能将数字沿线移向空白的点。请编程显示数字移动过程。

○—○—○

︱＼︱／︱

○—◇—○

︱／︱＼︱

○—○—○

\*\*输入格式要求："%d" 提示信息："Please enter original order of digits 1~8:"

\*\*输出格式要求："%2d\n" "The sorting process is as felow:\n" "---%2d---\n"

程序运行示例如下：

Please enter original order of digits 1~8:1 2 3 4 5 6 7 8

The sorting process is as felow:

1 2 3

8 0 4

7 6 5

--- 0---

int a[9]={0,1,2,5,8,7,6,3};

int b[9];

int c[9];

int count=0;

main()

{int i,j,k,t;

void print();

printf("Please enter original order of digits 1~8:");

for (i=0;i<8;i++)

    scanf("%d",&b[a[i]]);

printf("The sorting process is as felow:\n");

print();

for(t=-1,j=0;j<8 && t==-1;j++)

   if(b[a[j]==1]) t=j;

for(j=0;j<8;j++)

    c[j]=a[(j+t)%8];

for(i=2;i<9;i++)

    for(j=i-1;j<8;j++)

    if(b[c[j]]==i&&j!=i-1)

    { b[4]=i;

      b[c[j]]=0;

      print();

      for (k=j;k!=i-1;k--)

      {

          b[c[k]]=b[c[k-1]];

          b[c[k-1]]=0;

          print();

      }

      b[c[k]]=i;

      b[4]   =0;

      print();

      break;

    }

    else if(b[c[j]==i])

         break;

}

void print(void)

{ int c;

  for(c=0;c<9;c++)

    if(c%3==2)

        printf("%2d\n",b[c]);

    else

        printf("%2d",b[c]);

  printf("---%2d---\n",count++);

}

Q428.(10分)八皇后问题。在一个8\*8的国际象棋棋盘上，有八个皇后，每个皇后占一格；要求皇后间不会出现相互“攻击”的现象，即不能有两个皇后处在同一行、同一列或同一对角线上。问共有多少种不同的方法。

\*\*输入格式要求：提示信息："The possible configuration of 8 queens are:\n"

\*\*输出格式要求：" [%2d]:" " %d"

程序运行示例如下：

The possible configuration of 8 queens are:

[ 1]: 1 5 8 6 3 7 2 4 [ 2]: 1 6 8 3 7 4 2 5 [ 3]: 1 7 4 6 8 2 5 3

[ 4]: 1 7 5 8 2 4 6 3 [ 5]: 2 4 6 8 3 1 7 5 [ 6]: 2 5 7 1 3 8 6 4

[ 7]: 2 5 7 4 1 8 6 3 [ 8]: 2 6 8 3 1 4 7 5 [ 9]: 2 6 1 7 4 8 3 5

[10]: 2 7 3 6 8 5 1 4 [11]: 2 7 5 8 1 4 6 3 [12]: 2 8 6 1 3 5 7 4

[13]: 3 5 7 1 4 2 8 6 [14]: 3 5 8 4 1 7 2 6 [15]: 3 5 2 8 1 7 4 6

[16]: 3 5 2 8 6 4 7 1 [17]: 3 6 8 1 4 7 5 2 [18]: 3 6 8 1 5 7 2 4

[19]: 3 6 8 2 4 1 7 5 [20]: 3 6 2 5 8 1 7 4 [21]: 3 6 2 7 1 4 8 5

[22]: 3 6 2 7 5 1 8 4 [23]: 3 6 4 1 8 5 7 2 [24]: 3 6 4 2 8 5 7 1

[25]: 3 7 2 8 5 1 4 6 [26]: 3 7 2 8 6 4 1 5 [27]: 3 8 4 7 1 6 2 5

[28]: 3 1 7 5 8 2 4 6 [29]: 4 6 8 2 7 1 3 5 [30]: 4 6 8 3 1 7 5 2

[31]: 4 6 1 5 2 8 3 7 [32]: 4 7 1 8 5 2 6 3 [33]: 4 7 3 8 2 5 1 6

[34]: 4 7 5 2 6 1 3 8 [35]: 4 7 5 3 1 6 8 2 [36]: 4 8 1 3 6 2 7 5

[37]: 4 8 1 5 7 2 6 3 [38]: 4 8 5 3 1 7 2 6 [39]: 4 1 5 8 2 7 3 6

[40]: 4 1 5 8 6 3 7 2 [41]: 4 2 5 8 6 1 3 7 [42]: 4 2 7 3 6 8 1 5

[43]: 4 2 7 3 6 8 5 1 [44]: 4 2 7 5 1 8 6 3 [45]: 4 2 8 5 7 1 3 6

[46]: 4 2 8 6 1 3 5 7 [47]: 5 7 1 3 8 6 4 2 [48]: 5 7 1 4 2 8 6 3

[49]: 5 7 2 4 8 1 3 6 [50]: 5 7 2 6 3 1 4 8 [51]: 5 7 2 6 3 1 8 4

[52]: 5 7 4 1 3 8 6 2 [53]: 5 8 4 1 3 6 2 7 [54]: 5 8 4 1 7 2 6 3

[55]: 5 1 4 6 8 2 7 3 [56]: 5 1 8 4 2 7 3 6 [57]: 5 1 8 6 3 7 2 4

[58]: 5 2 4 6 8 3 1 7 [59]: 5 2 4 7 3 8 6 1 [60]: 5 2 6 1 7 4 8 3

[61]: 5 2 8 1 4 7 3 6 [62]: 5 3 8 4 7 1 6 2 [63]: 5 3 1 6 8 2 4 7

[64]: 5 3 1 7 2 8 6 4 [65]: 6 8 2 4 1 7 5 3 [66]: 6 1 5 2 8 3 7 4

[67]: 6 2 7 1 3 5 8 4 [68]: 6 2 7 1 4 8 5 3 [69]: 6 3 5 7 1 4 2 8

[70]: 6 3 5 8 1 4 2 7 [71]: 6 3 7 2 4 8 1 5 [72]: 6 3 7 2 8 5 1 4

[73]: 6 3 7 4 1 8 2 5 [74]: 6 3 1 7 5 8 2 4 [75]: 6 3 1 8 4 2 7 5

[76]: 6 3 1 8 5 2 4 7 [77]: 6 4 7 1 3 5 2 8 [78]: 6 4 7 1 8 2 5 3

[79]: 6 4 1 5 8 2 7 3 [80]: 6 4 2 8 5 7 1 3 [81]: 7 1 3 8 6 4 2 5

[82]: 7 2 4 1 8 5 3 6 [83]: 7 2 6 3 1 4 8 5 [84]: 7 3 8 2 5 1 6 4

[85]: 7 3 1 6 8 5 2 4 [86]: 7 4 2 5 8 1 3 6 [87]: 7 4 2 8 6 1 3 5

[88]: 7 5 3 1 6 8 2 4 [89]: 8 2 4 1 7 5 3 6 [90]: 8 2 5 3 1 7 4 6

[91]: 8 3 1 6 2 5 7 4 [92]: 8 4 1 3 6 2 7 5

|  |
| --- |
| #include<stdio.h>  #define NUM 8  int a[NUM+1];  main()  {  int number,i,k,flag,not\_finish=1,count=0;  i=1;  a[1]=1;  printf("The possible configuration of 8 queens are:\n");  while(not\_finish)  {      while(not\_finish&&i<=NUM)      {      for(flag=1,k=1;flag&&k<i;k++)          if(a[k]==a[i])  flag=0;      for(k=1;flag&&k<i;k++)          if((a[i]==a[k]-(k-i))||(a[i]==a[k]+(k-i)))              flag=0;      if(!flag)      {          if(a[i]==a[i-1])          {              i--;              if(i>1&&a[i]==NUM)                  a[i]=1;              else if(i==1&&a[i]==NUM)                  not\_finish=0;              else a[i]++;         }        else          if(a[i]==NUM)a[i]=1;          else         a[i]++;      }      else if(++i<=NUM)      {    if(a[i-1]==NUM)              a[i]=1;          else              a[i]=a[i-1]+1;      }      }      if(not\_finish)      {          ++count;          printf((count-1)%3?"  [%2d]:" : "\n   [%2d]:",count);          for(k=1;k<=NUM;k++)              printf(" %d",a[k]);          if(a[NUM-1]<NUM)              a[NUM-1]++;          else a[NUM-1]=1;          i=NUM-1;      }  }  } |

Q1392.(10分)程序改错。

以下程序的功能是统计字符数。判断一个由’0’ ~ ‘9’这10个字符组成的字符串中哪个字符出现的次数最多。

输入数据：第一行是测试数据的组数m，每组测试数据占1行，每行数据不超过1000个字符且非空。

输出要求：m行，每行对应一组输入，包括出现次数最多的字符和该字符出现的次数。如果有多个字符出现的次数相同且最多，那么输出ASCII码最小的那一个。

|  |
| --- |
| #include <stdio.h>  #include <string.h>  main( )  {  int  cases, sum[10], i, max;    char str[1000];      scanf("%d", case);    while (cases > 0)         {                  scanf("%c", str);              for( i = 0; i < 10; i++)                   sum[i] = 0;                      for(i < 0; i < strlen(str); i++)                  ++sum[str[i] – 0];              max = 0;              for (i = 1; i < 10; i++)                  if(sum[i] >＝ sum[max]) max = i;              printf("%c %d\n", max + '0', sum[0]);              cases --;         }  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  main( )  {      int  cases, sum[10], i, max;      char str[1001];      scanf("%d", &cases);      while (cases > 0)      {          scanf("%s", str);          for( i = 0; i < 10; i++)              sum[i] = 0;          for(i = 0; i < strlen(str); i++)              ++sum[str[i] - '0'];          max = 0;          for (i = 1; i < 10; i++)               if(sum[i] > sum[max]) max = i;          printf("%c %d\n", max + '0', sum[max]);          cases --;      }  } |

Q458.(20分)设数组a的定义如下：int a[20];

已存入数组中的数据值已经按由大到小的顺序存放，数组元素个数n和数组元素值通过键盘输入，现再从键盘输入一个数据x，把它插入到数组中，要求插入新数据以后，数组数据仍然保持有序。请编写一个程序实现上述功能。

\*\*输入格式要求："%d", 提示信息："输入数组中已有元素个数:" "输入数组中的元素:\n" "输入要插入的数据：" "输出插入数据后的所有数组元素:\n"

\*\*输出格式要求："%d "

程序运行示例如下：

输入数组中已有元素个数:5

输入数组中的元素:

8 7 5 3 1

输入要插入的数据：6

输出插入数据后的所有数组元素:

8 7 6 5 3 1

|  |
| --- |
| #include <stdio.h>    int main()  {      int a[20];      int i, j, n,x;      printf("输入数组中已有元素个数:");      scanf("%d", &n);      printf("输入数组中的元素:\n");      for (i=0;i<n;i++)          scanf("%d",&a[i]);      printf("输入要插入的数据：");      scanf("%d",&x);      for (i = 0; i < n; i++)          if (a[i] < x)          {              for (j = n; j > i; j--)                  a[j] = a[j - 1];              a[i] = x;              break;          }      if (i == n)          a[i] = x;      printf("输出插入数据后的所有数组元素:\n");      for (i = 0; i < n+1; i++)          printf("%d ", a[i]);      printf("\n");  } |

Q557.(20分)设有a[n],b[n]。求a[0]\*b[0]+a[1]\*b[1]+...+a[n-1]\*b[n-1] 。其中n，及数组a和b的元素值由键盘输入。

程序运行示例如下：

Input n

3

Input array a

1 2 3

Input array b

4 5 6

sum=32.000000

#include <stdio.h>

float product(float\*pa, float \*pb, int n)

{

    int i;

    float s;

    for (i = 0, s = 0; i < n; i++)

    {

        s = s + (\*pa \* \*pb);

        pa++;

        pb++;

    }

    return s;

}

int main(int argc, char\* argv[])

{

    float \*pa, \*pb, sum;

    int i, n;

    printf("Input n\n");

    scanf("%d", &n);

    pa = (float\*) malloc(n \* sizeof(float));

    pb = (float\*) malloc(n \* sizeof(float));

    printf("Input array a\n");

    for (i = 0; i < n; i++)

    {

        scanf("%f", pa);

        pa++;

    }

    printf("Input array b\n");

    for (i = 0; i < n; i++)

    {

        scanf("%f", pb);

        pb++;

    }

    pa = pa - n;

    pb = pb - n;

    sum = product(pa, pb, n);

    printf("\nsum=%f\n", sum);

    return 0;

}

AddWXY

Q1726.(10分)简单的口令检查程序

按下述要求编写口令检查程序（假设正确的口令为8888）。

（1）若输入口令正确，则提示“You are welcome!”，程序结束。

（2）若输入口令不正确，则提示“Wrong password!”，同时检查口令是否已输入3次，若未输入3次，则提示“Enter again：”，且允许用户再次输入口令；若已输入3次，则提示“You have entered three times! You are not welcome!”，且不允许用户再输入口令，程序结束。

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define MAXLEN 20  int main(void)  {  int n, flag = 0;  char str[MAXLEN];  printf("Please enter password:");  gets(str);  n = 1;  while ((n <= 3) && (flag == 0))  {  if (strcmp(str, "8888") == 0) //实际口令为8888  {  printf("You are welcome!\n");  flag = 1;  }  else  {  printf("Wrong password!\n");  if (n < 3)  {  printf("Please enter again:");  gets(str);  n++;  }  else  {  printf("You have entered three times!\n");  printf("You are not welcome!\n");  flag = 1;  }  }  }  return 0;  } |

Q1129.(10分)将字符串“Hello World”读入字符数组中去。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

#include <string.h>

main()

{

char ch[10];

scanf("%s", ch);

if (!strcmp(ch, "Hello World"))

printf("OK!");

else

printf("Error!");

}

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  main()  {  char ch[10];  gets(ch);  if (!strcmp(ch, "Hello World"))  printf("OK!");  else  printf("Error!");  } |

Q277.(20分)编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 80  void main()  {  int n;  char str[N], min[N];  printf("Input five countries' names:\n");  gets(str);//2  strcpy(min, str); //2  for (n=1; n<5; n++)//3  {  gets(str); //2  if (strcmp(str, min) < 0) //4  {  strcpy(min, str); //2  }  }  printf("The minimum is:%s\n", min);//2  } |

Q493.(10分)输入6个字符串，输出最小串及最大串。

\*\*输入格式要求："%s" 提示信息："请输入6行字符串：\n"

\*\*输出格式要求："The max string is: %s\n" "The min string is: %s\n"

程序示例运行如下：

请输入6行字符串：

hello,world

vb

vc

Java

c++

c#

The max string is: vc

The min string is: Java

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 6  #define N1 20  int main()  {  char str[N1], min[N1], max[N1];  int i;  printf("请输入6行字符串：\n");  gets(min); //假设第1个串最小  strcpy(max, min);//假设最大串的值也为min  //循环输入其它串，并与最小的串及最大的串比较  for (i = 2; i <= N; i++)  {  gets(str);  if (strcmp(str, min) < 0)  strcpy(min, str);  if (strcmp(str, max) > 0)  strcpy(max, str);  }  printf("The max string is: %s\n", max);  printf("The min string is: %s\n", min);  return 0;  } |

Q295.(15分)编程实现找出字符串中最大字符元素并输出该元素及其对应的ASCII值.

\*\*\*\*要求输入提示信息为：

"Input a string:\n"

\*\*\*\*输出格式要求为：

"The largest character of \"%s\" is \'%c\' ,The ASCII is %d."

参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  main()  {  char string[81];  int i=0,max;  printf("Input a string:"); //1  printf("\n");  gets(string);  max=string[i]; //1  while(string[i]!='\0') //2  {  i=i+1;  if(string[i]>max) //2  max=string[i]; //1  }  printf("The largest character of \"%s\" is \'%c\' ,The ASCII is %d.",string,max,max); //2  } |

Q1155.(10分)输入一行字符，统计其中的英文字符、数字字符、空格字符，以及其他字符的个数。请找出以下程序的错误，并改正之。

#include <stdio.h>

#include <string.h>

#define ARR\_SIZE = 80;

main()

{

char str[ARR\_SIZE];

int len, i;

int letter=0,digit=0,space=0,other=0;

printf("请输入一个字符串:");

gets(str);

len = strlen(str);

for (i=0; i<len; i++)

{

if (a=<str[i]<=z || A=<str[i]<=Z)

{

letter ++;

}

else if (0=<str[i]<=9)

{

digit ++;

}

else if (str[i]=' ' )

{

space ++;

}

else

other ++;

}

printf("英文字符数:%d\n", letter);

printf("数字字符数:%d\n", digit);

printf("空格数:%d\n", space);

printf("其他字符数:%d\n", other);

}

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define ARR\_SIZE 80  main()  {  char str[ARR\_SIZE];  int len, i;  int letter=0,digit=0,space=0,other=0;  printf("请输入一个字符串:");  gets(str);  len = strlen(str);  for (i=0; i<len; i++)  {  if (str[i]>='a'&&str[i]<='z' || str[i]>='A'&&str[i]<='Z')  {  letter ++;  }  else if (str[i]>='0'&&str[i]<='9')  {  digit ++;  }  else if (str[i]==' ' )  {  space ++;  }  else  other ++;  }  printf("英文字符数:%d\n", letter);  printf("数字字符数:%d\n", digit);  printf("空格数:%d\n", space);  printf("其他字符数:%d\n", other);  } |

Q1119.(10分)下面的程序将数字进行加密，其中的每一个数字转化为另一个不同的数字。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

char s[] = "24635", c, i;

for (i = 0; c = s[i]; i++)

{

switch ((c - '0')

{

case 2:

case 3:

putchar(c + 4);

continue;

case 4:

putchar(c + 4);

break;

case 5:

putchar(c + 3);

default:

putchar(c + 2);

}

putchar('\n');

}

}

参考答案

|  |
| --- |
| #include <stdio.h>  main()  {  char s[] = "24635", c, i;  for (i = 0; c = s[i]; i++)  {  switch (c - '0')  {  case 2:  case 3:  putchar(c + 4);  continue;  case 4:  putchar(c + 4);  break;  case 5:  putchar(c + 3);  break;  default:  putchar(c + 2);  }  putchar('\n');  }  } |

Q360.(10分)输入英文的星期几，查找星期表后输出对应的数字。具体的算法是：若发现与表中某项相同，则输出该字符串在表中的位置（序号）；若查到表尾仍不同，则输出错误信息。设星期表的内容如下图所示。

0 Sunday

1 Monday

2 Tuesday

3 Wednesday

4 Thursday

5 Friday

6 Saturday

\*\*输入格式要求："%s" 提示信息："Enter a string:"

\*\*输出格式要求："%s is %d\n" "Error\n"

程序运行示例如下：

Enter a string:wednesday

wednesday is 3

参考答案

|  |
| --- |
| #include <stdio.h>  char week\_day[][10]={"sunday","monday","tuesday","wednesday","thursday","friday","saturday"};  main()  {  int i,j;  char id[10],c;  printf("Enter a string:");  scanf("%s",id);  for(i=0;i<7;i++)  {  for(j=0;(c=week\_day[i][j])==id[j]&&c!='\0';j++)  ;  if(c=='\0')  break;  }  if(i<7)  printf("%s is %d\n",id,i);  else  printf("Error\n");  } |

Q592.(10分)编写一个程序，将一个字符串s2插入到字符串s1中，其起始插入位置为n。

\*\*输入格式要求："%d" 提示信息："main string:" "sub string:" "site of begining:(<=%d)"

\*\*输出格式要求："After instert:%s\n"

程序运行示例如下：

main string:hello,world!

sub string:c

site of begining:(<=12)6

After instert:hello,c world!

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 100  int main()  {  int n, i, j, k, len;  char s1[N], s2[N], s3[2 \* N];  printf("main string:");  gets(s1);  printf("sub string:");  gets(s2);  len = strlen(s1);  do  {  printf("site of begining:(<=%d)", len);  scanf("%d", &n);  }  while (n > len);  for (i = 0; i < n; i++)  {  s3[i] = s1[i];  }  for (j = 0; s2[j] != '\0'; j++)  {  s3[i + j] = s2[j];  }  for (k = n; s1[k] != '\0'; k++)  {  s3[j + k] = s1[k];  }  s3[j + k] = '\0';  printf("After instert:%s\n", s3);  return 0;  } |

Q598.(10分)将字符串s1从第m个字符开始剩余的所有字符，送入字符数组s2中。

\*\*输入格式要求："%d" 提示信息："input a string:\n" "input start point:\n"

程序运行示例如下：

input a string:hello,world!

input start point:7

world!

参考答案

|  |
| --- |
| #include <stdio.h>  main()  {  int i, j, m;  char s1[80], s2[80];  printf("input a string:\n");  gets(s1);  printf("input start point:\n");  scanf("%d", &m);  i = m - 1;  j = 0;  while (s1[i] != '\0')  {  s2[j] = s1[i];  i++;  j++;  }  s2[j] = '\0';  puts(s2);  } |

Q1327.(10分)有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。

程序的运行示例如下：

请输入一个字符串：1234567890

从第几个字符开始复制？3

复制结果为：34567890

参考答案

|  |
| --- |
| #include <stdio.h>  copystr(char \*p1, char \*p2, int m)  {  int n;  n = 0;  while (n < m - 1)  {  n++;  p1++;  }  while (\*p1 != '\0')  {  \*p2 = \*p1;  p1++;  p2++;  }  \*p2 = '\0';  }  int main(void)  {  int m;  char str1[20], str2[20];  printf("请输入一个字符串：\n");  scanf("%s", str1);  printf("从第几个字符开始复制？");  scanf("%d", &m);  if (strlen(str1) < m)  {  printf("输入错误。");  }  else  {  copystr(str1, str2, m);  printf("复制结果为：%s", str2);  }  return 0;  } |

AddLHK

Q307.(15分)利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

参考答案

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

Q133.(5分)设计一个函数MinCommonMultiple,其功能是计算两个正整数的最小公倍数。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include <stdio.h>

int MinCommonMultiple(int a,int b)

main()

{

int a,b,x;

printf("Input a,b:");

scanf("%d,%d",a,b);

x = MinCommonMultiple(a,b);

printf("MinCommonMultiple = %d\n", x);

}

int MinCommonMultiple(int a,int b);

{

int i;

for(i=1; i<a\*b; i++)

{

if(i%a == 0 || i%b == 0)

return i;

}

}

参考答案

int MinCommonMultiple(int a,int b);

main()

{

int a,b,x;

printf(&quot;Input a,b:&quot;);

scanf(&quot;%d,%d&quot;,&amp;a,&amp;b);

x = MinCommonMultiple(a,b);

printf(&quot;MinCommonMultiple = %d\n&quot;, x);

}

int MinCommonMultiple(int a,int b)

{

int i;

for(i=1; i&lt;a\*b; i++)

{

if(i%a == 0 &amp;&amp; i%b == 0)

return i;

}

}

Q173.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

参考答案

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q191.(15分)用二维数组编程计算并输出n(n<20)行杨辉三角形。

\*\*输入提示信息要求："Input n(n<20):\n"

\*\*输入格式："%d"

\*\*输出要求："%4d"

每输出一行后就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include<stdio.h>

#define N 20

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

main() {

int a[N][N] = {0}, n;//1

printf("Input n(n<20):\n");

scanf("%d", &n);//1

CaculateYH(a, n);//1

PrintYH(a, n); //1

}

void CaculateYH(int a[][N], int n) {

int i, j;

for (i = 0; i < n; i++) //1

{

a[i][0] = 1;//1

a[i][i] = 1;//1

}

for (i = 2; i < n; i++)//1

{

for (j = 1; j <= i - 1; j++) //1

{

a[i][j] = a[i - 1][j - 1] + a[i - 1][j];//1

}

}

}

void PrintYH(int a[][N], int n) {

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j <= i; j++) //1

{

printf("%4d", a[i][j]);//1

}

printf("\n");//1

}

}

Q235.(10分)用函数编程计算两整数的最大值，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大值。

\*\*输入格式要求："%d%d" 提示信息："Input a,b:"

\*\*输出格式要求："max = %d\n"

程序运行示例如下：

Input a,b:5 8

max = 8

参考答案

#include <stdio.h>

int Max(int a, int b);

int main()

{

int a, b;

printf("Input a,b:");

scanf("%d%d", &a, &b);

printf("max = %d\n", Max(a, b));

return 0;

}

/\* 函数功能：计算a和b的最大值 \*/

int Max(int a, int b)

{

return a > b ? a : b;

}

Q602.(10分)一辆卡车违反了交通规则，撞人后逃逸。现场有三人目击该事件，但都没有记住车号，只记住车号的一些特征。甲说：车号的前两位数字是相同的；乙说：车号的后两位数字是相同的，但与前两位不同；丙是位数学家，他说：4位的车号正好是一个整数的平方。现在请根据以上线索帮助警方找出车号以便尽快破案。

\*\*输出格式要求："The number is:%d\n"

参考答案

#include<stdio.h>

int main()

{

int a, b, n;

for (a = 1; a <= 9; a++)

{

for (b = 1; b <= 9; b++)

{

for (n = 1; n <= 100; n++)

{

if (1000 \* a + 100 \* a + 10 \* b + b == n \* n)

printf("The number is:%d\n", n \* n);

}

}

}

return 0;

}

Q282.(10分)打印所有的水仙花数。

所谓“水仙花数”是指一个三位数，其各位数字立方和等于该数本身。

例如153是“水仙花数”，因为153 = 1\*1\*1+3\*3\*3+5\*5\*5。

\*\*输出格式要求为"%6d"。

参考答案

#include <stdio.h>

int main()

{

int i,j,k,n;

for(n=100;n<1000;n++) //1

{

i=n/100; //1

j=(n-i\*100)/10; //1

k=n-i\*100-j\*10; //1

if(i\*i\*i+j\*j\*j+k\*k\*k==n) //1

printf("%6d",n);

}

printf("\n");

return 0;

}

Q413.(10分)马克思手稿中的数学问题。马克思手稿中有一道趣味数学题：有30个人，其中有男人、女人和小孩，在一家饭馆吃饭共花了50先令：每个男人花3先令，每个女人花2先令，每个小孩花1先令，问男人、女人和小孩各几人？

\*\*输入格式要求：无

\*\*输出格式要求："\tMEN\tWOMEN\tCHILDREN\n"

"-----------------------------------------\n"

"%2d:\t%d\t%d\t%d\n"

程序运行示例如下：

MEN WOMEN CHILDREN

-----------------------------------------

1: 0 20 10

2: 1 18 11

3: 2 16 12

4: 3 14 13

5: 4 12 14

6: 5 10 15

7: 6 8 16

8: 7 6 17

9: 8 4 18

10: 9 2 19

11: 10 0 20

12: 11 -2 21

13: 12 -4 22

14: 13 -6 23

15: 14 -8 24

16: 15 -10 25

17: 16 -12 26

参考答案

#include <stdio.h>

main()

{

int x, y, z, count = 0;

printf("\tMEN\tWOMEN\tCHILDREN\n");

printf("-----------------------------------------\n");

for (x = 0; x <= 16; x++)

{

y = 20 - 2 \* x;

z = 30 - x - y;

if (3 \* x + 2 \* y + z == 50)

printf("%2d:\t%d\t%d\t%d\n", ++count, x, y, z);

}

}

Q319.(10分)从键盘输入一个字符，如果它是大写的英文字母，则将其转换成小写

的英文字母，然后将转换的字符及其ASCII码值显示在屏幕上；如果它

是小写的英文字母，则将其转换成大写的英文字母，然后将转换的字符

及其ASCII码值显示在屏幕上；如果输入的是其它字符则显示输入错误。

\*\*输入提示信息格式: 无

\*\*输入数据格式要求: %c

\*\*输出数据格式要求:

若输入的是大小写英文字母，则输出: "%c,%d"

若输入的是其它字符，则输出: "Input error!"

注：已知大写英文字母A的ASCII码值是65

小写英文字母a的ASCII码值是97

参考答案

#include <stdio.h>

int main()

{

char ch;

scanf("%c", &ch); //1

if (ch >= 'A' && ch <= 'Z') //1

{

ch = ch + 32; //1

printf("%c,%d", ch, ch);

}

else if (ch >= 'a' && ch <= 'z') //1

{

ch = ch - 32; //1

printf("%c,%d", ch, ch);

}

else //1

{

printf("Input error!");

}

return 0;

}

Q408.(10分)四位反序数。设N是一个四位数，它的9倍恰好是其反序数，求N。反序数就是将整数的数字倒过来形成的整数，例如：1234的反序数是4321。

\*\*输入格式要求：无

\*\*输出格式要求："The number satisfied states condition is : %d\n"

参考答案

#include <stdio.h>

main()

{

int i;

for(i=1002;i<1111;i++)

if(i%10\*1000+i/10%10\*100+i/100%10\*10+i/1000==i\*9)

printf("The number satisfied states condition is : %d\n",i);

}

Q209.(10分)已知三角形的三边长为a，b，c，计算三角形面积的公式为：

area =sqrt(s(s-a)(s-b)(s-c)),s=1/2(a+b+c)

其中，a，b，c为浮点数，sqrt为开平方计算。

要求编写程序，从键盘输入a，b，c的值，计算并输出三角形的面积。

注意：不用判断输入的边长值是否满足三角形要求，

只需进行简单计算即可。

\*\*要求输入提示信息格式为："Input a,b,c:\n"；

\*\*输出格式要求："area=%.2f\n"

参考答案

#include <stdio.h>

#include <math.h>

main()

{

float a, b, c;

float s, area;

printf("Input a,b,c:");

scanf("%f,%f,%f",&a,&b,&c); //1

s = 1.0 / 2 \* (a + b + c); //1

area = sqrt(s \* (s - a) \* (s - b) \* (s - c)); //1

printf("area=%.2f\n", area); //1

}

编程输出如下菜单界面：

=======大学信息管理系统=======

------------------------------

   办公室管理      财务管理

   教务管理        图书管理

   科研管理        设备管理

   人事管理        后勤管理

   退出系统

------------------------------

请您在上述功能中选择（0——8）：

#include <stdio.h>

int main()

{

    printf("=======大学信息管理系统======= \n");

    printf("------------------------------\n");

    printf("1. 办公室管理     5. 财务管理 \n");

    printf("2. 教务管理       6. 图书管理 \n");

    printf("3. 科研管理       7. 设备管理 \n");

    printf("4. 人事管理       8. 后勤管理 \n");

    printf("0. 退出系统\n");

    printf("------------------------------\n");

    printf("请您在上述功能中选择（0——8）:\n");

    return 0;

}

Q308.(10分)体型判断。医务工作者经广泛的调查和统计分析，根据身高与体重因素给出了以下按“体指数”进行体型判断的方法：

体指数t = 体重w /(身高h\*h) ，其中w单位为千克，h单位为米

当 t<18 时，为低体重；

当 18≤t<25 时，为正常体重；

当 25≤t<27 时，为超重体重；

当 t≥27 时，为肥胖。

根据上述给定的公式计算体指数t，然后判断你的体重属于何种类型。

\*\*输入提示信息格式："Please enter h,w:\n"

\*\*输入数据格式要求："%f,%f"

\*\*输出数据格式要求:

当 t<18 时，输出："Lower weight!\n"

当 18≤t<25 时，输出："Standard weight!\n"

当 25≤t<27 时，输出："Higher weight!\n"

当 t≥27 时， 输出："Too fat!\n"

#include <stdio.h>

int main()

{

float h, w, t;

printf("Please enter h,w:\n");

scanf("%f,%f", &h, &w); //1

t = w / (h \* h); //1

if (t < 18) //1

{

printf("Lower weight!\n");

}

if (t >= 18 && t < 25) //1

{

printf("Standard weight!\n");

}

if (t >= 25 && t < 27) //1

{

printf("Higher weight!\n");

}

if (t >= 27) //1

{

printf("Too fat!\n");

}

return 0;

}

Q241.(10分)在一种室内互动游戏中，魔术师要每位观众心里想一个三位数abc（a、b、c分别是百位、十位和个位数字），然后魔术师让观众心中记下acb、bac、bca、cab、cba五个数以及这5个数的和值。只要观众说出这个和是多少，则魔术师一定能猜出观众心里想的原数abc是多少。例如，观众甲说他计算的和值是1999，则魔术师立即说出他想的数是443，而观众乙说他计算的和值是1998，则魔术师说：“你算错了！”。请编程模拟这个数字魔术游戏。

\*\*输入格式要求："%d" 提示信息："Input a sum:"

\*\*输出格式要求："The sum you calculated is wrong!\n" "The number is %d\n"

程序运行示例如下：

Input a sum:1999

The number is 443

#include <stdio.h>

int Magic(int m);

int main()

{

int m, ret;

printf("Input a sum:");

scanf("%d", &m);

ret = Magic(m);

if (ret != 1)

{

printf("The sum you calculated is wrong!\n");

}

return 0;

}

int Magic(int m)

{

int a, b, c, n;

for (a = 1; a < 9; a++)

{

for (b = 1; b < 9; b++)

{

for (c = 1; c < 9; c++)

{

n = 122 \* a + 212 \* b + 221 \* c;

if (m == n)

{

printf("The number is %d\n", 100 \* a + 10 \* b + c);

return 1;

}

}

}

}

return 0;

}

Q1246.(10分)

|  |
| --- |
| 下面程序用于输入三角形的三条边a,b,c，判断它们能否构成三角形，若能则指出是何种三角形：等腰三角形、直角三角形、一般三角形。（找出其中的3处错误并改正之）  #include <stdio.h>  #include <math.h>    main()  {      float   a, b, c;      int flag = 1;        scanf("%d,%d,%d", &a, &b , &c);      if (a+b>c && b+c>a && a+c>b)      {      if (fabs(a-b)<=0.1||fabs(b-c)<=0.1||fabs(c-a)<=0.1)      {          printf("等腰三角形\n");          flag = 0;      }      else if (fabs(a\*a+b\*b-c\*c)<=0.1                  ||fabs(a\*a+c\*c-b\*b)<=0.1                  ||fabs(c\*c+b\*b-a\*a)<=0.1)      {          printf("直角三角形\n");          flag = 0;      }      if (!flag)      {          printf("一般三角形\n");      }      }      else      {      printf("不是三角形\n");      }  } |

#include <stdio.h>

#include <math.h>

main()

{

float a, b, c;

int flag = 1;

scanf("%f, %f, %f", &a, &b , &c); //原来%d错

if (a+b>c && b+c>a && a+c>b)

{

if (fabs(a-b)<=0.1||fabs(b-c)<=0.1||fabs(c-a)<=0.1)

{

printf("等腰三角形\n");

flag = 0;

}

if (fabs(a\*a+b\*b-c\*c)<=0.1

||fabs(a\*a+c\*c-b\*b)<=0.1

||fabs(c\*c+b\*b-a\*a)<=0.1) //原来多了else

{

printf("直角三角形\n");

flag = 0;

}

if (flag) //原来flag前面多了!

{

printf("一般三角形\n");

}

}

else

{

printf("不是三角形\n");

}

}

Q780.(5分)改错题：用户从键盘任意输入一个数字表示月份值n，程序显示该月份对应的英文表示，若n不在1～12之间，则输出“Illegal month”。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

|  |
| --- |
| #include <stdio.h>  main()  {      int n;      char monthName[][20] = {"Illegal month", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"};        printf("Input month number:");      scanf("%d", n);      if ((n < 12) || (n > 1))      {          printf("month %d is %c\n", n, monthName[n][n]);        }      else      {          printf("%s\n", monthName[n]);      }  } |

#include <stdio.h>

main()

{

int n;

char monthName[][20] = {"Illegal month", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"};

printf("Input month number:");

scanf("%d", &n);

if ((n <= 12) && (n >= 1))

{

printf("month %d is %s\n", n, monthName[n]);

}

else

{

printf("%s\n", monthName[0]);

}

}

Q370.(10分)编写函数，通过指针求字符串的长度。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："string length=%d\n"

程序运行示例如下：

Enter string:nihao

string length=5

#include <stdio.h>

strlen(char \*s)

{

    char \*p=s;

    while(\*p)

        p++;

    return p-s;

}

main()

{

    char a[50];

    printf("Enter string:");

    scanf("%s",a);

    printf("string length=%d\n",strlen(a));

}

Q1310.(10分)打印出以下图案：

\*

\* \* \*

\* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*

#include <stdio.h>

int main()

{

int i, j, k;

for (i = 1; i <= 4; i++)

{

for (j = 1; j <= (4 - i) + 5; j++)

{

printf(" ");

}

for (k = 1; k <= (2 \* i - 1) + 5; k++)

{

if (k <= 5)

{

printf(" ");

}

else

{

printf(" \* ");

}

}

printf("\n");

}

for (i = 1; i <= 3; i++)

{

for (j = 1; j <= i + 5; j++)

{

printf(" ");

}

for (k = 1; k <= (7 - 2 \* i) + 5; k++)

{

if (k <= 5)

{

printf(" ");

}

else

{

printf(" \* ");

}

}

printf("\n");

}

return 0;

}

Q341.(10分)求555555的约数中最大的三位数是多少？

\*\*输入格式要求："%ld" 提示信息："Please input number"

\*\*输出格式要求："The max factor with 3 digits in %ld is %d.\n"

程序运行示例如下：

Please input number 555555

The max factor with 3 digits in 555555 is 777.

#include <stdio.h>

main()

{

int j; long n;

printf("Please input number");

scanf("%ld",&n);

for(j=999;j>=100;j--)

if(n%j==0)

{

printf("The max factor with 3 digits in %ld is %d.\n",n,j);

break;

}

}

Q1283.(10分)

|  |
| --- |
| 程序改错。以下程序用于打印输出结构体数组的所有元素。  struct s  {      char name[10];      int age;  }  main()  {      struct s a[3] = {”John”,19,”Paul”,17,”Marry”,18};      int \*p;        for(p = a; p < 3; p++)      {          printf("%s,%d\n", p->name, p->age );      }    }  #include <stdio.h> |
| struct s  {      char name[10];      int age;  };  main()  {      struct s a[3] = {"John",19,"Paul",17,"Marry",18};      struct s \*p;        for(p = a; p < a+3; p++)      {          printf("%s,%d\n", p->name, p->age );      }    } |

Q378.(10分)

\*汉诺塔问题是一个著名的问题，初始模型如图所示。其来源据说是在约19世纪末欧洲的商店中出售一种智力玩具，在一块铜板上有三根杆，最左边的杆自上而下、由小到大顺序串着64个圆盘构成的塔，游戏的目的是将最左边A杆上的圆盘，借助最右边的C杆，全部移动到中间的B杆上，条件是一次仅能移动一个盘，且不允许大盘放在小盘的上面。

\*\*输入格式要求："%d" 提示信息："Please enter the number of discs:"

\*\*输出格式要求："\tTotal:%d\n" "("%2d-(%2d):%c==>%c\n"

程序运行示例如下：

Please enter the number of discs:

Please enter the number of discs: 5

 1-( 1):a==>b

 2-( 2):a==>c

 3-( 1):b==>c

 4-( 3):a==>b

 5-( 1):c==>a

 6-( 2):c==>b

 7-( 1):a==>b

 8-( 4):a==>c

 9-( 1):b==>c

10-( 2):b==>a

11-( 1):c==>a

12-( 3):b==>c

13-( 1):a==>b

14-( 2):a==>c

15-( 1):b==>c

16-( 5):a==>b

17-( 1):c==>a

18-( 2):c==>b

19-( 1):a==>b

20-( 3):c==>a

21-( 1):b==>c

22-( 2):b==>a

23-( 1):c==>a

24-( 4):c==>b

25-( 1):a==>b

26-( 2):a==>c

27-( 1):b==>c

28-( 3):a==>b

29-( 1):c==>a

30-( 2):c==>b

31-( 1):a==>b

Total:31

|  |
| --- |
| #include <stdio.h>  int i=0;  main()  {      unsigned n;      printf("Please enter the number of discs:");      scanf("%d",&n);      movedisc(n,'a','b','c');      printf("\tTotal:%d\n",i);  }  movedisc(unsigned n,char fromneedle,char toneedle,char usingneedle)  {      if(n==1)          printf("%2d-(%2d):%c==>%c\n",++i,n,fromneedle,toneedle);      else      {          movedisc(n-1,fromneedle,usingneedle,toneedle);          printf("%2d-(%2d):%c==>%c\n",++i,n,fromneedle,toneedle);          movedisc(n-1,usingneedle,toneedle,fromneedle);      }  }Q378.(10分)  \*汉诺塔问题是一个著名的问题，初始模型如图所示。其来源据说是在约19世纪末欧洲的商店中出售一种智力玩具，在一块铜板上有三根杆，最左边的杆自上而下、由小到大顺序串着64个圆盘构成的塔，游戏的目的是将最左边A杆上的圆盘，借助最右边的C杆，全部移动到中间的B杆上，条件是一次仅能移动一个盘，且不允许大盘放在小盘的上面。  \*\*输入格式要求："%d" 提示信息："Please enter the number of discs:"  \*\*输出格式要求："\tTotal:%d\n" "("%2d-(%2d):%c==>%c\n"  程序运行示例如下：  Please enter the number of discs:  Please enter the number of discs: 5  1-( 1):a==>b  2-( 2):a==>c  3-( 1):b==>c  4-( 3):a==>b  5-( 1):c==>a  6-( 2):c==>b  7-( 1):a==>b  8-( 4):a==>c  9-( 1):b==>c  10-( 2):b==>a  11-( 1):c==>a  12-( 3):b==>c  13-( 1):a==>b  14-( 2):a==>c  15-( 1):b==>c  16-( 5):a==>b  17-( 1):c==>a  18-( 2):c==>b  19-( 1):a==>b  20-( 3):c==>a  21-( 1):b==>c  22-( 2):b==>a  23-( 1):c==>a  24-( 4):c==>b  25-( 1):a==>b  26-( 2):a==>c  27-( 1):b==>c  28-( 3):a==>b  29-( 1):c==>a  30-( 2):c==>b  31-( 1):a==>b  Total:31  #include <stdio.h>  int i=0;  main()  {  unsigned n;  printf("Please enter the number of discs:");  scanf("%d",&n);  movedisc(n,'a','b','c');  printf("\tTotal:%d\n",i);  }  movedisc(unsigned n,char fromneedle,char toneedle,char usingneedle)  {  if(n==1)  printf("%2d-(%2d):%c==>%c\n",++i,n,fromneedle,toneedle);  else  {  movedisc(n-1,fromneedle,usingneedle,toneedle);  printf("%2d-(%2d):%c==>%c\n",++i,n,fromneedle,toneedle);  movedisc(n-1,usingneedle,toneedle,fromneedle);  }  }  Q607.(10分)请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);  \*\*输入格式要求：无  \*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身  程序运行的输入输出样例1：  输入：  abc  输出：  No!  程序运行的输入输出样例2：  输入：  12321  输出：  12321  #include <stdio.h>  #include <string.h>  int Judger( char a[ ]);  int main()  {  char a[80];  int flag;  gets(a);  flag = Judger(a);  if (flag == 1)  {  puts(a);  }  else  {  printf("No!");  }  return 0;  }  int Judger( char a[ ])  {  char \*pStar, \*pEnd;  int i, j, flag = 1;  j = strlen(a);  pStar = &a[0];  pEnd = &a[j - 1];  for (i = 0; i < (j / 2); i++)  {  if (\*pStar != \*pEnd)  {  flag = 0;  }  pStar++;  pEnd--;  }  return flag;  }  Q357.(10分)某班期终考试科目为高等数学MT，英语EN和物理PH，有5人参加考试。为评定奖学金，要求统计并输出一个表格，表格内容包括学号、各科分数、总分和平均分，并标出三门课均在90分以上者（该栏标志输出“Y”，否则输出“N”），表格形式如下：  NO MT EN PH SUM V >90  ------------------------------------------------------  1 88 87 83 258 86 N  2 69 90 88 247 82 N  3 77 70 79 226 75 N  4 96 99 98 293 97 Y  5 77 88 90 255 85 N  \*\*输入格式要求："%d" 提示信息："Enter No. and score as: MT EN PH\n"  \*\*输出格式要求：  "NO\tMT\tEN\tPH\tSUM\tV\t>90\n"  "------------------------------------------------------\n"  "%d\t%d\t%d\t%d\t%d\t%d\t%c\n"  #include <stdio.h>  main()  {  int i, j, a[5][4], s = 0;  char c;  for (i = 0; i <= 4; i++)  {  printf("Enter No. and score as: MT EN PH\n");  for (j = 0; j <= 3; j++)  scanf("%d", &a[i][j]);  }  printf("NO\tMT\tEN\tPH\tSUM\tV\t>90\n");  printf("------------------------------------------------------\n");  for (i = 0; i <= 4; i++)  {  printf("%d\t", a[i][0]);  for (s = 0, j = 1; j <= 3; j++)  {  s += a[i][j];  printf("%d\t", a[i][j]);  }  if (a[i][1] >= 90 && a[i][2] >= 90 && a[i][3] >= 90)  c = 'Y';  else c = 'N';  print#include <stdio.h>  main()  {  int i, j, a[5][4], s = 0;  char c;  for (i = 0; i <= 4; i++)  {  printf("Enter No. and score as: MT EN PH\n");  for (j = 0; j <= 3; j++)  scanf("%d", &a[i][j]);  }  printf("NO\tMT\tEN\tPH\tSUM\tV\t>90\n");  printf("------------------------------------------------------\n");  for (i = 0; i <= 4; i++)  {  printf("%d\t", a[i][0]);  for (s = 0, j = 1; j <= 3; j++)  {  s += a[i][j];  printf("%d\t", a[i][j]);  }  if (a[i][1] >= 90 && a[i][2] >= 90 && a[i][3] >= 90)  c = 'Y';  else c = 'N';  printf("%d\t%d\t%c\n", s, s / 3, c);  }  }f("%d\t%d\t%c\n", s, s / 3, c);  }  }  Q1615.(10分)按如下函数原型编写程序，用函数编程实现两个数组中对应元素值的交换。  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void Swap(int \*x, int \*y);  输入提示信息：  "Input array size(n<=10):"  "Input array a:"  "Input array b:"  输入格式："%d"  输出提示信息：  "Output array a:"  "Output array b:"  输出格式："%5d" |

#include <stdio.h>

#define N 10

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void Swap(int \*x, int \*y);

int main()

{

int a[N], b[N], i, n;

printf("Input array size(n<=10):");

scanf("%d", &n);

printf("Input array a:");

ReadData(a, n);

printf("Input array b:");

ReadData(b, n);

for (i = 0; i < n; i++)

{

Swap(&a[i], &b[i]);

}

printf("Output array a:");

PrintData(a, n);

printf("Output array b:");

PrintData(b, n);

return 0;

}

/\* 函数功能：输入数组a的n个元素值 \*/

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]);

}

}

/\* 函数功能：输出数组a的n个元素值 \*/

void PrintData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

printf("%5d", a[i]);

}

printf("\n");

}

/\* 函数功能：两整数值互换 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

Q256.(10分)

按如下函数原型，采用梯形法编程实现，在积分区间[a,b]内计算函数$y\_1=\int\_{0}^{1}(1+x^2)\,dx$和$y\_2=\int\_{0}^{3}\frac{x}{1+x^2}\,dx$的定积分。其中，指向函数的指针变量f用于接收被积函数的入口地址。 Integral(float (\*f)(float), float a, float b);

\*\*输出格式要求："y1=%f\ny2=%f\n"

#include <stdio.h>

float Fun1(float x);

float Fun2(float x);

float Integral(float (\*f)(float), float a, float b);

int main()

{

float y1, y2;

y1 = Integral(Fun2, 0.0, 3.0);

y2 = Integral(Fun1, 0.0, 1.0);

printf("y1=%f\ny2=%f\n", y1, y2);

return 0;

}

/\* 函数功能：计算函数1+x\*x的函数值 \*/

float Fun1(float x)

{

return 1 + x \* x;

}

/\* 函数功能：计算函数x/(1+x\*x)的函数值 \*/

float Fun2(float x)

{

return x / (1 + x \* x);

}

/\* 函数功能：用梯形法计算函数的定积分 \*/

float Integral(float (\*f)(float), float a, float b)

{

float s, h;

int n = 100, i;

s = ((\*f)(a) + (\*f)(b)) / 2;

h = (b - a) / n;

for (i = 1; i < n; i++)

{

s += (\*f)(a + i \* h);

}

return s \* h;

}

Q1688.(10分)按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

输入某年某月某日，计算并输出它是这一年的第几天。

/\* 函数功能： 对给定的某年某月某日，计算它是这一年的第几天

函数参数： 整型变量year、month、day，分别代表年、月、日

函数返回值：这一年的第几天 \*/

int DayofYear(int year, int month, int day);

输入提示信息："Please enter year, month, day:"

输入格式："%d,%d,%d"

输出提示信息和格式："yearDay = %d\n"

#include <stdio.h>

int DayofYear(int year, int month, int day);

int dayTab[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}, {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

int main()

{

int year, month, day, yearDay;

printf("Please enter year, month, day:");

scanf("%d,%d,%d", &year, &month, &day);

yearDay = DayofYear(year, month, day);

printf("yearDay = %d\n", yearDay);

return 0;

}

/\* 函数功能：对给定的某年某月某日，计算并返回它是这一年的第几天 \*/

int DayofYear(int year, int month, int day)

{

int i, leap;

/\* 若year为闰年，即leap值为1，则用第1行元素dayTab[1][i]计算；

否则leap值为0，用第0行dayTab[0][i]计算 \*/

leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

for (i = 1; i < month; i++)

{

day = day + dayTab[leap][i];

}

return day; /\* 返回计算出的day的值 \*/

}

Q255.(10分)从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并返回它们所在数组中的位置。

\*\*输入格式要求："%d" 提示信息："Input n(n<=10):" "Input %d numbers:"

\*\*输出格式要求："%5d" "Exchange results:"

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

void Swap(int \*x, int \*y);

int main()

{

int a[10], n;

printf("Input n(n<=10):");

scanf("%d", &n);

printf("Input %d numbers:", n);

ReadData(a, n);

MaxMinExchang(a, n);

printf("Exchange results:");

PrintData(a, n);

return 0;

}

/\* 函数功能：输入数组a的n个元素值 \*/

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]);

}

}

/\* 函数功能：输出数组a的n个元素值 \*/

void PrintData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

printf("%5d", a[i]);

}

printf("\n");

}

/\* 函数功能：将数组a中的最大数与最小数位置互换 \*/

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;

int i;

for (i = 1; i < n; i++)

{

if (a[i] > maxValue)

{

maxValue = a[i];

maxPos = i;

}

if (a[i] < minValue)

{

minValue = a[i];

minPos = i;

}

}

Swap(&a[maxPos], &a[minPos]);

}

/\* 函数功能：两整数值互换 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

按如下格式打印100以内整数的平方根表。

#include <stdio.h>

#include <math.h>

main()

{

int m, n;

for (m = 0; m < 10; m++)

{

printf("%7d", m); //打印表头

}

printf("\n");

for (n = 0; n < 10; n++) //乘数n从1变化到9

{

printf("%d", n); //输出每行的开头数字

for (m = 0; m < 10; m++) //被乘数m从1变化到9

{

printf("%7.3f", sqrt(n \* 10 + m));

} //输出第m行n列中的值

printf("\n"); //输出换行符,准备打印下一行

}

}

Q140.(15分)按如下函数原形编程计算并输出求m\*m阶矩阵的转置矩阵。期中m的值由用户从键盘输入。已知m的值都不超过10。

求转置矩阵函数原形：void Transpose(int a[][M], int at[][M], int m);

\*\*\*\*输入提示信息和格式要求为：

"Input m"

"%d"

"Input %d\*%d matrix:\n"

"%d"

\*\*\*\*输出提示信息和格式要求为：

"The transposed matrix is:\n"

"%8d"

#include <stdio.h>

#define M 10

void Transpose(int a[][M], int at[][M], int m);

int main()

{

int a[M][M],at[M][M];

int m;

int i,j;

printf("Input m:"); //1

scanf("%d",&m);

printf("Input %d\*%d matrix:\n",m,m); //1

for (i=0;i<m;i++)

{

for (j=0;j<m;j++)

{

scanf("%d",&a[i][j]);//1

}

}

Transpose(a,at,m); //2

for (i=0;i<m;i++)

{

for (j=0;j<m;j++)

{

printf("%8d",at[i][j]);//1

}

printf( "\n" );

}

return 0;

}

void Transpose(int a[][M], int at[][M], int m)

{

int i,j;

for (i=0;i<m;i++) //1

{

for (j=0;j<m;j++) //1

{

at[i][j]=a[j][i]; //1

}

}

}#include <stdio.h>

#define M 10

void Transpose(int a[][M], int at[][M], int m);

int main()

{

int a[M][M],at[M][M];

int m;

int i,j;

printf("Input m:"); //1

scanf("%d",&m);

printf("Input %d\*%d matrix:\n",m,m); //1

for (i=0;i<m;i++)

{

for (j=0;j<m;j++)

{

scanf("%d",&a[i][j]);//1

}

}

Transpose(a,at,m); //2

for (i=0;i<m;i++)

{

for (j=0;j<m;j++)

{

printf("%8d",at[i][j]);//1

}

printf( "\n" );

}

return 0;

}

void Transpose(int a[][M], int at[][M], int m)

{

int i,j;

for (i=0;i<m;i++) //1

{

for (j=0;j<m;j++) //1

{

at[i][j]=a[j][i]; //1

}

}

}

Q263.(10分)从键盘任意输入一个整型表示的月份值，用指针数组编程输出该月份的英文表示，若输入的月份值不在1～12之间，则输出“Illegal month”。

\*\*输入格式要求："%d" 提示信息："Input month number:"

\*\*输出格式要求："month %d is %s\n"

"Illegal month", "January", "February", "March", "April", "May", "June", "July", August", "September", "October", "November", "December"

程序运行示例1如下：

Input month number:5

month 5 is May

程序运行示例2如下：

Input month number:13

Illegal month

|  |
| --- |
| #include <stdio.h>  int main()  {      int n;      static char \*monthName[] = {"Illegal month", "January", "February",                                  "March", "April", "May", "June", "July",                                  "August", "September", "October",                                  "November", "December"                                 };      printf("Input month number:");      scanf("%d", &n);      if ((n <= 12) && (n >= 1))      {          printf("month %d is %s\n", n, monthName[n]);/\* 输出相应月份 \*/      }      else      {          printf("%s\n", monthName[0]);        /\* 输出错误提示信息 \*/      }      return 0;  } |

Q186.(15分)两个正整数的最大公约数是能够整除这两个整数的最大整数。

请按如下函数原型编写计算最大公约数的函数Gcd()。

int Gcd(int a, int b);

在主函数中调用该函数计算，并输出从键盘任意输入的两整数的最大公约数。

\*\*要求输入提示信息为: "Input a,b:\n"

\*\*要求输入格式为: "%d,%d"

\*\*要求输出格式为

如果a或b不是正整数，输出: "Input number should be positive!\n"

否则输出: "Greatest Common Divisor of %d and %d is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

|  |
| --- |
| #include <stdio.h>  int Gcd(int a, int b);  main()  {      int a, b, c;      printf("Input a,b:\n");      scanf("%d,%d", &a, &b); //1      c = Gcd(a, b);  //1      if (c != -1) //1      {          printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c); //1      }      else      {          printf("Input number should be positive!\n"); //1      }  }  int Gcd(int a, int b) //1  {      int i, t;      if (a <= 0 || b <= 0) //1      {          return -1; //1      }      t = a < b ? a : b;  //2      for (i = t; i > 0; i--)//2      {          if (a % i == 0 && b % i == 0)//1             return i;  //1      }      return 1; //1  } |

Q1681.(10分)按如下函数原型编程计算并输出n×n阶矩阵的转置矩阵。其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int (\*a)[N], int n);

void Swap(int \*x, int \*y);

void InputMatrix(int (\*a)[N], int n);

void PrintMatrix(int (\*a)[N], int n);

输入提示信息："Input n:"

输入格式："%d"

输出提示信息和格式："The transposed matrix is:\n"

|  |
| --- |
| #include <stdio.h>  #define N 10  void  Swap(int \*x, int \*y);  void Transpose(int (\*a)[N], int n);  void InputMatrix(int (\*a)[N], int n);  void PrintMatrix(int (\*a)[N], int n);  int main()  {      int s[N][N], n;      printf("Input n:");      scanf("%d", &n);      InputMatrix(s, n);      Transpose(s, n);      printf("The transposed matrix is:\n");      PrintMatrix(s, n);      return 0;  }  /\* 函数功能：交换两个整型数的值 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  }  /\* 函数功能：计算n\*n矩阵的转置矩阵 \*/  void Transpose(int (\*a)[N], int n)  {      int i, j;      for (i = 0; i < n; i++)      {          for (j = i; j < n; j++)          {              Swap(\*(a + i) + j, \*(a + j) + i);          }      }  }  /\* 函数功能：输入n\*n矩阵的值 \*/  void InputMatrix(int (\*a)[N], int n)  {      int i, j;      printf("Input %d\*%d matrix:\n", n, n);      for (i = 0; i < n; i++)      {          for (j = 0; j < n; j++)          {              scanf("%d", \*(a + i) + j);          }      }  }  /\* 函数功能：输出n\*n矩阵的值 \*/  void PrintMatrix(int (\*a)[N], int n)  {      int i, j;      for (i = 0; i < n; i++)      {          for (j = 0; j < n; j++)          {              printf("%d\t", \*(\*(a + i) + j));          }          printf("\n");      }  } |

Q269.(25分)编程模拟洗牌和发牌过程。一副扑克有52张牌，分为4种花色（suit）：黑桃（Spades）、红桃（Hearts）、草花（Clubs）、方块（Diamonds）。每种花色又有13张牌面（face）：A，2，3，4，5，6，7，8，9，10，Jack，Queen，King。要求用结构体数组card表示52张牌，每张牌包括花色和牌面两个字符型数组类型的数据成员。

#include <stdio.h>

#include <string.h>

#include <time.h>

#include <stdlib.h>

typedef struct card

{

char suit[10];

char face[10];

} CARD;

void Deal(CARD \*wCard);

void Shuffle(CARD \*wCard);

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[]);

int main()

{

char \*suit[] = {"Spades", "Hearts", "Clubs", "Diamonds"};

char \*face[] = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10",

"Jack", "Queen", "King"

};

CARD card[52];

srand (time(NULL));

FillCard(card, face, suit);

Shuffle(card);

Deal(card);

return 0;

}

/\* 函数功能：花色按黑桃、红桃、草花、方块的顺序，面值按A~K的顺序，排列52张牌 \*/

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[])

{

int i;

for (i = 0; i < 52; i++)

{

strcpy(wCard[i].suit, wSuit[i / 13]);

strcpy(wCard[i].face, wFace[i % 13]);

}

}

/\* 函数功能：将52张牌的顺序打乱以模拟洗牌过程 \*/

void Shuffle(CARD \*wCard)

{

int i, j;

CARD temp;

for (i = 0; i < 52; i++) /\*每次循环产生一个随机数，交换当前牌与随机数指示的牌\*/

{

j = rand() % 52; /\* 每次循环产生一个0~51的随机数 \*/

temp = wCard[i];

wCard[i] = wCard[j];

wCard[j] = temp;

}

}

/\* 函数功能：输出每张牌的花色和面值以模拟发牌过程 \*/

void Deal(CARD \*wCard)

{

int i;

for (i = 0; i < 52; i++)

{

printf("%9s%9s%c", wCard[i].suit, wCard[i].face, i % 2 == 0 ? '\n' : '\t');

}

printf("\n");

}

Q268.(30分)编程统计候选人的得票数。设有3个候选人zhang、li、wang，10个选民，选民每次输入一个得票的候选人的名字，若选民输错候选人姓名，则按废票处理。选民投票结束后程序自动显示各候选人的得票结果和废票信息。要求用结构体数组candidate表示3个候选人的姓名和得票结果。

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #include  <string.h>  #define NUM\_ELECTORATE 10  #define NUM\_CANDIDATE 3  struct candidate  {      char  name[20];      int   count;  } candidate[3] = {"li", 0, "zhang", 0, "wang", 0};  int main()  {      int  i, j, flag = 1, wrong = 0;      char  name[20];      for (i = 1; i <= NUM\_ELECTORATE; i++)      {          printf("Input vote %d:", i);          scanf("%s", name);          flag = 1;          for (j = 0; j < NUM\_CANDIDATE; j++)          {              if (strcmp(name, candidate[j].name) == 0)              {                  candidate[j].count++;                  flag = 0;              }          }          if (flag)          {              wrong++;  /\* 废票计数 \*/              flag = 0;          }      }      printf("Election results:\n");      for (i = 0; i < NUM\_CANDIDATE; i++)      {          printf("%8s:%d\n", candidate[i].name, candidate[i].count);      }      printf("Wrong election:%d\n", wrong);      return 0;  } |

Q1341.(10分)计算并输出1到20的平方，当结果大于100时停止。

\*\*输出格式要求："%d \* %d = %d\n"

程序的运行示例如下：

1\*1=1

2\*2=4

3\*3=9

4\*4=16

5\*5=25

6\*6=36

7\*7=49

8\*8=64

9\*9=81

10\*10=100

|  |
| --- |
| #include <stdio.h>    int main()  {      int i, result;      for (i = 1; i <= 20; i++)      {          result = i \* i;          if (result > 100)              break;          else              printf("%d\*%d=%d\n", i, i, i \* i);      }  } |

AddGH

Q1699.(10分)根据最大公约数的如下3条性质，采用迭代法编写计算最大公约数的函数Gcd()，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最大公约数。

性质1 如果a>b，则a和b与a-b和b的最大公约数相同，即Gcd(a, b) = Gcd(a-b, b)

性质2 如果b>a，则a和b与a和b-a的最大公约数相同，即Gcd(a, b) = Gcd(a, b-a)

性质3 如果a=b，则a和b的最大公约数与a值和b值相同，即Gcd(a, b) = a = b

要求如下：

（1）从键盘任意输入的两整数

主函数调用Gcd()函数，并输出两整数的最大公约数。

（2）Gcd函数原型为：

int Gcd(int a, int b);

如果输入的数不是正整数，则返回-1，否则，返回两个数的最大公约数。

（3）\*\*输入提示信息格式要求："Input a,b:\n"

输入两个整数时用,号分隔

\*\*输出提示信息要求：

如果输入的数不是正整数，则输出"Input error!\n"，否则按如下格式输出"Gcd=%d\n"

#include <stdio.h>

int Gcd(int a, int b);

int main()

{

int a, b, c;

printf("Input a,b:");

scanf("%d,%d", &a, &b);

c = Gcd(a, b);

if (c != -1)

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c);

else

printf("Input number should be positive!\n");

return 0;

}

int Gcd(int a, int b)

{

if (a <= 0 || b <= 0) return -1;

while (a != b)

{

if (a > b)

a = a - b;

else if (b > a)

b = b - a;

}

return a;

}

Q241.(10分)在一种室内互动游戏中，魔术师要每位观众心里想一个三位数abc（a、b、c分别是百位、十位和个位数字），然后魔术师让观众心中记下acb、bac、bca、cab、cba五个数以及这5个数的和值。只要观众说出这个和是多少，则魔术师一定能猜出观众心里想的原数abc是多少。例如，观众甲说他计算的和值是1999，则魔术师立即说出他想的数是443，而观众乙说他计算的和值是1998，则魔术师说：“你算错了！”。请编程模拟这个数字魔术游戏。

\*\*输入格式要求："%d" 提示信息："Input a sum:"

\*\*输出格式要求："The sum you calculated is wrong!\n" "The number is %d\n"

程序运行示例如下：

Input a sum:1999

The number is 443

#include <stdio.h>

int Magic(int m);

int main()

{

int m, ret;

printf("Input a sum:");

scanf("%d", &m);

ret = Magic(m);

if (ret != 1)

{

printf("The sum you calculated is wrong!\n");

}

return 0;

}

int Magic(int m)

{

int a, b, c, n;

for (a = 1; a < 9; a++)

{

for (b = 1; b < 9; b++)

{

for (c = 1; c < 9; c++)

{

n = 122 \* a + 212 \* b + 221 \* c;

if (m == n)

{

printf("The number is %d\n", 100 \* a + 10 \* b + c);

return 1;

}

}

}

}

return 0;

}

Q238.(10分)利用求阶乘函数Fact()，编程计算并输出1! + 2! + …… + n!的值。

\*\*输入格式要求："%u" 提示信息："Input n(n>0):"

\*\*输出格式要求："sum = %lu\n

#include <stdio.h>

unsigned long Fact(unsigned int n);

int main()

{

unsigned int i, n;

unsigned long sum = 0;

printf("Input n(n>0):");

scanf("%u", &n);

for (i = 1; i <= n; i++)

{

sum = sum + Fact(i);

}

printf("sum = %lu\n", sum);

return 0;

}

/\* 函数功能：用迭代法计算无符号整型变量n的阶乘 \*/

unsigned long Fact(unsigned int n)

{

unsigned int i;

unsigned long result = 1;

for (i = 2; i <= n; i++)

result \*= i;

return result;

}

Q536.(10分)利用静态变量计算n的阶乘。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："%d! = %ld\n"

程序运行示例如下：

Input n:10

1! = 1

2! = 2

3! = 6

4! = 24

5! = 120

6! = 720

7! = 5040

8! = 40320

9! = 362880

10! = 3628800

#include <stdio.h>

long Func(int n);

int main()

{

int i, n;

printf("Input n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

printf("%d! = %ld\n", i, Func(i));

}

return 0;

}

long Func(int n)

{

static long p = 1; /\*定义静态变量\*/

p = p \* n;

return p;

}

Q534.(10分)编程求解汉诺塔问题。

汉诺塔（Hanoi）是必须用递归方法才能解决的经典问题。它来自于印度神话。上帝创造世界时作了三根金刚石柱子，在第一根柱子上从下往上按大小顺序摞着64片黄金圆盘，如图7-3所示。上帝命令婆罗门把圆盘从下面开始按大小顺序重新摆放到第二根柱子上，并且规定，每次只能移动一个圆盘，在小圆盘上不能放大圆盘。有人预言说，这件事完成时宇宙会在一瞬间闪电式毁灭，也有人相信婆罗门至今仍在一刻不停地搬动着圆盘。

\*\*输入格式要求："%d" 提示信息："Input the number of disks:"

\*\*输出格式要求："Steps of moving %d disks from A to B by means of C:\n" "Move %d: from %c to %c\n"

程序运行示例如下：

Input the number of disks:3

Steps of moving 3 disks from A to B by means of C:

Move 1: from A to B

Move 2: from A to C

Move 1: from B to C

Move 3: from A to B

Move 1: from C to A

Move 2: from C to B

Move 1: from A to B

#include <stdio.h>

void Hanoi(int n, char a, char b, char c);

void Move(int n, char a, char b);

int main()

{

int n;

printf("Input the number of disks:");

scanf("%d", &n);

printf("Steps of moving %d disks from A to B by means of C:\n", n);

Hanoi(n, 'A', 'B', 'C'); /\*调用递归函数Hanoi()将n个圆盘借助于C由A移动到B\*/

return 0;

}

/\* 函数功能：用递归方法将n个圆盘借助于柱子c从源柱子a移动到目标柱子b上 \*/

void Hanoi(int n, char a, char b, char c)

{

if (n == 1)

{

Move(n, a, b); /\* 将第n个圆盘由a移到b \*/

}

else

{

Hanoi(n - 1, a, c, b); /\* 递归调用Hanoi()，将第n-1个圆盘借助于b由a移动到c\*/

Move(n, a, b); /\* 第n个圆盘由a移到b \*/

Hanoi(n - 1, c, b, a); /\*递归调用Hanoi()，将第n-1个圆盘借助于a由c移动到b\*/

}

}

/\* 函数功能： 将第n个圆盘从源柱子a移到目标柱子b上 \*/

void Move(int n, char a, char b)

{

printf("Move %d: from %c to %c\n", n, a, b);

}

Q251.(10分)冒泡排序（Bubble Sort），也称为沉降排序（Sinking Sort），之所以称其为冒泡排序，是因为算法中值相对较小的数据会像水中的气泡一样逐渐上升到数组的最顶端。与此同时，较大的数据逐渐地下沉到数组的底部。这个处理过程需在整个数组范围内反复执行多遍。每一遍执行时，比较相邻的两个元素。若顺序不对，则将其位置交换，当没有数据需要交换时，数据也就排好序了。编程将排序函数DataSort()改用冒泡法实现。

\*\*输入格式要求："%d" 提示信息："Input n:" "Input %d numbers:"

\*\*输出格式要求："Sorting results:" "%4d"

程序运行示例如下：

Input n:10

Input 10 numbers:10 2 9 3 4 0 6 8 7 5 1

Sorting results: 0 1 2 3 4 5 6 7 8 9

#include<stdio.h>

#define N 10

void BubbleSort(int a[], int n);

int main()

{

int i, n, a[N];

printf("Input n:");

scanf("%d", &n);

printf("Input %d numbers:", n);

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]);

}

BubbleSort(a, n);

printf("Sorting results:");

for (i = 0; i < n; i++)

{

printf("%4d", a[i]);

}

printf("\n");

return 0;

}

/\* 函数功能：冒泡法实现数组a 的n个元素的升序排序 \*/

void BubbleSort(int a[], int n)

{

int i, j, temp;

for (i = 0; i < n - 1; i++)

{

for (j = 1; j < n - i; j++)

{

if (a[j] < a[j - 1])

{

temp = a[j];

a[j] = a[j - 1];

a[j - 1] = temp;

}

}

}

}

Q538.(10分)从键盘输入某年某月（包括闰年），编程输出该年的该月拥有的天数。

\*\*输入格式要求："%d,%d" 提示信息："Input year,month:" "The number of days is %d\n"

\*\*输出格式要求："%d"

程序运行示例1如下：

Input year,month:1984,2

The number of days is 29

程序运行示例2如下：

Input year,month:2000,2

The number of days is 29

程序运行示例3如下：

Input year,month:1985,2

The number of days is 28

程序运行示例4如下：

Input year,month:1983,13

Input year,month:1983,-1

Input year,month:1983,1

The number of days is 31

#include <stdio.h>

#define  MONTHS 12

int main()

{

    int days[2][MONTHS] = {{31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},                                      {31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

    int year, month;

    do

    {

        printf("Input year,month:");

        scanf("%d,%d", &year, &month);

    }

    while (month < 1 || month > 12);   /\* 处理不合法数据的输入 \*/

    if (((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0)) /\*闰年\*/

        printf("The number of days is %d\n", days[1][month - 1]);

    else  /\*非闰年\*/

        printf("The number of days is %d\n", days[0][month - 1]);

    return 0;

}

Q542.(10分)从键盘输入某班学生某门课的成绩（每班人数最多不超过40人），当输入为负值时，表示输入结束，试编程将分数按从高到低顺序进行排序输出。用函数编程实现排序功能。

\*\*输入格式要求："%d" 提示信息："Total students are %d\n" "Sorted scores:" "Input score:"

\*\*输出格式要求："%4d"

程序的运行示例如下：

Input score:84

Input score:83

Input score:88

Input score:87

Input score:61

Input score:-1

Total students are 5

Sorted scores: 88 87 84 83 61

#include <stdio.h>

#define N 40

int ReadScore(int score[]); /\* ReadScore()函数原型 \*/

void DataSort(int score[], int n); /\* DataSort()原函数型 \*/

void PrintScore(int score[], int n); /\* PrintScore()函数原型 \*/

int main()

{

int score[N], n;

n = ReadScore(score); /\*调用函数ReadScore()输入成绩，返回学生人数\*/

printf("Total students are %d\n", n);

DataSort(score, n); /\* 调用函数DataSort()进行成绩排序 \*/

printf("Sorted scores:");

PrintScore(score, n); /\* 调用函数Printscore()输出成绩排序结果 \*/

return 0;

}

/\* 函数功能：输入学生某门课的成绩，当输入负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[]) /\* ReadScore()函数定义 \*/

{

int i = -1; /\* i初始化为-1，可保证循环体内i增1后数组下标从0开始 \*/

do

{

i++;

printf("Input score:");

scanf("%d", &score[i]);

}

while (score[i] >= 0); /\* 输入负值时结束成绩输入 \*/

return i; /\* 返回学生人数 \*/

}

/\* 函数功能：按交换法将数组score的元素值元素按从高到低排序 \*/

void DataSort(int score[], int n) /\* DataSort()函数定义 \*/

{

int i, j, temp;

for (i = 0; i < n - 1; i++)

{

for (j = i + 1; j < n; j++)

{

if (score[j] > score[i]) /\* 按数组score的元素值从高到低排序 \*/

{

temp = score[j];

score[j] = score[i];

score[i] = temp;

}

}

}

}

/\* 函数功能： 打印学生成绩 \*/

void PrintScore(int score[], int n) /\* PrintScore()函数定义 \*/

{

int i;

for (i = 0; i < n; i++)

{

printf("%4d", score[i]);

}

printf("\n");

}

Q1596.(10分)从键盘输入10个整数，编程计算并输出其最大值、最小值及其所在元素的下标位置。

输入格式："%d"

输出格式："max=%d, pos=%d\n"

"min=%d, pos=%d\n"

输入样例：

1 2 3 4 5 6 7 8 9 10

输出样例：

max=10, pos=9

min=1, pos=0

#include <stdio.h>

int main()

{

int a[10], n, max, min, maxPos, minPos;

for (n = 0; n < 10; n++)

{

scanf("%d", &a[n]);

}

max = min = a[0];

maxPos = minPos = 0;

for (n = 0; n < 10; n++)

{

if (a[n] > max)

{

max = a[n];

maxPos = n;

}

else if (a[n] < min)

{

min = a[n];

minPos = n;

}

}

printf("max=%d, pos=%d\n", max, maxPos);

printf("min=%d, pos=%d\n", min, minPos);

return 0;

}

Q398.(10分)梯形法。用梯形法编程求函数f(a)=x 2 +2x+1 的定积分，∫ 2 0 f(x)dx 的值。

\*\*输入格式要求："%lf,%lf,%lf" 提示信息："Enter n,a,b:"

\*\*输出格式要求："Sum=%lf\n"

程序运行示例如下：

Enter n,a,b:10,0,2

Sum=10.604000

#include "stdio.h"

main()

{ int i;

double s=0,h,f0,f1,n,a,b;

printf("Enter n,a,b:");

scanf("%lf,%lf,%lf",&n,&a,&b);

h=(b-a)/n;

f0=a\*a+2\*a+1;

for(i=0;i<=n;i++)

{ a=a+h;

f1=a\*a+2\*a+1;

s+=(f0+f1)\*h/2;

f0=f1;

}

printf("Sum=%lf\n",s);

}

Q1683.(10分)用二维数组作为函数参数，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int a[][N], int at[][M], int m, int n);

void InputMatrix(int a[][N], int m, int n);

void PrintMatrix(int at[][M], int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输出提示信息和格式："The transposed matrix is:\n"

#include <stdio.h>

#define M 10

#define N 10

void Transpose(int a[][N], int at[][M], int m, int n);

void InputMatrix(int a[][N], int m, int n);

void PrintMatrix(int at[][M], int n, int m);

int main()

{

int s[M][N], st[N][M], m, n;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(s, m, n);

Transpose(s, st, m, n);

printf("The transposed matrix is:\n");

PrintMatrix(st, n, m);

return 0;

}

/\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/

void Transpose(int a[][N], int at[][M], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

at[j][i] = a[i][j];

}

}

}

/\* 函数功能：输入m\*n矩阵a的值 \*/

void InputMatrix(int a[][N], int m, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i][j]);

}

}

}

/\* 函数功能：输出n\*m矩阵at的值 \*/

void PrintMatrix(int at[][M], int n, int m)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

printf("%d\t", at[i][j]);

}

printf("\n");

}

}

Q248.(10分)兔子生崽问题。假设一对小兔的成熟期是一个月，即一个月可长成成兔，那么如果每对成兔每个月都可以生一对小兔，一对新生的小兔从第二个月起就开始生兔子，试问从一对兔子开始繁殖，一年以后可有多少对兔子？请编程求解该问题。

\*\*输出格式要求："\nTotal = %d\n" "%4d"

程序运行示例如下：

1 2 3 5 8 13 21 34 55 89 144 233

Total = 233

#include <stdio.h>

#define N 12

void Fibonacci(int f[], int n);

main()

{

int f[N], i;

Fibonacci(f, N);

printf("\nTotal = %d\n", f[N - 1]);

}

/\* 函数功能：计算并打印Fibonacci数列的前n项 \*/

void Fibonacci(int f[], int n)

{

int i;

f[0] = 1;

f[1] = 2;

for (i = 2; i < n; i++)

{

f[i] = f[i - 1] + f[i - 2];

}

for (i = 0; i < N; i++)

{

printf("%4d", f[i]);

}

}

Q1681.(10分)按如下函数原型编程计算并输出n×n阶矩阵的转置矩阵。其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int (\*a)[N], int n);

void Swap(int \*x, int \*y);

void InputMatrix(int (\*a)[N], int n);

void PrintMatrix(int (\*a)[N], int n);

输入提示信息："Input n:"

输入格式："%d"

输出提示信息和格式："The transposed matrix is:\n"

参考答案

#include <stdio.h>

#define N 10

void Swap(int \*x, int \*y);

void Transpose(int (\*a)[N], int n);

void InputMatrix(int (\*a)[N], int n);

void PrintMatrix(int (\*a)[N], int n);

int main()

{

int s[N][N], n;

printf("Input n:");

scanf("%d", &n);

InputMatrix(s, n);

Transpose(s, n);

printf("The transposed matrix is:\n");

PrintMatrix(s, n);

return 0;

}

/\* 函数功能：交换两个整型数的值 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* 函数功能：计算n\*n矩阵的转置矩阵 \*/

void Transpose(int (\*a)[N], int n)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = i; j < n; j++)

{

Swap(\*(a + i) + j, \*(a + j) + i);

}

}

}

/\* 函数功能：输入n\*n矩阵的值 \*/

void InputMatrix(int (\*a)[N], int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", n, n);

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", \*(a + i) + j);

}

}

}

/\* 函数功能：输出n\*n矩阵的值 \*/

void PrintMatrix(int (\*a)[N], int n)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

printf("%d\t", \*(\*(a + i) + j));

}

printf("\n");

}

}

Q1645.(10分)按如下函数原型编程从键盘输入一个m行n列的二维数组，然后计算数组中元素的最大值及其所在的行列下标值。其中，m和n的值由用户键盘输入。已知m和n的值都不超过10。

void InputArray(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);

输入提示信息：

"Input m, n:"

"Input %d\*%d array:\n"

输入格式："%d,%d"

输出提示信息和格式："max = %d, row = %d, col = %d\n"

参考答案

#include <stdio.h>

#define M 10

#define N 10

void InputMatrix(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);

int main()

{

int a[M][N], m, n, row, col, max;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(\*a, m, n);

max = FindMax(\*a, m, n, &row, &col);

printf("max = %d, row = %d, col = %d\n", max, row, col);

return 0;

}

/\* 函数功能：输入m\*n矩阵的值 \*/

void InputMatrix(int \*p, int m, int n)

{

int i, j;

printf("Input %d\*%d array:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &p[i \* n + j]);

}

}

}

/\* 函数功能：在m\*n矩阵中查找最大值及其所在的行列号 \*/

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol)

{

int i, j, max = p[0];

\*pRow = 0;

\*pCol = 0;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

if (p[i \* n + j] > max)

{

max = p[i \* n + j];

\*pRow = i; /\*记录行下标\*/

\*pCol = j; /\*记录列下标\*/

}

}

}

return max;

}

Q1685.(10分)用二维数组的列指针作为函数实参，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int \*a, int \*at, int m, int n);

void InputMatrix(int \*a, int m, int n);

void PrintMatrix(int \*at, int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输出提示信息和格式："The transposed matrix is:\n"

参考答案

#include <stdio.h>

#define M 10

#define N 10

void Transpose(int \*a, int \*at, int m, int n);

void InputMatrix(int \*a, int m, int n);

void PrintMatrix(int \*at, int n, int m);

int main()

{

int s[M][N], st[N][M], m, n;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(\*s, m, n);

Transpose(\*s, \*st, m, n);

printf("The transposed matrix is:\n");

PrintMatrix(\*st, n, m);

return 0;

}

/\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/

void Transpose(int \*a, int \*at, int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

at[j \* m + i] = a[i \* n + j];

}

}

}

/\* 函数功能：输入m\*n矩阵a的值 \*/

void InputMatrix(int \*a, int m, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i \* n + j]);

}

}

}

/\* 函数功能：输出n\*m矩阵at的值 \*/

void PrintMatrix(int \*at, int n, int m)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

printf("%d\t", at[i \* m + j]);

}

printf("\n");

}

}

Q258.(30分)用字符数组作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

#include <stdio.h>

void Squeeze(char s[], char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char s[], char c)

{

int i, j;

for (i = j = 0; s[i] != '\0'; i++)

{

if (s[i] != c)

{

s[j] = s[i];

j++;

}

}

s[j] = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

Q1638.(35分)按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char \*dstStr, char \*srcStr);

将srcStr字符串连接到dstStr字符串后面，；

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

参考答案

#include <stdio.h>

#include <string.h>

#define N 80

void MyStrcat(char \*dstStr, char \*srcStr);

int main()

{

char s[N], t[N];

printf("Input a string:");

gets(s);

printf("Input another string:");

gets(t);

MyStrcat(s, t);

printf("Concatenate results:%s\n", s);

return 0;

}

/\* 函数功能：将字符串srcStr连接到字符串dstStr之后 \*/

void MyStrcat(char \*dstStr, char \* srcStr)

{

while (\*dstStr != '\0')

{

dstStr++;

}

while (\*srcStr != '\0')/\* 若srcStr所指字符不是字符串结束标志 \*/

{

\*dstStr = \*srcStr; /\* 将srcStr所指字符复制到dstStr所指内存中\*/

srcStr++; /\* 使srcStr指向下一个字符 \*/

dstStr++; /\* 使dstStr指向下一个存储单元 \*/

}

\*dstStr = '\0'; /\* 在字符串dstStr的末尾添加字符串结束标志 \*/

}

Q1686.(10分)用字符指针作函数参数编程实现在字符串每个字符间插入一个空格的功能。

要求：字符串的输入和输出在主函数中完成。

\*\*输入提示信息："Input a string:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Insert results:%s\n"

程序运行示例1如下：

Input a string:Howareyou

Insert results:H o w a r e y o u

参考答案

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#define N 100

void Insert(char \*s);

int main()

{

char str[N];

printf("Input a string:");

gets(str);

Insert(str);

printf("Insert results:%s\n", str);

return 0;

}

void Insert(char \*s)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; s++, t++)

{

\*s = \*t;

s++;

\*s = ' ';

}

\*s = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

Q1636.(10分)按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char dstStr[], char srcStr[]);

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void MyStrcat(char dstStr[], char srcStr[]);

int main()

{

char s[N], t[N];

printf("Input a string:");

gets(s);

printf("Input another string:");

gets(t);

MyStrcat(s, t);

printf("Concatenate results:%s\n", s);

return 0;

}

/\* 函数功能：将字符串srcStr连接到字符串dstStr之后 \*/

void MyStrcat(char dstStr[], char srcStr[])

{

unsigned int i, j;

i = strlen(dstStr); /\* 将下标移动到字符串dstStr的末尾 \*/

for (j = 0; j <= strlen(srcStr); j++, i++)

{

dstStr[i] = srcStr[j];

}

}

Q548.(15分)从键盘任意输入一个字符串，计算其实际字符个数并打印输出，即不使用字符串处理函数strlen(),编程实现strlen()的功能。

\*\*输入格式要求："%s" 提示信息："Please enter a string:"

\*\*输出格式要求："The length of the string is: %u\n"

程序的运行示例如下：

Please enter a string:Hello China

The length of the string is: 11

#include <stdio.h>

unsigned int MyStrlen(char str[]);

/\* 函数功能：用字符型数组作函数参数，计算字符串的长度 \*/

unsigned int MyStrlen(char str[])

{

int i ;

unsigned int len = 0; /\* 计数器置0 \*/

for (i = 0; str[i] != '\0'; i++)

{

len++; /\* 利用循环统计不包括’\0’在内的字符个数 \*/

}

return (len); /\* 返回实际字符个数 \*/

}

int main()

{

char a[80];

unsigned int len;

printf("Please enter a string:");

gets(a);

len = MyStrlen(a); /\* 计算字符串实际字符个数 \*/

printf("The length of the string is: %u\n", len);

return 0;

}

Q1632.(10分)按如下函数原型编写程序，用字符数组作函数参数编程实现字符串逆序存放功能。

void Inverse(char str[], char ptr[]);

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[], char ptr[]);

int main()

{

char a[N], b[N];

printf("Input a string:");

gets(a);

Inverse(a, b);

printf("Inversed results:%s\n", b);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char str[], char ptr[])

{

int len, i, j;

len = strlen(str);

for (i = 0, j = len - 1; str[i] != '\0'; i++, j--)

{

ptr[j] = str[i];

}

ptr[i] = '\0'; /\* 在字符串ptr的末尾添加字符串结束标志 \*/

}

Q1687.(10分)用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

#include <stdio.h>

#include <string.h>

#define N 100

void Squeeze(char \*s, char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char \*s, char c)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; t++)

{

if (\*t != c)

{

\*s = \*t;

s++;

}

}

\*s = '\0'; /\* 在字符串t2的末尾添加字符串结束标志 \*/

}

Q257.(10分)输入一行字符，用函数编程统计其中有多少单词。假设单词之间以空格分开。

\*\*输入格式要求：提示信息："Input a string:"

\*\*输出格式要求："Numbers of words = %d\n"

#include <stdio.h>

int CountWords(char str[]);

int main()

{

char str[20];

printf("Input a string:");

gets(str);

printf("Numbers of words = %d\n", CountWords(str));

return 0;

}

int CountWords(char str[])

{

int i, num;

num = (str[0] != ' ') ? 1 : 0;

for (i = 1; str[i] != '\0'; i++)

{

if (str[i] != ' ' && str[i - 1] == ' ')

{

num++;

}

}

return num;

}

Q1626.(10分)用字符指针作函数参数编程实现字符串逆序存放功能。

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char \*pStr);

int main()

{

char a[N];

printf("Input a string:");

gets(a);

Inverse(a);

printf("Inversed results:%s\n", a);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char \*pStr)

{

int len;

char temp;

char \*pStart; /\* 指针变量pStart指向字符串的第一个字符 \*/

char \*pEnd; /\* 指针变量pEnd指向字符串的最后一个字符 \*/

len = strlen(pStr); /\* 求出字符串长度 \*/

for (pStart = pStr, pEnd = pStr + len - 1; pStart < pEnd; pStart++, pEnd--)

{

temp = \*pStart;

\*pStart = \*pEnd;

\*pEnd = temp;

}

}

Q1633.(10分)按如下函数原型编写程序，用字符数组作函数参数编程实现字符串逆序存放功能。

void Inverse(char str[]);

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

int main()

{

char a[N];

printf("Input a string:");

gets(a);

Inverse(a);

printf("Inversed results:%s\n", a);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char str[])

{

int len, i, j;

char temp;

len = strlen(str);

for (i = 0, j = len - 1; i < j; i++, j--)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

}

}

Q606.(10分)任意输入英文的星期几，通过查找星期表，输出其对应的数字，

若查到表尾，仍未找到，则输出错误提示信息。

\*\*输入格式要求："%s" 提示信息："Please enter a string:\n"

\*\*输出格式要求："%s is %d\n" "Not found!\n"

查找表中信息：

"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"

程序运行示例1如下：

Please enter a string:

Monday

Monday is 1

程序运行示例2如下：

Monkey

Not found!

#include <stdio.h>

#include <string.h>

#define WEEKDAYS 7

#define MAX\_STR\_LEN 10

int main()

{

int i, pos;

int findFlag = 0;

char x[MAX\_STR\_LEN];

char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};

printf("Please enter a string:\n");

scanf("%s", x);

for (i = 0; i < WEEKDAYS && !findFlag; i++)

{

if (strcmp(x, weekDay[i]) == 0)

{

pos = i;

findFlag = 1;

}

}

if (findFlag)

{

printf("%s is %d\n", x, pos);

}

else

{

printf("Not found!\n");

}

return 0;

}

Q196.(25分)利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

Q546.(10分)请编程实现按奥运会参赛国国名在字典中的顺序对其入场次序进行排序。假设参赛国不超过150个。要求通过二维字符数组传参调用函数实现排序,输入和输出在主函数中实现。

\*\*输入格式要求："%d" 提示信息："How many countries?" "Input their names:\n"

\*\*输出格式要求："Sorted results:\n"

程序的运行示例如下：

How many countries? 5

Input their names:

America

England

Australia

Sweden

Finland

Sorted results:

America

Australia

England

Finland

Sweden

#include <stdio.h>

#include <string.h>

#define MAX\_LEN 10 /\* 字符串最大长度 \*/

#define N 150 /\* 字符串个数 \*/

void SortString(char str[][MAX\_LEN], int n);

int main()

{

int i, n;

char name[N][MAX\_LEN]; /\* 定义二维字符数组 \*/

printf("How many countries?");

scanf("%d", &n);

getchar(); /\* 读走输入缓冲区中的回车符 \*/

printf("Input their names:\n");

for (i = 0; i < n; i++)

{

gets(name[i]); /\* 输入n个字符串 \*/

}

SortString(name, n); /\* 字符串按字典顺序排序 \*/

printf("Sorted results:\n");

for (i = 0; i < n; i++)

{

puts(name[i]); /\* 输出排序后的n个字符串 \*/

}

return 0;

}

/\* 函数功能：交换法实现字符串按字典顺序排序 \*/

void SortString(char str[][MAX\_LEN], int n)

{

int i, j;

char temp[MAX\_LEN];

for (i = 0; i < n - 1; i++)

{

for (j = i + 1; j < n; j++)

{

if (strcmp(str[j], str[i]) < 0)

{

strcpy(temp, str[i]);

strcpy(str[i], str[j]);

strcpy(str[j], temp);

}

}

}

}

Q259.(10分)用字符数组作函数参数编程实现在字符串每个字符间插入一个空格的功能。

\*\*输入提示信息："Input a string:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Insert results:%s\n"

程序运行示例1如下：

Input a string:Howareyou

Insert results:H o w a r e y o u

#include <stdio.h>

#include <string.h>

#define N 100

void Insert(char s[]);

main()

{

char str[N];

printf("Input a string:");

gets(str);

Insert(str);

printf("Insert results:%s\n", str);

}

void Insert(char s[])

{

char t[N];

int i, j;

strcpy(t, s);

for (i = 0, j = 0; t[i] != '\0'; i++, j++)

{

s[j] = t[i];

j++;

s[j] = ' ';

}

s[j] = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

Q1624.(10分)用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*输入格式要求："%s"

输入提示信息：

"Input a string:"

"Input a character:"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

#include <stdio.h>

#include <string.h>

#define N 100

void Squeeze(char \*s, char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char \*s, char c)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; t++)

{

if (\*t != c)

{

\*s = \*t;

s++;

}

}

\*s = '\0'; /\* 在字符串t2的末尾添加字符串结束标志 \*/

}

Q549.(10分)不使用字符串处理函数strcat()编程实现strcat()的功能，即任意输入两个字符串，然后连接这两个字符串，返回连接后字符串的首地址。

程序的运行示例如下：

Input the first string:Hello

Input the second string:China

The result is: HelloChina

#include <stdio.h>

#define N 80

char \*MyStrcat(char \*dstStr, char \*srcStr);

int main()

{

char first[2 \* N]; /\* 这个数组应该足够大，以便存放连接后的字符串 \*/

char second[N];

char \*result = NULL;

printf("Input the first string:");

gets(first);

printf("Input the second string:");

gets(second);

result = MyStrcat(first, second);

printf("The result is: %s\n", result);

return 0;

}

/\* 函数功能：连接两个字符串 \*/

char \*MyStrcat(char \*dstStr, char \*srcStr)

{

char \*pStr = dstStr; /\* 保存字符串首地址指针 \*/

/\* 将指针移到字符串末尾 \*/

while (\*dstStr != '\0')

{

dstStr++;

}

/\* 将字符串2内容复制到字符串1的后面 \*/

for (; \*srcStr != '\0'; dstStr++, srcStr++)

{

\*dstStr = \*srcStr;

}

\*dstStr = '\0'; /\* 在连接后的字符串的末尾添加字符串结束标志 \*/

return pStr; /\* 返回连接后的字符串的指针 \*/

}

Q1690.(10分)按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

/\* 函数功能： 对给定的某年某月某日，计算它是这一年的第几天

函数参数： 整型变量year、month、day，分别代表年、月、日

函数返回值：这一年的第几天 \*/

int DayofYear(int year, int month, int day);

/\* 函数功能： 对给定的某一年的第几天，计算它是这一年的第几月第几日

函数入口参数：整型变量year，存储年

整型变量yearDay，存储这一年的第几天

函数出口参数：整型指针pMonth，指向存储这一年第几月的整型变量

整型指针pDay，指向存储第几日的整型变量

函数返回值： 无 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

输出如下菜单，用switch语句实现根据用户输入的选择执行相应的操作。

1. year/month/day -> yearDay

2. yearDay -> year/month/day

3. Exit

Please enter your choice:

输入提示信息：

"Please enter year, month, day:"

"Please enter year, yearDay:"

输入格式：

"%d,%d,%d"

"%d,%d"

输出提示信息和格式：

"yearDay = %d\n"

"month = %d, day = %d\n"

#include <stdio.h>

#include <stdlib.h>

int DayofYear(int year, int month, int day);

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

void Menu(void);

int dayTab[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}, {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

int main()

{

int year, month, day, yearDay;

char c;

Menu(); /\* 调用Menu函数显示一个固定式菜单 \*/

c = getchar(); /\* 输入选择 \*/

switch (c) /\* 判断选择的是何种操作 \*/

{

case '1':

printf("Please enter year, month, day:");

scanf("%d,%d,%d", &year, &month, &day);

yearDay = DayofYear(year, month, day);

printf("yearDay = %d\n", yearDay);

break;

case '2':

printf("Please enter year, yearDay:");

scanf("%d,%d", &year, &yearDay);

MonthDay(year, yearDay, &month, &day);

printf("month = %d,day = %d\n", month, day);

break;

case '3':

exit(0); /\* 退出程序的运行 \*/

default:

printf("Input error!");

}

return 0;

}

/\* 函数功能：对给定的某年某月某日，计算并返回它是这一年的第几天 \*/

int DayofYear(int year, int month, int day)

{

int i, leap;

/\* 若year为闰年，即leap值为1，则用第1行元素dayTab[1][i]计算；

否则leap值为0，用第0行dayTab[0][i]计算 \*/

leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

for (i = 1; i < month; i++)

{

day = day + dayTab[leap][i];

}

return day; /\* 返回计算出的day的值 \*/

}

/\* 函数功能：对给定的某一年的第几天，计算它是这一年的第几月第几日 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay)

{

int i, leap;

leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

for (i = 1; yearDay > dayTab[leap][i]; i++)

{

yearDay = yearDay - dayTab[leap][i];

}

\*pMonth = i; /\* 将计算出的月份值赋值给pMonth所指向的变量 \*/

\*pDay = yearDay; /\* 将计算出的日号赋值给pDay所指向的变量 \*/

}

/\* 函数功能：显示菜单 \*/

void Menu(void)

{

printf("1. year/month/day -> yearDay\n");

printf("2. yearDay -> year/month/day\n");

printf("3. Exit\n");

printf("Please enter your choice:");

}

Q384.(30分)求复数之积。利用结构变量求解两个复数之积。

za={3,4}, zb={5,6}

za={10,20}, zb={30,40}

\*\*输出格式要求："(%d+%di)\*(%d+%di)=" "(%d+%di)\n"

程序运行示例如下：

(3+4i)\*(5+6i)=(-9+38i)

(10+20i)\*(30+40i)=(-500+1000i)

#include "stdio.h"

struct complx

{

int real;

int im;

};

main()

{

static struct complx za={3,4};

static struct complx zb={5,6};

struct complx x,y,z;

struct complx cmult();

void cpr();

z=cmult(za,zb);

cpr(za,zb,z);

x.real = 10; x.im = 20;

y.real = 30; y.im = 40;

z=cmult(x,y);

cpr(x,y,z);

}

struct complx cmult (za,zb)

struct complx za,zb;

{

struct complx w;

w.real = za.real\*zb.real - za.im\*zb.im;

w.im = za.real\*zb.im+za.im\*zb.real;

return (w);

}

void cpr(za,zb,z)

struct complx za,zb,z;

{

printf("(%d+%di)\*(%d+%di)=",za.real,za.im,zb.real,zb.im);

printf("(%d+%di)\n",z.real,z.im);

}

Q1330.(50分)13个人围成一圈，从第1个人开始顺序报号1、2、3，凡报到3的人退出圈子。找出最后留在圈子里的人原来的序号。要求用结构体编程实现。

程序的运行示例如下：

出圈成员及顺序： 3 6 9 12 2 7 11 4 10 5 1 8

最后的成员是： 13

#include <stdio.h>

#define N 13

struct person

{

int number;

int nextp;

} link[N + 1];

int main(void)

{

int i, count, h;

/\* 建立队列 \*/

for (i = 1; i <= N; i++)

{

if (i == N)

{

link[i].nextp = 1;

}

else

{

link[i].nextp = i + 1;

}

link[i].number = i;

}

printf("\n");

count = 0;

h = N;

printf("出圈成员及顺序：");

while (count < N - 1)

{

i = 0;

while (i != 3)

{

h = link[h].nextp;

if (link[h].number)

{

i++;

}

}

printf("%3d", link[h].number);

link[h].number = 0;

count++;

}

printf("\n最后的成员是：");

for (i = 1; i <= N; i++)

{

if (link[i].number)

{

printf("%3d", link[i].number);

}

}

return 0;

}

Q268.(30分)编程统计候选人的得票数。设有3个候选人zhang、li、wang，10个选民，选民每次输入一个得票的候选人的名字，若选民输错候选人姓名，则按废票处理。选民投票结束后程序自动显示各候选人的得票结果和废票信息。要求用结构体数组candidate表示3个候选人的姓名和得票结果。

#include <stdio.h>

#include <string.h>

#define NUM\_ELECTORATE 10

#define NUM\_CANDIDATE 3

struct candidate

{

char name[20];

int count;

} candidate[3] = {"li", 0, "zhang", 0, "wang", 0};

int main()

{

int i, j, flag = 1, wrong = 0;

char name[20];

for (i = 1; i <= NUM\_ELECTORATE; i++)

{

printf("Input vote %d:", i);

scanf("%s", name);

flag = 1;

for (j = 0; j < NUM\_CANDIDATE; j++)

{

if (strcmp(name, candidate[j].name) == 0)

{

candidate[j].count++;

flag = 0;

}

}

if (flag)

{

wrong++; /\* 废票计数 \*/

flag = 0;

}

}

printf("Election results:\n");

for (i = 0; i < NUM\_CANDIDATE; i++)

{

printf("%8s:%d\n", candidate[i].name, candidate[i].count);

}

printf("Wrong election:%d\n", wrong);

return 0;

}

Q269.(25分)编程模拟洗牌和发牌过程。一副扑克有52张牌，分为4种花色（suit）：黑桃（Spades）、红桃（Hearts）、草花（Clubs）、方块（Diamonds）。每种花色又有13张牌面（face）：A，2，3，4，5，6，7，8，9，10，Jack，Queen，King。要求用结构体数组card表示52张牌，每张牌包括花色和牌面两个字符型数组类型的数据成员。

#include <stdio.h>

#include <string.h>

#include <time.h>

#include <stdlib.h>

typedef struct card

{

char suit[10];

char face[10];

} CARD;

void Deal(CARD \*wCard);

void Shuffle(CARD \*wCard);

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[]);

int main()

{

char \*suit[] = {"Spades", "Hearts", "Clubs", "Diamonds"};

char \*face[] = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10",

"Jack", "Queen", "King"

};

CARD card[52];

srand (time(NULL));

FillCard(card, face, suit);

Shuffle(card);

Deal(card);

return 0;

}

/\* 函数功能：花色按黑桃、红桃、草花、方块的顺序，面值按A~K的顺序，排列52张牌 \*/

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[])

{

int i;

for (i = 0; i < 52; i++)

{

strcpy(wCard[i].suit, wSuit[i / 13]);

strcpy(wCard[i].face, wFace[i % 13]);

}

}

/\* 函数功能：将52张牌的顺序打乱以模拟洗牌过程 \*/

void Shuffle(CARD \*wCard)

{

int i, j;

CARD temp;

for (i = 0; i < 52; i++) /\*每次循环产生一个随机数，交换当前牌与随机数指示的牌\*/

{

j = rand() % 52; /\* 每次循环产生一个0~51的随机数 \*/

temp = wCard[i];

wCard[i] = wCard[j];

wCard[j] = temp;

}

}

/\* 函数功能：输出每张牌的花色和面值以模拟发牌过程 \*/

void Deal(CARD \*wCard)

{

int i;

for (i = 0; i < 52; i++)

{

printf("%9s%9s%c", wCard[i].suit, wCard[i].face, i % 2 == 0 ? '\n' : '\t');

}

printf("\n");

}

Q502.(10分)编程统计候选人的得票数。设有3个候选人，每次输入一个得票候选人的名字，不考虑弃权情况，要求最后输出各个候选人的得票结果（候选人姓名和参加投票人数由程序运行时输入）。

要求输入、输出在主函数中实现，统计候选人由函数实现。

按如下函数原型编写程序

void Vote(struct Person ld[], int n)

程序的运行示例如下：

输入3个候选人的基本信息：姓名

wang

song

zhao

输入参加投票的人数:5

输入5个参加投票人的投票情况

wang

wang

song

song

zhao

输出3个候选人的基本信息：姓名，票数

wang 2

song 2

zhao 1

#include <stdio.h>

#define N 3

struct Person

{

char name[16];

int count;

};

void Input(struct Person ld[], int n); //输入n个候选人的基本信息

void Vote(struct Person ld[], int n); //对n个候选人进行投票

void Output(struct Person ld[], int n); //输出n个候选人的基本信息

int main(int argc, char \*argv[])

{

struct Person leader[N];

Input(leader, N);

Vote(leader, N);

Output(leader, N);

return 0;

}

void Input(struct Person ld[], int n)

{

int i;

printf("输入%d个候选人的基本信息：姓名\n", n);

for (i = 0; i < n; i++)

scanf("%s", ld[i].name);

}

void Vote(struct Person ld[], int n)

{

int m, i, j; //m表示参加投票的人数

char ename[16];

printf("输入参加投票的人数:");

scanf("%d", &m);

printf("输入%d个参加投票人的投票情况\n", m);

for (i = 0; i < m; i++)

{

ld[i].count = 0;

}

for (i = 0; i < m; i++)

{

scanf("%s", ename);

for (j = 0; j < n; j++)

if (strcmp(ld[j].name, ename) == 0)

ld[j].count++;

}

}

void Output(struct Person ld[], int n)

{

int i;

printf("输出%d个候选人的基本信息：姓名，票数\n", n);

for (i = 0; i < n; i++)

printf("%s %d\n", ld[i].name, ld[i].count);

}

Q1328.(10分)定义一个结构体变量（包括年、月、日）。计算该日在本年中是第几天？注意闰年问题。

程序的运行示例如下：

请输入日期（年，月，日）

1990,2,14

2月14日是1990年的第45天。

#include <stdio.h>

struct

{

int year;

int month;

int day;

} date;

int main(void)

{

int days;

printf("请输入日期（年，月，日）\n");

scanf("%d,%d,%d", &date.year, &date.month, &date.day);

switch (date.month)

{

case 1:

days = date.day;

break;

case 2:

days = date.day + 31;

break;

case 3:

days = date.day + 59;

break;

case 4:

days = date.day + 90;

break;

case 5:

days = date.day + 120;

break;

case 6:

days = date.day + 151;

break;

case 7:

days = date.day + 181;

break;

case 8:

days = date.day + 212;

break;

case 9:

days = date.day + 243;

break;

case 10:

days = date.day + 273;

break;

case 11:

days = date.day + 304;

break;

case 12:

days = date.day + 334;

break;

}

if ((date.year % 4 == 0 && date.year % 100 != 0

|| date.year % 400 == 0) && date.month >= 3)

{

days += 1;

}

printf("\n%d月%d日是%d年的第%d天。", date.month, date.day, date.year, days);

return 0;

}

Q157.(15分)从键盘任意输入10个整数存入一个数组中，

然后任意输入一个整数x，采用顺序查找法，在数组中查找该数，

如果找到，则函数返回该数在数组中的下标位置，

并在主函数中打印该值；

如果没有找到，则返回-1，并在主函数中打印“Not Found!”。

要求按如下函数原型编程实现查找功能。

int Search(int a[], int n, int x);

在主函数中调用函数Search顺序查找x，

然后在主函数中打印查找结果。

要求必须按照题目要求和用函数编程，否则不给分。

\*\*要求输入10个整数的提示信息格式为：

"Input 10 numbers:\n"（每输入一个数，键一次回车）；

\*\*要求输入整数x的提示信息格式为：

"Input x:\n"

\*\*要求输出格式为:

找到时的打印格式为"Subscript of x is %d\n"

没找到时的打印格式为"Not Found!\n"

#include <stdio.h>

int Search(int a[], int n, int x);

main()

{

int a[10];

int n=10, i, pos, x;

printf("Input 10 numbers:\n");

for (i=0; i<n; i++)

{

scanf("%d", &a[i]); //1

}

printf("Input x:\n");

scanf("%d",&x); //1

pos = Search(a, n, x); //1

if (pos != -1)

{

printf("Subscript of x is %d\n", pos);

}

else

{

printf("Not found!\n");

}

}

int Search(int a[], int n, int x)

{

int i;

for (i=0; i<n; i++)

{

if (a[i] == x) //1

return i; //1

}

return -1; //1

}

Q113.(30分)输入某班学生某门课的成绩（最多不超过40人），

当输入为负值时，表示输入结束，

用函数编程统计成绩不低于平均分的学生人数。

要求：

(1)按如下函数原型进行编程：

int GetAboveAver(int score[], int n);

(2)在主函数中：

输入学生成绩，

然后调用函数GetAboveAver计算成绩不低于平均分的学生人数，

最后输出该人数。

要求:

(1)学生成绩和平均分均定义为int类型

(2)\*\*无输入提示信息

\*\*输入格式为："%d"

\*\*输出格式为："Students of above average is %d\n"

#include <stdio.h>

#define N 40

int GetAboveAver(int score[], int n);

void main()

{

int score[N], m, n;

n = -1; //1

do{

n++;

scanf("%d", &score[n]);//1

}while (score[n] >= 0); //e2

m = GetAboveAver(score, n); //2

printf("Students of above average is %d\n", m);

}

int GetAboveAver(int score[], int n)

{

int i, sum, aver,count;

count = 0;//1

sum = 0;//1

for (i=0; i<n; i++)//1

{

sum += score[i];//2

}

aver=sum/n;//1

for (i=0; i<n; i++)//1

{

if (score[i] >= aver)//2

count++;//1

}

return count;//1

}

Q381.(10分)\*验证卡布列克运算。即：任意一个四位数，只要它们各个位上的数字是不全相同的，就有这样的规律：

（1）将组成该四位数的四个数字由大到小排列，形成由这四个数字构成的最大的四位数；

（2）将组成该四位数的四个数字由小到大排列，形成由这四个数字构成的最小的四位数（如果四个数字中含有0，则得到的数不足四位）；

（3）求两个数的差，得到一个新的四位数（高位零保留）。

重复以上过程，最后得到的结果总是6174。这个数被称为卡布列克常数。请编写一个递归函数，完成以上的卡布列克运算。

\*\*输入格式要求："%d" 提示信息："Enter number:"

\*\*输出格式要求：" [%d]:%d-%d=%d\n"

程序运行示例如下：

Enter number: 1234

[1]:4321-1234=3087

[2]:8730-378=8352

[3]:8532-2358=6174

#include <stdio.h>

int count=0;

main()

{

int n;

printf("Enter number:");

scanf("%d",&n);

vr6174(n);

}

vr6174(int num)

{

int each[4],max,min;

if(num!=6174&&num)

{

parse\_sort(num,each);

max\_min(each,&max,&min);

num=max-min;

printf(" [%d]:%d-%d=%d\n",++count,max,min,num);

vr6174(num);

}

}

parse\_sort(int num,int \*each)

{

int i,\*j,\*k,temp;

for(i=0;i<=4;i++)

{

j=each+3-i;

\*j=num%10;

num/=10;

}

for(i=0;i<3;i++)

for(j=each,k=each+1;j<each+3-i;j++,k++)

if(\*j>\*k)

{ temp=\*j; \*j=\*k; \*k=temp;}

return;

}

max\_min(int \*each,int \*max,int \*min)

{

int \*i;

\*min=0;

for(i=each;i<each+4;i++)

\*min=\*min\*10+\*i;

\*max=0;

for(i=each+3;i>=each;i--)

\*max=\*max\*10+\*i;

return;

}

Q310.(15分)从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。

要求按如下函数原型编写程序

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

\*\*输入提示信息要求：

"Input n(n<=10):\n"

"Input %d numbers:\n"

\*\*要求输入格式为："%d"

\*\*输出提示信息："Exchange results:"

\*\*要求输出格式为："%5d"

输出数组中所有数据后换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

int main()

{

int a[10], n;

printf("Input n(n<=10):\n");

scanf("%d", &n);

printf("Input %d numbers:\n", n);

ReadData(a, n);//1

MaxMinExchang(a, n);//1

printf("Exchange results:");

PrintData(a, n);//1

return 0;

}

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++) //1

{

scanf("%d", &a[i]);

}

}

void PrintData(int a[], int n) {

int i;

for (i = 0; i < n; i++)//1

{

printf("%5d", a[i]);//1

}

printf("\n");//1

}

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;//1

int i, temp;

for (i = 1; i < n; i++) //1

{

if (a[i] > maxValue) //1

{

maxValue = a[i];//1

maxPos = i;//1

}

if (a[i] < minValue) //1

{

minValue = a[i];

minPos = i;

}

}

temp = a[maxPos];//1

a[maxPos] = a[minPos];//1

a[minPos] = temp;//1

}

Q590.(10分)编写一个程序，将十进制整数n转换成二进制数。

\*\*输入格式要求："%d" 提示信息："n="

\*\*输出格式要求："%d" "the number of binary is "

程序运行示例如下：

n=37

the number of binary is 0000000000100101

#include <stdio.h>

void trans(int n, int b[]);

int main()

{

int b[16], n, i;

for (i = 0; i <= 15; i++)

{

b[i] = 0;

}

printf("n=");

scanf("%d", &n);

trans(n, b);

printf("the number of binary is ");

for (i = 15; i >= 0; i--)

{

printf("%d", b[i]);

}

printf("\n");

return 0;

}

void trans(int n, int b[])

{

int i = 0;

while (n != 0)

{

b[i] = n % 2;

i++;

n /= 2;

}

}

Q183.(15分)任意从键盘输入10个整数，用函数编程实现计算其中的最大值。

要求用数组作为函数参数编程实现计算最大值的函数FindMax()，

即按照如下函数原型编写该函数：

int FindMax(int num[], int n);

要求：1）在主函数中打印输出函数求得的最大值，

必须用函数编程，否则不给分。

2）\*\*输入格式要求：无任何信息提示，

每输入一个数，键一次回车，

格式要求为"%d"

3）\*\*输出格式要求："max=%d\n"

#include <stdio.h>

int FindMax(int a[],int n)

{

int Max,j;

Max=0; //1

for(j=1;j<n;j++) //1

{

if(a[j]>a[Max])//1

{

Max=j; //1

}

}

return a[Max];

}

main()

{

int a[10];

int i;

int Max;

for(i=0;i<10;i++)

{

scanf("%d",&a[i]); //1

}

Max=FindMax(a,10); //1

printf("max=%d\n",Max);

}

Q568.(40分)把如下两个无序数组a和b从小到大排序后，在按照从小到大的顺序一次存放到新的数组中。

int a[5]={9,78,33,12,23};

int b[8]={1,34,63,10,5,94,39,27};

要求输出在主函数中实现，排序和数组合并分别用函数实现，函数原型如下：

void sort(int a[], int n);

void addins(int a[], int b[], int c[], int n, int m);

\*\*输出格式要求："%4d"

程序运行示例为：

9 12 23 33 78

1 5 10 27 34 39 63 94

1 5 9 10 12 23 27 33 34 39 63 78 94

#include <stdio.h>

void sort(int a[], int n)

{

int i, j, t, \*p;

for (i = 0; i < n - 1; i++)

{

p = a + i;

for (j = i + 1; j < n; j++)

{

if (\*p > \*(a + j))

p = a + j;

}

if (p != (a + i))

{

t = \*(a + i);

\*(a + i) = \*p;

\*p = t;

}

}

}

void addins(int a[], int b[], int c[], int n, int m)

{

int i, j, k, l;

int \*p1 = a, \*p2 = b , \*p3 = c;

i = j = k = 0;

while (i < n && j < m)

{

if (\*(p1 + i) <= \*(p2 + j))

{

\*(p3 + k) = \*(p1 + i);

i++;

}

else

{

\*(p3 + k) = \*(p2 + j);

j++;

}

k++;

}

if (i == n)

{

for (l = j; l < m; l++)

{

\*(p3 + k) = \*(p2 + l);

k++;

}

}

if (j == m)

{

for (l = i; l < n; l++)

{

\*(p3 + k) = \*(p1 + l);

k++;

}

}

}

void main()

{

int a[5] = {9, 78, 33, 12, 23};

int b[8] = {1, 34, 63, 10, 5, 94, 39, 27};

int c[13];

int i;

int \*p1 = a, \*p2 = b, \*p3 = c;

sort(a, 5);

for (i = 0; i < 5; i++)

{

printf("%4d", \*(p1 + i));

}

printf("\n");

sort(b, 8);

for (i = 0; i < 8; i++)

{

printf("%4d", \*(p2 + i));

}

printf("\n");

addins(b, a, c, 8, 5);

for (i = 0; i < 13; i++)

{

printf("%4d", \*(p3 + i));

}

printf("\n");

}

Q440.(10分)填表格

将1、2、3、4、5和6填入2行3列的表中，要使得每一个列右边的数字比左边的数字大，每一行下面的数字比上面的数字大。按此要求，可有几种填写方法？

\*\*输入格式要求：提示信息："The possible table satisfied above conditions are:\n"

\*\*输出格式要求："\nNo.:%d" "\n%3d" "%3d"

程序运行示例如下：

The possible table satisfied above conditions are:

No.:1

1 2 4

3 5 6

No.:2

1 2 5

3 4 6

参考答案

#include <stdio.h>

#include <stdlib.h>

int count;

int main()

{

static int a[] = {1, 2, 3, 4, 5, 6};

printf("The possible table satisfied above conditions are:\n");

for (a[1] = a[0] + 1; a[1] <= 5; ++a[1])

for (a[2] = a[2] + 1; a[2] <= 5; ++a[2])

for (a[3] = a[0] + 1; a[3] <= 5; ++a[3])

for (a[4] = a[1] > a[3] ? a[1] + 1 : a[3] + 1; a[4] <= 5; ++a[4])

if (judl(a))

print(a);

return 0;

}

int judl(int s[])

{

int i, l;

for (l = 1; l < 4; l++)

for (i = l + 1; i < 5; ++i)

if (s[l] == s[i])

return(0);

return(1);

}

int print(int u[])

{

int k;

printf("\nNo.:%d", ++count);

for (k = 0; k < 6; k++)

if (k % 3 == 0)

printf("\n%3d", u[k]);

else

printf("%3d", u[k]);

}

Q543.(10分)从键盘输入某班学生某门课的成绩（每班人数最多不超过40人），当输入为负值时，表示输入结束，试编程从键盘任意输入一个学号，查找该学号学生的成绩。

\*\*输入格式要求："%ld"(学号) "%ld%d" 提示信息："Total students are %d\n" "Input the searching ID:" "Input student’s ID and score:"

\*\*输出格式要求："score = %d\n" "Not found!\n"

程序的两次运行示例如下：

① Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

Input the searching ID:070310123

score = 83

② Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

Input the searching ID:070310128

Not found!

参考答案

#include <stdio.h>

#define N 40

int ReadScore(int score[], long num[]); /\*ReadScore()函数原型\*/

int LinSearch(long num[], long x, int n); /\* LinSearch()原函数型\*/

int main()

{

int score[N], n, pos;

long num[N], x;

n = ReadScore(score, num); /\* 输入成绩和学号，返回学生总数 \*/

printf("Total students are %d\n", n);

printf("Input the searching ID:");

scanf("%ld", &x); /\* 以长整型格式从键盘输入待查找的学号x \*/

pos = LinSearch(num, x, n); /\* 查找学号为num的学生 \*/

if (pos != -1) /\* 若找到，则打印其分数 \*/

{

printf("score = %d\n", score[pos]);

}

else /\* 若未找到，则打印"未找到"提示信息\*/

{

printf("Not found!\n");

}

return 0;

}

/\* 函数功能：输入学生的学号及其某门课成绩，当输入负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[], long num[]) /\* ReadScore()函数定义 \*/

{

int i = -1; /\*i初始化为-1，循环体内增1后可保证数组下标从0开始\*/

do

{

i++;

printf("Input student’s ID and score:");

scanf("%ld%d", &num[i], &score[i]);

}

while (num[i] > 0 && score[i] >= 0); /\* 输入负值时结束成绩输入 \*/

return i; /\* 返回学生总数 \*/

}

/\*按线性查找法查找值为x的数组元素，若找到则返回x在数组中的下标位置，否则返回-1\*/

int LinSearch(long num[], long x, int n) /\* LinSearch()函数定义 \*/

{

int i;

for (i = 0; i < n; i++)

{

if (num[i] == x) return (i); /\* 若找到则返回x在数组中的下标 \*/

}

return (-1); /\* 若循环结束仍未找到，则返回-1 \*/

}

Q292.(15分)用函数编程统计平均成绩。要求在主函数中输入学生成绩，输入负数时,结束输入，调用子函数Average()计算平均成绩，并输出结果。

函数原型：int Average(int score[],int n)

\*\*\*\*要求输入提示信息为："Input score:"

\*\*\*\*输出格式要求为：

"Total students are %d"

"Average score is %d"

#include <stdio.h>

#define N 40

int Average(int score[],int n)

{

int i,sum=0;

for(i=0;i<n;i++)

{

sum=sum+score[i]; //1

}

return sum/n;//1

}

main()

{

int score[N],m;

int i=-1;

do{

i++;

printf("Input score:");

scanf("%d",&score[i]);

}while(score[i]>=0);//e2

printf("Total students are %d\n",i);//1

m=Average(score,i);//2

printf("Average score is %d\n",m);//1

}

Q250.(10分)插入排序（Insertion Sort）。用函数编程实现在一个按升序排序的数组中查找x应插入的位置，将x插入数组中，使数组元素仍按升序排列。

\*\*输入格式要求："%d" 提示信息："Input array size:" "Input array:" "Input x:"

\*\*输出格式要求："After insert %d:\n" "%4d"

程序运行示例如下：

Input array size:5

Input array:1 3 5 7 9

Input x:4

After insert 4:

1 3 4 5 7 9

参考答案

#include<stdio.h>

#define N 20 /\* 插入前数组最大元素个数 \*/

void Insert(int a[], int n, int x);

main()

{

int a[N+1]; /\* 定义数组长度为插入前的数组元素个数加1 \*/

int x, i, n;

printf("Input array size:");

scanf("%d", &n); /\* 输入插入前数组元素个数 \*/

printf("Input array:");

for (i=0; i<n; i++)

{

scanf("%d", &a[i]); /\* 输入插入前已按升序排序的数组元素 \*/

}

printf("Input x:");

scanf("%d", &x); /\* 输入待插入的元素x \*/

Insert(a, n, x); /\* 插入元素x到已排序数组中 \*/

printf("After insert %d:\n", x);

for (i=0; i<n+1; i++)

{

printf("%4d", a[i]); /\* 输出插入x后的数组元素 \*/

}

printf("\n");

}

/\* 函数功能：将x插入到一个已按升序排序的数组中 \*/

void Insert(int a[], int n, int x)

{

int i = 0, pos;

while (i < n && x > a[i]) /\* 查找待插入位置 \*/

{

i++;

}

pos = i; /\* 记录元素x应插入的数组下标位置pos \*/

for (i = n-1; i>= pos; i--)/\* 从尾部开始移动pos及其后所有的元素 \*/

{

a[i+1] = a[i]; /\* 向后复制数组元素 \*/

}

a[pos] = x; /\* 插入元素x到位置pos \*/

}

Q1344.(10分)求500以内的10个最大素数及其和并分别输出。

程序的运行示例如下：

499 491 487 479 467 463 461 457 449 443

sum=4696

参考答案

#include <stdio.h>

#include <math.h>

int sum\_prime(int a[]);

int prime(int n);

int main()

{

int i, a[10], sum;

sum = sum\_prime(a);

for (i = 0; i < 10; i++)

printf("%6d", a[i]);

printf("\n sum=%d\n", sum);

return 0;

}

int sum\_prime(int a[])

{

int i, j = 499, sum = 0;

for (i = 0; i < 10; )

{

if (prime(j))

{

sum = sum + j ;

a[i] = j;

i++;

}

j--;

}

return sum;

}

int prime(int n)

{

int i, k;

k = (int) sqrt((double)n);

for (i = 2; i <= k; i++)

{

if (n % i == 0)

return 0;

}

if (i >= k + 1)

return 1;

}

Q205.(15分)从键盘任意输入10个整数存入一个数组中，然后任意输入一个整数x，采用顺序查找法，在数组中查找该数，要求按如下函数原型编程实现查找功能。

int Search(int a[], int n, int x);

在主函数中调用函数Search顺序查找x，然后在主函数中打印查找结果。

如果找到，则函数返回该数在数组中的下标位置，并在主函数中打印该值；

如果没有找到，则返回-1，并在主函数中打印“Not Found!”。

要求必须按照题目要求和用函数编程，否则不给分。

\*\*要求输入10个整数的提示信息格式为：

"Input 10 numbers:\n"（每输入一个数，键一次回车）；

\*\*要求输入整数x的提示信息格式为：

"Input x:\n"

\*\*要求输出格式为:

找到时的打印格式为"Subscript of x is %d\n"

没找到时的打印格式为"Not Found!\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include <stdio.h>

int Search(int a[], int n, int x);

main()

{

int a[10];

int n=10, i, pos, x;

printf("Input 10 numbers:\n");

for (i=0; i<n; i++)//1

{

scanf("%d", &a[i]); //1

}

printf("Input x:\n");

scanf("%d",&x); //1

pos = Search(a, n, x); //2

if (pos != -1) //1

{

printf("Subscript of x is %d\n", pos);//1

}

else

{

printf("Not found!\n");//1

}

}

int Search(int a[], int n, int x)

{

int i;

for (i=0; i<n; i++) //1

{

if (a[i] == x) //2

return i; //1

}

return -1; //1

}

Q254.(10分)用函数编程实现两个数组中对应元素值的交换。

\*\*输入格式要求："%d" 提示信息："Input array size(n<=10):" "Input array a:" "Input array b:"

\*\*输出格式要求："Output array a:" "Output array b:" "%5d"

程序运行示例如下：

Input array size(n<=10):5

Input array a:

1 2 3 4 5

Input array b:

6 7 8 9 10

Output array a: 6 7 8 9 10

Output array b: 1 2 3 4 5

#include <stdio.h>

#define N 10

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void Swap(int \*x, int \*y);

int main()

{

int a[N], b[N], i, n;

printf("Input array size(n<=10):");

scanf("%d", &n);

printf("Input array a:");

ReadData(a, n);

printf("Input array b:");

ReadData(b, n);

for (i = 0; i < n; i++)

{

Swap(&a[i], &b[i]);

}

printf("Output array a:");

PrintData(a, n);

printf("Output array b:");

PrintData(b, n);

return 0;

}

/\* 函数功能：输入数组a的n个元素值 \*/

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]);

}

}

/\* 函数功能：输出数组a的n个元素值 \*/

void PrintData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

printf("%5d", a[i]);

}

printf("\n");

}

/\* 函数功能：两整数值互换 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

Q540.(10分)从键盘输入某班学生某门课的整数成绩（每班人数最多不超过40人），当输入成绩为负值时，表示输入结束，试编程计算并输出其平均分，并输出实际输入的学生人数。

\*\*输入格式要求："%d" 提示信息："Total students are %d\n" "Average score is %d\n" "Input score:"

\*\*输出格式要求："%d"

程序的运行示例如下：

Input score:80

Input score:100

Input score:-1

Total students are 2

Average score is 90

#include <stdio.h>

#define N 40

int Average(int score[], int n); /\* Average()函数原型 \*/

int ReadScore(int score[]); /\* ReadScore()函数原型 \*/

int main()

{

int score[N], aver, n;

n = ReadScore(score); /\*调用函数ReadScore()输入成绩，返回学生人数\*/

printf("Total students are %d\n", n);

aver = Average(score, n); /\*调用函数Average()计算平均分，返回平均分\*/

printf("Average score is %d\n", aver);

return 0;

}

/\* 函数功能：计算n个学生成绩的平均分 \*/

int Average(int score[], int n) /\* Average()函数定义 \*/

{

int i, sum = 0;

for (i = 0; i < n; i++)

{

sum += score[i];

}

return sum / n;

}

/\* 函数功能：输入学生某门课成绩，当输入成绩为负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[]) /\* ReadScore()函数定义 \*/

{

int i = -1; /\*i初始化为-1，循环体内增1后可保证数组下标从0开始\*/

do

{

i++;

printf("Input score:");

scanf("%d", &score[i]);

}

while (score[i] >= 0); /\* 输入负值时结束输入 \*/

return i; /\* 返回学生人数 \*/

}

Q253.(10分)大奖赛现场统分。已知某大奖赛有n个选手参赛，m（m>2）个评委为参赛选手评分（最高10分，最低0分）。统分规则为：在每个选手的m个得分中，去掉一个最高分和一个最低分后，取平均分作为该选手的最后得分。要求编程实现：

（1）根据n个选手的最后得分，从高到低输出选手的得分名次表，以确定获奖名单；

（2）根据各选手的最后得分与各评委给该选手所评分数的差距，对每个评委评分的准确性和评分水准给出一个定量的评价，从高到低输出各评委得分的名次表。

程序运行示例如下：

How many Athletes?5

How many judges?5

Scores of Athletes:

Athlete 1 is playing.

Please enter his number code:11

Judge 1 gives score:9.5

Judge 2 gives score:9.6

Judge 3 gives score:9.7

Judge 4 gives score:9.4

Judge 5 gives score:9.0

Delete a maximum score:9.7

Delete a minimum score:9.0

The final score of Athlete 11 is 9.500

Athlete 2 is playing.

Please enter his number code:12

Judge 1 gives score:9.0

Judge 2 gives score:9.2

Judge 3 gives score:9.1

Judge 4 gives score:9.3

Judge 5 gives score:8.9

Delete a maximum score:9.3

Delete a minimum score:8.9

The final score of Athlete 12 is 9.100

Athlete 3 is playing.

Please enter his number code:13

Judge 1 gives score:9.6

Judge 2 gives score:9.7

Judge 3 gives score:9.5

Judge 4 gives score:9.8

Judge 5 gives score:9.4

Delete a maximum score:9.8

Delete a minimum score:9.4

The final score of Athlete 13 is 9.600

Athlete 4 is playing.

Please enter his number code:14

Judge 1 gives score:8.9

Judge 2 gives score:8.8

Judge 3 gives score:8.7

Judge 4 gives score:9.0

Judge 5 gives score:8.6

Delete a maximum score:9.0

Delete a minimum score:8.6

The final score of Athlete 14 is 8.800

Athlete 5 is playing.

Please enter his number code:15

Judge 1 gives score:9.0

Judge 2 gives score:9.1

Judge 3 gives score:8.8

Judge 4 gives score:8.9

Judge 5 gives score:9.2

Delete a maximum score:9.2

Delete a minimum score:8.8

The final score of Athlete 15 is 9.000

Order of Athletes:

order final score number code

1 9.600 13

2 9.500 11

3 9.100 12

4 9.000 15

5 8.800 14

Order of judges:

order final score number code

1 9.937 1

2 9.911 2

3 9.859 3

4 9.833 4

5 9.714 5

Over!Thank you!

参考答案

#include <stdio.h>

#include <math.h>

#define ATHLETE 40 /\* 选手人数最高限 \*/

#define JUDGE 20 /\* 评委人数最高限 \*/

void CountAthleteScore(int sh[], float sf[], int n, float f[], int m);

void Sort(int h[], float f[], int n);

void Print(int h[], float f[], int n);

void CountJudgeScore(int ph[], float pf[], int m, float sf[], float f[],

int n);

int main()

{

int j, m, n;

int sh[ATHLETE]; /\* 选手的编号 \*/

int ph[JUDGE]; /\* 评委的编号 \*/

float sf[ATHLETE]; /\* 选手的最后得分 \*/

float pf[JUDGE]; /\* 评委的得分 \*/

float f[ATHLETE][JUDGE]; /\* 评委给选手的评分 \*/

printf("How many Athletes?");

scanf("%d", &n); /\* 输入选手人数 \*/

printf("How many judges?");

scanf("%d", &m); /\* 输入评委人数 \*/

for (j = 1; j <= m; j++)

{

ph[j] = j;

}

printf("Scores of Athletes:\n");

CountAthleteScore(sh, sf, n, \*f, m); /\* 现场为选手统计分数 \*/

CountJudgeScore(ph, pf, m, sf, \*f, n); /\* 为各个评委打分 \*/

printf("Order of Athletes:\n");

Sort(sh, sf, n); /\* 选手得分排序 \*/

Print(sh, sf, n); /\* 打印选手名次表 \*/

printf("Order of judges:\n");

Sort(ph, pf, m); /\* 评委得分排序 \*/

Print(ph, pf, m); /\* 打印评委名次表 \*/

printf("Over!Thank you!\n");

return 0;

}

/\* 函数功能：统计参赛选手的得分 \*/

void CountAthleteScore(int sh[], float sf[], int n, float f[], int m)

{

int i, j;

float max, min;

for (i = 1; i <= n; i++) /\* 第i个选手 \*/

{

printf("\nAthlete %d is playing.", i);

printf("\nPlease enter his number code:");

scanf("%d", &sh[i]);

sf[i] = 0;

max = 0; /\* 最高分初值设为最小值 \*/

min = 100; /\* 最低分初值设为最大值 \*/

for (j = 1; j <= m; j++) /\* 第j个评委 \*/

{

printf("Judge %d gives score:", j);

scanf("%f", &f[i \* m + j]);

sf[i] = sf[i] + f[i \* m + j]; /\* 累加评委对第i个选手的评分 \*/

if (max < f[i \* m + j]) /\* 找出最高分 \*/

{

max = f[i \* m + j];

}

if (min > f[i \* m + j]) /\* 找出最低分 \*/

{

min = f[i \* m + j];

}

}

printf("Delete a maximum score:%.1f\n", max);

printf("Delete a minimum score:%.1f\n", min);

sf[i] = (sf[i] - max - min) / (m - 2); /\*去掉一个最高分和最低分\*/

printf("The final score of Athlete %d is %.3f\n", sh[i], sf[i]);

}

}

/\* 函数功能：对分数从高到低排序 \*/

void Sort(int h[], float f[], int n)

{

int i, j, k, temp2;

float temp1;

for (i = 1; i <= n - 1; i++)

{

k = i;

for (j = i + 1; j <= n; j++)

{

if (f[j] > f[k]) k = j;

}

if (k != i)

{

temp1 = f[k];

f[k] = f[i];

f[i] = temp1;

temp2 = h[k];

h[k] = h[i];

h[i] = temp2;

}

}

}

/\* 函数功能：打印名次表 \*/

void Print(int h[], float f[], int n)

{

int i;

printf("order\tfinal score\tnumber code\n");

for (i = 1; i <= n; i++)

{

printf("%5d\t%11.3f\t%6d\n", i, f[i], h[i]);

}

}

/\* 函数功能：统计评委的得分 \*/

void CountJudgeScore(int ph[], float pf[], int m, float sf[], float f[],

int n)

{

int i, j;

for (j = 1; j <= m; j++) /\* 第j个评委 \*/

{

pf[j] = 0;

for (i = 1; i <= n; i++) /\* 第i个选手 \*/

{

pf[j] = pf[j] + (f[i \* m + j] - sf[i]) \* (f[i \* m + j] - sf[i]);

}

pf[j] = 10 - sqrt(pf[j] / n);

}

}

Q779.(15分)请按给定的函数原型编程实现两个数组中对应该元素值的交换（数组的长度定义成5）。

函数原型：void exchange(int a[5],int b[5])

要求：

在主函数中输入两个数组中各元素的内容;

调用函数exchange实现两个数组中对应该元素值的交换;

在主函数中输出交换后两个数组的内容。

\*\*\*\*输入提示信息格式: 无

\*\*\*\*输入数据格式要求: "%d"

\*\*\*\*输出提示信息格式:"Output array a:"

"Output array b:"

\*\*\*\*输出数据格式要求:"%5d"

参考答案

#include <stdio.h>

#include <stdlib.h>

void exchange(int a[5], int b[5]);

int main()

{

int a[5], b[5], i;

for (i = 0; i < 5; i++) //1

{

scanf("%5d", &a[i]);

}

for (i = 0; i < 5; i++) //1

{

scanf("%5d", &b[i]);

}

exchange(a, b);//1

printf ("Output array a:");//1

for (i = 0; i < 5; i++)//1

{

printf("%5d", a[i]);//1

}

printf("\n");

printf ("Output array b:");//1

for (i = 0; i < 5; i++)

{

printf("%5d", b[i]);//1

}

return 0;

}

void exchange(int a[5], int b[5])

{

int i, temp;

for (i = 0; i < 5; i++)//1

{

temp = a[i];//1

a[i] = b[i];//1

b[i] = temp;//1

}

}

Q243.(10分)输入某班学生某门课的成绩（最多不超过40人，具体人数由用户键盘输入），用函数编程统计不及格人数。

\*\*输入格式要求："%d" 提示信息："How many students?"

\*\*输出格式要求："Fail students = %d\n"

程序运行示例如下：

How many students?5

45 56 78 99 100

Fail students = 2

#include <stdio.h>

#define N 40

int GetFailNum(int score[], int n);

int main()

{

int i, n, fail, score[N];

printf("How many students?");

scanf("%d", &n);

for (i = 0; i < n; i++)

{

scanf("%d", &score[i]);

}

fail = GetFailNum(score, n);

printf("Fail students = %d\n", fail);

return 0;

}

/\* 函数功能：统计不及格人数 \*/

int GetFailNum(int score[], int n)

{

int i, count = 0;

for (i = 0; i < n; i++)

{

if (score[i] < 60) count++;

}

return count;

}

Q614.(10分)有n个整数，使其前面各数顺序向后移m个位置，最后m个数变成最前面的m个数。

\*\*输入格式要求："%d" 提示信息："the total numbers is:" "back m:"

\*\*输出格式要求："%d," "%d"

程序运行示例1如下：

the total numbers is:5

back m:2

1 2 3 4 5

4,5,1,2,3

程序运行示例2如下：

the total numbers is:6

back m:3

1 2 3 4 5 6

4,5,6,1,2,3

参考答案

int main()

{

int number[20], n, m, i;

printf("the total numbers is:");

scanf("%d", &n);

printf("back m:");

scanf("%d", &m);

for (i = 0; i < n - 1; i++)

scanf("%d,", &number[i]);

scanf("%d", &number[n - 1]);

move(number, n, m);

for (i = 0; i < n - 1; i++)

printf("%d,", number[i]);

printf("%d", number[n - 1]);

}

move(array, n, m)

int n, m, array[20];

{

int \*p, array\_end;

array\_end = \*(array + n - 1);

for (p = array + n - 1; p > array; p--)

\*p = \*(p - 1);

\*array = array\_end;

m--;

if (m > 0) move(array, n, m);

}

Q255.(10分)从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并返回它们所在数组中的位置。

\*\*输入格式要求："%d" 提示信息："Input n(n<=10):" "Input %d numbers:"

\*\*输出格式要求："%5d" "Exchange results:"

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

void Swap(int \*x, int \*y);

int main()

{

int a[10], n;

printf("Input n(n<=10):");

scanf("%d", &n);

printf("Input %d numbers:", n);

ReadData(a, n);

MaxMinExchang(a, n);

printf("Exchange results:");

PrintData(a, n);

return 0;

}

/\* 函数功能：输入数组a的n个元素值 \*/

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]);

}

}

/\* 函数功能：输出数组a的n个元素值 \*/

void PrintData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

printf("%5d", a[i]);

}

printf("\n");

}

/\* 函数功能：将数组a中的最大数与最小数位置互换 \*/

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;

int i;

for (i = 1; i < n; i++)

{

if (a[i] > maxValue)

{

maxValue = a[i];

maxPos = i;

}

if (a[i] < minValue)

{

minValue = a[i];

minPos = i;

}

}

Swap(&a[maxPos], &a[minPos]);

}

/\* 函数功能：两整数值互换 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

Q199.(50分)从键盘输入n个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。其中，n由用户从键盘输入。已知n值不超过10。

要求按如下函数原型编写程序

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

\*\*输入提示信息要求：

"Input n(n<=10):\n"

\*\*要求输入格式为："%d"

\*\*输出提示信息：

"Input %d numbers:\n"

"Exchange results:"

\*\*要求输出格式为："%5d"

输出数组中所有数据后换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

int main()

{

int a[10], n;

printf("Input n(n<=10):\n");

scanf("%d", &n);

printf("Input %d numbers:\n", n);

ReadData(a, n);//1

MaxMinExchang(a, n);//1

printf("Exchange results:");

PrintData(a, n);//1

return 0;

}

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++) //1

{

scanf("%d", &a[i]);

}

}

void PrintData(int a[], int n) {

int i;

for (i = 0; i < n; i++)//1

{

printf("%5d", a[i]);//1

}

printf("\n");//1

}

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;//1

int i, temp;

for (i = 1; i < n; i++) //1

{

if (a[i] > maxValue) //1

{

maxValue = a[i];//1

maxPos = i;//1

}

if (a[i] < minValue) //1

{

minValue = a[i];

minPos = i;

}

}

temp = a[maxPos];//1

a[maxPos] = a[minPos];//1

a[minPos] = temp;//1

}

Q275.(20分)输入某班学生某门课的成绩（最多不超过40人），

当输入为负值时，表示输入结束，

用函数编程统计成绩不低于平均分的学生人数。

要求：

(1)按如下函数原型进行编程：

int GetAboveAver(int score[], int n);

(2)在主函数中：

输入学生成绩，

然后调用函数GetAboveAver计算成绩不低于平均分的学生人数，

最后输出该人数。

要求:

(1)学生成绩和平均分均定义为int类型

(2)\*\*无输入提示信息

\*\*输入格式为："%d"

\*\*输出格式为："Students of above average is %d\n"

参考答案

#include <stdio.h>

#define N 40

int GetAboveAver(int score[], int n);

void main()

{

int score[N], m, n;

n = -1; //1

do{

n++;

scanf("%d", &score[n]);//1

}while (score[n] >= 0); //e2

m = GetAboveAver(score, n); //2

printf("Students of above average is %d\n", m);

}

int GetAboveAver(int score[], int n)

{

int i, sum, aver,count;

count = 0;//1

sum = 0;//1

for (i=0; i<n; i++)//1

{

sum += score[i];//2

}

aver=sum/n;//1

for (i=0; i<n; i++)//1

{

if (score[i] >= aver)//2

count++;//1

}

return count;//1

}

Q298.(15分)输入某班学生某门课的成绩（最多不超过40人），当输入为负值时，表示输入结束，

用函数编程统计成绩高于平均分的学生人数。

要求按如下原型编写统计成绩在全班平均分及平均分之上的学生人数

int GetAboveAver(int score[], int n);

\*\*输入要求：无输入提示

\*\*输出要求：

"Total students are %d\n"

"Average score is %d\n"

"Students of above average is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

参考答案

#include <stdio.h>

#define N 40

int Average(int score[], int n);

int ReadScore(int score[]);

int GetAboveAver(int score[], int n);

main() {

int score[N], m, n;

n = ReadScore(score);//1

printf("Total students are %d\n", n);//1

m = GetAboveAver(score, n);//1

printf("Students of above average is %d\n", m);//1

}

int Average(int score[], int n) {

int i, sum = 0;

for (i = 0; i < n; i++)//1

{

sum += score[i];//1

}

return sum / n;//1

}

int ReadScore(int score[]) {

int i = -1;

do {

i++;

scanf("%d", &score[i]);//1

} while (score[i] >= 0);//e2

return i;//1

}

int GetAboveAver(int score[], int n) {

int i, count = 0, aver;

aver = Average(score, n);//1

printf("Average score is %d\n", aver);//1

for (i = 0; i < n; i++)//1

{

if (score[i] >= aver) //1

count++;//1

}

return count;//1

}

Q123.(25分)从键盘输入10个整型数据，放入数组a 中，求其最大值，并输出。

\*\*要求输入提示信息为：无输入提示信息

\*\*要求输入格式为："%d"

\*\*要求输出格式为："max=%d\n"

参考答案

#include <stdio.h>

main()

{

int a[10],i,max;

for(i=0;i<10;i++)

{

scanf("%d",&a[i]); //2

}

max=a[0]; //2

for(i=1;i<10;i++) //2

{

if(max<a[i]) //2

{

max=a[i]; //2

}

}

printf("max=%d\n",max); //2

}

Q315.(15分)从键盘任意输入10个整数存入一个数组中，然后任意输入一个整数x，采用顺序查找法，在数组中查找该数，要求按如下函数原型编程实现查找功能。

int Search(int a[], int n, int x);

在主函数中调用函数Search顺序查找x，然后在主函数中打印查找结果。

如果找到，则函数返回该数在数组中的下标位置，并在主函数中打印该值；

如果没有找到，则返回-1，并在主函数中打印“Not Found!”。

要求必须按照题目要求和用函数编程，否则不给分。

\*\*要求输入10个整数的提示信息格式为：

"Input 10 numbers:\n"（每输入一个数，键一次回车）；

\*\*要求输入整数x的提示信息格式为：

"Input x:\n"

\*\*要求输出格式为:

找到时的打印格式为"Subscript of x is %d\n"

没找到时的打印格式为"Not Found!\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

参考答案

#include <stdio.h>

int Search(int a[], int n, int x);

main()

{

int a[10];

int n=10, i, pos, x;

printf("Input 10 numbers:\n");

for (i=0; i<n; i++)//1

{

scanf("%d", &a[i]); //1

}

printf("Input x:\n");

scanf("%d",&x); //1

pos = Search(a, n, x); //2

if (pos != -1) //1

{

printf("Subscript of x is %d\n", pos);//1

}

else

{

printf("Not found!\n");//1

}

}

int Search(int a[], int n, int x)

{

int i;

for (i=0; i<n; i++) //1

{

if (a[i] == x) //2

return i; //1

}

return -1; //1

}

Q244.(10分)从键盘输入某班学生某门课的成绩和学号（最多不超过40人），当输入为负值时，表示输入结束，用函数编程通过返回数组中最大元素的下标，查找并输出成绩的最高分及其所在的学生学号。

\*\*输入格式要求："%ld%d" 提示信息："Input student’s ID and score:" "input error!\n"

\*\*输出格式要求："Total students are %d\n" "The highest is:%ld, %d\n"

程序运行示例如下：

Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

The highest is:70310124, 88

参考答案

#include <stdio.h>

#define N 40

int ReadScore(int score[], long num[]);

int FindMax(int score[], int n);

int main()

{

int score[N], maxNum, n;

long num[N];

n = ReadScore(score, num); /\* 输入成绩，返回学生人数 \*/

printf("Total students are %d\n", n);

maxNum = FindMax(score, n); /\* 计算并返回最高分所在数组的下标 \*/

printf("The highest is:%ld, %d\n", num[maxNum], score[maxNum]);

return 0;

}

/\* 函数功能：输入学生某门课的成绩，当输入负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[], long num[])

{

int i = -1;

do

{

i++;

printf("Input student’s ID and score:");

scanf("%ld%d", &num[i], &score[i]);

}

while (score[i] >= 0 && num[i] >= 0);

return i;

}

/\* 函数功能：计算并返回最高分所在数组的下标 \*/

int FindMax(int score[], int n)

{

int max, i, maxNum = 0;

max = score[0];

for (i = 1; i < n; i++)

{

if (score[i] > max)

{

max = score[i];

maxNum = i;

}

}

return maxNum;

}

Q373.(10分)\*将1到9这九个数字分成三个3位数，要求第二个3位数，正好是第一个3位数的二倍，第三个3位数正好是是第一个3位数的三倍，问应当怎样分。

\*\*输出格式要求："No.%d: %d %d %d\n"

程序运行示例如下：

No.1: 192 384 576

No.2: 219 438 657

No.3: 273 546 819

No.4: 327 654 981

参考答案

#include <stdio.h>

int a[9];

main()

{

int m,count=0;

for(m=123;m<=333;m++)

if(ok(m,a)&&ok(2\*m,a+3)&&ok(3\*m,a+6))

printf("No.%d: %d %d %d\n",++count,m,2\*m,3\*m);

}

int ok(t,z)

int t,\*z;

{

int \*p1,\*p2;

for(p1=z;p1<z+3;p1++)

{

\*p1=t%10;

t/=10;

for(p2=a;p2<p1;p2++)

if(\*p1==0||\*p2==\*p1)

return 0;

}

return 1;

}

Q421.(10分)黑白子交换。有三个白子和三个黑子如下图布置：(-表示一个空格)

○ ○ ○ - ● ● ●

游戏的目的是用多少的步数将上图中白子和黑子的位置进行交换：

● ● ● - ○ ○ ○

游戏的规则是：一次只能移动一个棋子；棋子可以向空格中移动，也可以跳过一个对方的格子进入空格，但不能向后跳，也不能跳过两个子。请用计算机实现上述游戏。

程序运行示例如下：

No. 0:-----------

|\*|\*|\*| |@|@|@|

-----------

No. 1:-----------

|\*|\*| |\*|@|@|@|

-----------

No. 2:-----------

|\*|\*|@|\*| |@|@|

-----------

No. 3:-----------

|\*|\*|@|\*|@| |@|

-----------

No. 4:-----------

|\*|\*|@| |@|\*|@|

-----------

No. 5:-----------

|\*| |@|\*|@|\*|@|

-----------

No. 6:-----------

| |\*|@|\*|@|\*|@|

-----------

No. 7:-----------

|@|\*| |\*|@|\*|@|

-----------

No. 8:-----------

|@|\*|@|\*| |\*|@|

-----------

No. 9:-----------

|@|\*|@|\*|@|\*| |

-----------

No.10:-----------

|@|\*|@|\*|@| |\*|

-----------

No.11:-----------

|@|\*|@| |@|\*|\*|

-----------

No.12:-----------

|@| |@|\*|@|\*|\*|

-----------

No.13:-----------

|@|@| |\*|@|\*|\*|

-----------

No.14:-----------

|@|@|@|\*| |\*|\*|

-----------

No.15:-----------

|@|@|@| |\*|\*|\*|

-----------

参考答案

int number;

print(a)

int a[];

{

int i;

printf("No.%2d:-----------\n",number++);

printf(" ");

for(i=0;i<=6;i++)

printf("|%c",a[i]==1?'\*':(a[i]==2?'@':' '));

printf("|\n -----------\n");

}

main()

{

int t[7]={1,1,1,0,2,2,2};

int i,flag;

print(t);

while(t[0]+t[1]+t[2]!=6||t[4]+t[5]+t[6]!=3)

{

flag=1;

for(i=0;flag&&i<5;i++)

if(t[i]==1&&t[i+1]==2&&t[i+2]==0)

{change(&t[i],&t[i+2]);print(t);flag=0;}

for(i=0;flag&&i<5;i++)

if(t[i]==0&&t[i+1]==1&&t[i+2]==2)

{change(&t[i],&t[i+2]);print(t);flag=0;}

for(i=0;flag&&i<6;i++)

if(t[i]==1&&t[i+1]==0&&(i==0||t[i-1]!=t[i+2]))

{change(&t[i],&t[i+1]);print(t);flag=0;}

for(i=0;flag&&i<6;i++)

if(t[i]==0&&t[i+1]==2&&(i==5||t[i-1]!=t[i+2]))

{change(&t[i],&t[i+1]);print(t);flag=0;}

}

}

change(n,m)

int \*n,\*m;

{

int term;

term=\*n;\*n=\*m;\*m=term;

}

Q110.(30分)用一个整型数组feedback保存调查的20个反馈意见，

其中反馈意见是1-10范围中的一个整数。

用函数编程计算反馈意见的众数。

众数是数组中出现次数最多的那个数。

（假设不会发生两个或两个以上的反馈意见出现次数相同的情况）

要求：

（1）任意从键盘输入20个值在1-10范围中的整数，

编写函数Mode，求众数，

主函数调用函数Mode，并输出众数。

（2）计算n个数的众数的函数原型：

int Mode(int answer[], int n)

（3）\*\*输入提示信息格式要求："Input 20 feedbacks:\n"

\*\*输出格式要求："Mode value=%d\n"

（4）不要使用结构体

参考答案

#include <stdio.h>

#define M 20

#define N 11

int Mode(int answer[], int n);

main()

{

int i, feedback[M];//1

int modeval;

printf("Input 20 feedbacks:\n");

for (i=0; i<M; i++)//2

{

scanf("%d", &feedback[i]);//1

}

modeval=Mode(feedback, M);//2

printf("Mode value=%d\n", modeval);//1

}

int Mode(int answer[], int n) //1

{

int i, grade, max = 0, modeValue = 0, count[N] = {0};//1

for (i=0; i<n; i++)//2

{

count[answer[i]]++;//2

}

for (grade=1; grade<=N-1; grade++)//2

{

if (count[grade] > max)//2

{

max = count[grade];//1

modeValue = grade;//1

}

}

return modeValue;//1

}

Q1385 现从键盘任意输入10个整数，用函数编程实现将其中最大数与最小数的位置对换后，再输出调整后的数组。

#include <stdio.h>

#define ARRSIZE 10 //去掉末尾的;

void MaxMinExchang(int a[], int n)

{

int maxValue=a[0], minValue=a[0], maxPos=0, minPos=0;

int i, temp;

for (i=1; i<n; i++)

{

if (a[i] > maxValue)

{

maxValue = a[i];

maxPos = i;

}

else if (a[i] < minValue)

{

minValue = a[i];

minPos = i;

}

}

temp = a[maxPos];

a[maxPos] = a[minPos];

a[minPos] = temp;

}

main()

{

int a[ARRSIZE], i, n;

printf("Input n(n<=10):");

scanf("%d", &n) ;

printf("Input %d Numbers:\n", n);

for (i=0; i<n; i++)

{

scanf("%d", &a[i]);

}

MaxMinExchang(a, n);

printf("After MaxMinExchange:\n");

for (i=0; i<n; i++)

{

printf("%4d", a[i]);

}

printf("\n") ;

}

Q177.(15分)任意从键盘输入10个整数，用函数编程实现计算其中的最大值。

要求用数组作为函数参数编程实现计算最大值的函数FindMax()，

即按照如下函数原型编写该函数：

int FindMax(int num[], int n);

要求：1）在主函数中打印输出函数求得的最大值，

必须用函数编程，否则不给分。

2）\*\*输入格式要求：无任何信息提示，

每输入一个数，键一次回车，

格式要求为"%d"

3）\*\*输出格式要求："max=%d\n"

参考答案

#include <stdio.h>

int FindMax(int a[],int n)

{

int Max,j;

Max=0; //1

for(j=1;j<n;j++) //1

{

if(a[j]>a[Max])//1

{

Max=j; //1

}

}

return a[Max];

}

main()

{

int a[10];

int i;

int Max;

for(i=0;i<10;i++)

{

scanf("%d",&a[i]); //1

}

Max=FindMax(a,10); //1

printf("max=%d\n",Max);

}

Q503.(10分)建立一个链表，使链表中从头到尾的结点数据域依次是一个数组的各个元素的值。程序先建立链表然后再遍历输出（假定链表和数组均有6个整型元素）。

程序运行示例如下：

输入数组6个元素的值。

1 3 5 7 9 11

此链表各个结点的数据域为：1 3 5 7 9 11

#include <stdio.h>

#include <stdlib.h>

#define N 6

struct LNode

{

int data;

struct LNode \*next;

};

struct LNode\* create\_rear(int a[], int n);

void output(struct LNode \*h);

int main(int argc, char \*argv[])

{

/\*①定义一个数组、头指针\*/

int a[N], i;

struct LNode\* head;

/\*②输入数组元素的值\*/

printf("输入数组%d个元素的值。\n", N);

for (i = 0; i < N; i++)

scanf("%d", &a[i]);

/\*③创建链表head,其结点的值依次为数组a元素的值\*/

head = create\_rear(a, N);

/\*④输出链表head\*/

printf("此链表各个结点的数据域为：");

output(head);

return 0;

}

struct LNode\* create\_rear(int a[], int n)

{

/\*新建一个链表h，每个结点依次插入到链尾，将链表的头指针返回 \*/

struct LNode \*h = NULL;

struct LNode \*s, \*r; /\*用s指向要插入结点，r指向链表的尾结点\*/

int i;

for (i = 0; i < n; i++)

{

s = (struct LNode \*)malloc(sizeof(struct LNode));

s->data = a[i];

s->next = NULL;

if (h == NULL)

h = s; /\*如果链表为空，则头指针h指向s \*/

else

r->next = s; /\*否则将s链接到尾结点r之后 \*/

r = s; /\*将r指向尾结点 \*/

}

return h; /\*返回链表的头指针\*/

}

void output(struct LNode \*h)

{

/\*将链表h的各个结点的数据域依次输出，即遍历该链表\*/

struct LNode \*p = h;/\*从第一个结点开始，用p依次指向各个结点\*/

while (p)

{

/\*只要p是一个非空结点，则输出其数据域，然后将p后移\*/

printf("%d ", p->data);

p = p->next; //将p后移

}

printf("\n");

}

Q565.(30分)程序中函数 transform()的功能: 将一个由二进制数字字符组成的字符串转换为与其值相等的十进制整数。规定输入的字符串最多只能包含10位二进制数字字符。

\*\*输入格式要求：

要求对输入的二进制数字字符组成的字符串进行长度和内容确为二进制的判断。

提示信息："输入一个二进制字符串（10位）:" "错误：字符串太长\n" "错误:%c 不是一个二进制字符\n"

\*\*输出格式要求："输入的二进制字符串为" "\n%s 转换成十进制整数为：%d\n"

程序运行示例如下：

输入一个二进制字符串（10位）:10010

输入的二进制字符串为10010

10010 转换成十进制整数为：18

提示：exit(0)函数在stdlib.h库中定义。

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

int transform(char ch[])

{

int n=0,i=0;

while (ch[i] != '\0')

{

n = n \* 2 + ch[i] - '0';

i++;

}

return n;

}

void main()

{

char s[10];

int i;

int n;

printf("输入一个二进制字符串（10位）:");

gets(s);

if (strlen(s) > 10)

{

printf("错误：字符串太长\n");

exit(0);

}

for (i=0;s[i]!='\0';i++)

{

if (s[i] != '0' && s[i] != '1')

{

printf("错误:%c 不是一个二进制字符\n", s[i]);

exit(0);

}

}

printf("输入的二进制字符串为");

puts(s);

n = transform(s);

printf("\n%s 转换成十进制整数为：%d\n", s, n);

}

Q1327.(10分)有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。

程序的运行示例如下：

请输入一个字符串：1234567890

从第几个字符开始复制？3

复制结果为：34567890

#include <stdio.h>

copystr(char \*p1, char \*p2, int m)

{

int n;

n = 0;

while (n < m - 1)

{

n++;

p1++;

}

while (\*p1 != '\0')

{

\*p2 = \*p1;

p1++;

p2++;

}

\*p2 = '\0';

}

int main(void)

{

int m;

char str1[20], str2[20];

printf("请输入一个字符串：\n");

scanf("%s", str1);

printf("从第几个字符开始复制？");

scanf("%d", &m);

if (strlen(str1) < m)

{

printf("输入错误。");

}

else

{

copystr(str1, str2, m);

printf("复制结果为：%s", str2);

}

return 0;

}

Q607.(10分)请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);

\*\*输入格式要求：无

\*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身

程序运行的输入输出样例1：

输入：

abc

输出：

No!

程序运行的输入输出样例2：

输入：

12321

输出：

12321

#include <stdio.h>

#include <string.h>

int Judger( char a[ ]);

int main()

{

char a[80];

int flag;

gets(a);

flag = Judger(a);

if (flag == 1)

{

puts(a);

}

else

{

printf("No!");

}

return 0;

}

int Judger( char a[ ])

{

char \*pStar, \*pEnd;

int i, j, flag = 1;

j = strlen(a);

pStar = &a[0];

pEnd = &a[j - 1];

for (i = 0; i < (j / 2); i++)

{

if (\*pStar != \*pEnd)

{

flag = 0;

}

pStar++;

pEnd--;

}

return flag;

}

Q1384.(10分)下面的函数MyStrcmp()用于实现函数strcmp()的功能，将两个字符串s和t进行比较，要求将两个字符串中第一个不相同字符的ASCII码值之差作为MyStrcmp()函数的返回值

#include <stdio.h>

int MyStrcmp(char s[], char t[]);

main()

{

char str1[20],str2[20];

printf("Input string:");

gets(str1);

printf("Input another string:");

gets(str2);

if (MyStrcmp(str1,str2) > 0)

{

printf("str1 大于 str2\n");

}

else if (MyStrcmp(str1,str2) < 0)

{

printf("str1 小于 str2\n");

}

else

{

printf("str1 等于 str2\n");

}

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++)

{

if (s[i] == '\0' ) return 0 ;

}

return (s[i] - t[i]);

}

Q173.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q594.(10分)编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

#include <stdio.h>

#include <stdlib.h>

#define N 100

int getline(char \*s, int len);

int main()

{

char line[N], b[20], \*ptr1, \*ptrb;

int n = 0, j, a[50];

printf("Please enter a string:");

getline(line, N);

ptr1 = line;

while (\*ptr1 != '\0')

{

ptrb = b;

for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)

{

\*ptrb = \*ptr1;

}

if (j)

{

\*ptrb = '\0';

\*(a + n) = atoi(b);

n++;

}

if (\*ptr1 != '\0')ptr1++;

}

printf("the result of output:\n");

for (j = 0; j < n; j++)

{

printf("%10d\n", \*(a + j));

}

return 0;

}

int getline(char \*s, int len)

{

int c;

char \*p = s;

while (--len > 0 && (c = getchar()) != '\n')

{

\*s++ = c;

}

\*s = '\0';

return s - p;

}

Q1349.(10分)题目：去除字符串中的空格并分解单词。

要求：在main()中调用下述各函数，完成字符串中去除空格并分解单词的功能。

函数原型如下：

函数一：单词内是否包含数字 int IsNumIn(char word[])

函数二：单词内是否包含空格 int IsSpcIn(char word[])

函数三：去掉单词的前后空格，tab键和换行符 Trim(char oldWord[], char newWord[])

函数四：单词内部有空格，分解成多个单词 Seg(char words[], char wArray[][100] ) 假设单词内部只有一个空格，没有两个连续空格的情况发生。

要求利用gets函数来获得用户输入的字符串，不能用scanf来获得用户输入的字符串

参考输入，输出

参考输入

ros2e

参考输出

error

参考输入

︺︺hello︺（ ︺代表一个空格）

参考输出

hello

参考输入

︺︺hello world︺

参考输出

hello

world

#include <stdio.h>

#include <ctype.h>

#define N 100

int IsNumIn(char word[]);

int IsSpcIn(char word[]);

void Trim(char oldWord[], char newWord[]);

void Seg(char words[], char wArray[][100] );

int main()

{

char str1[N],str[N],w\_Arr[20][100];

int i,j;

for(i=0;i<N;i++)

{

\*(str1+i)=0;

\*(str+i)=0;

}

for(i=0;i<20;i++)

{

for(j=0;j<100;j++)

w\_Arr[i][j]=0;

}

gets(str1);

Trim(str1,str);

if(IsNumIn(str))

printf("error");

else

{

if(IsSpcIn(str))

{

Seg(str,w\_Arr);

for(i=0;w\_Arr[i][0]!='\0';i++)

{

puts(w\_Arr[i]);

}

}

else

puts(str);

}

return 0;

}

int IsNumIn(char word[])

{

int i;

for(i=0;\*(word+i)!='\0';i++)

{

if(isdigit(\*(word+i)))

return 1;

}

return 0;

}

int IsSpcIn(char word[])

{

int i;

for(i=0;\*(word+i)!='\0';i++)

{

if(isspace(\*(word+i)))

return 1;

}

return 0;

}

void Trim(char oldWord[], char newWord[])

{

int i=0,j=0;

while(isspace(\*(oldWord+i)))

i++;

while( \*(oldWord+i) != '\0')

{

\*(newWord+j)=\*(oldWord+i);

i++;

j++;

}

i--;

j--;

for(;isspace(\*(oldWord+i));i--,j--)

{

\*(newWord+j)=0;

}

}

void Seg(char words[], char wArray[][100] )

{

int i,j=0,k=0;

for(i=0;i<N;i++)

{

if(isalpha(\*(words+i)))

{

wArray[j][k]=\*(words+i);

k++;

}

else

{

j++;

k=0;

}

}

}

Q499.(10分)实现验证输入的必须全部是数字0～9构成的字符串，利用函数checkNumber验证输入的字符串中的每个字符是否全为数字0～9之一，如果是则返回1，否则返回0。然后利用函数getNumberString容错地读取一个全由数字0～9组成的数字字符串。要求编写完整的程序测试运行你所写的函数。

程序的运行示例如下：

输入一个全为数字的字符串: abc

请输入一个全为数字的字符串:a123

请输入一个全为数字的字符串:1a23

请输入一个全为数字的字符串:123

你输入的正确的数字串为：123

#include <stdio.h>

//检查是否全是数字构成的串

int checkNumber(char \*str)

{

int i;

for (i = 0; i < strlen(str); i++)

{

if (str[i] < '0' || str[i] > '9')

{

return 0;

}

}

return 1;

}

char \*getNumberString()

{

static char string[100];

do

{

printf("请输入一个全为数字的字符串:\n");

scanf("%s", string);

}

while (!checkNumber(string));

return string;

}

int main(int argc, char \*argv[])

{

char input[100];

strcpy(input, getNumberString());

printf("你输入的正确的数字串为：%s\n", input);

return 0;

}

Q276.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q1125.(10分)下列给定程序中，函数fun()的功能是：从s所指字符串中，找出t所指字符串的个数作为函数值返回。如，当s所指字符串中的内容为abcdabfab，t所指字符串的内容为ab，则函数返回整数3。

#include <stdio.h>

int fun(char \*s, char \*t)

{

int n;

char \*p, \*r;

n = 0;

while (\*s)

{

p = s;

r = t;

while (\*r)

if (\*r == \*p)

{

r++;

p++;

}

else break;

if (\*r == '\0')

n++;

s++;

}

return n;

}

main()

{

char s[100], t[100];

int m;

printf("\nPlease enter string s: ");

scanf("%s", s);

printf("\nPlease enter substring t:");

scanf("%s", t);

m = fun(s, t);

printf("\nThe result is: m=%d\n", m);

}

Q1133.(10分)下面程序的Squeeze函数的功能是删除字符串s中所出现的与变量c相同的字符。

#include <stdio.h>

void Squeeze(char s[], char c);

main()

{

char a[80], c;

gets(a);

scanf("%c", &c);

Squeeze(a, c);

printf("%s\n", a);

}

void Squeeze(char s[], char c)

{

int i, j;

for (i = 0, j = 0; s[i] != '\0'; i++)

{

if (s[i] != c)

{

s[j] = s[i];

j++;

}

}

s[j] = '\0';

}

Q1369.(10分)以下程序实现从键盘输入一个字符串，将其字符顺序颠倒后重新存放，并输出这个字符串

#include <stdio.h>

#include <string.h>

void Inverse(char rstr[]);

main()

{

char str[80];

printf("Input a string:\n");

gets(str);

Inverse(str);

printf("The inversed string is:\n");

puts(str);

}

void Inverse(char rstr[])

{

int i,n;

char temp;

for(i=0, n=(strlen(rstr)-1); i<n; i++,n--)

{

temp = rstr[i];

rstr[i] = rstr[n];

rstr[n] = temp;

}

}

Q1316.(10分)写一个函数，将一个字符串中的元音字母复制到另一个字符串，然后输出。

程序的运行示例如下：

输入字符串：Hello,world!

字符串中的元音字母是eoo

#include <stdio.h>

int main()

{

char str[80], c[80];

void cpy();

printf("\n输入字符串：");

gets(str);

printf("%s", str);

cpy(str, c);

printf("\n字符串中的元音字母是%s", c);

return 0;

}

void cpy(char s[], char c[])

{

int i, j;

for (i = 0, j = 0; s[i] != '\0'; i++)

{

if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||

s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||

s[i] == 'u' || s[i] == 'U')

{

c[j] = s[i];

j++;

}

}

c[j] = '\0';

}

Q167.(20分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q108.(15分)用字符数组作函数参数编程，

利用一个数组实现字符串（允许输入带空格的字符串）的逆序存放。

要求如下：

（1）在主函数中从键盘输入字符串，字符串的最大长度为80个字符

调用Inverse函数，并输出逆序后的字符串。

（2）在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(3)\*\*输入提示信息格式要求："Input a string:\n"

\*\*输出提示信息要求："Inversed results:\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);//1

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])//1

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

Q128.(5分)调用GetInput()函数从键盘任意输入一个字符串，然后分别在GetInput()和main()中输出该字符串。

注意：

#include <stdio.h>

void GetInput(char\* s);

int main(void)

{

char str[80];

char \*ptr = str;

GetInput(ptr);

puts(ptr);

return 0;

}

void GetInput(char\* s)

{

scanf("%s", s);

puts(s);

}

Q139.(15分)请按给定的函数原型编程实现将字符数组中的字符串的第m个字符开始的n个字符逆序存放。要求在主函数读入字符串，且逆序存放后的字符串也在主函数打印。函数原型：

void inverse(char str[],int m,int n);

输入要求: 输入的一行字符串，应包含字母，数字以及空格字符

友情提示:

在执行输入字符串的函数之前,请用getchar();把输入缓冲区中的换行符读出!!!

\*\*\*\*输入提示信息和格式要求为：

"input m,n:"

"%d,%d"

"input the string:"

\*\*\*\*输出提示信息为:"the inverse string:"

#include <stdio.h>

#define N 100

void inverse(char str[],int m,int n);

main()

{

char str[N];

int m,n;

printf("input m,n:"); //1

scanf("%d,%d",&m,&n);

getchar();

printf("input the string:"); //1

gets(str);

inverse(str,m,n); //2

printf("the inverse string:%s",str); //1

}

void inverse(char str[],int m,int n)

{

int i;

char a[N];

for(i=0;i<n;i++)

{

a[i]=str[m-1+i]; //1

}

for(i=0;i<n;i++)

{

str[m+i-1]=a[n-1-i]; //1

}

}

Q498.(20分)对输入的字符串进行长度验证，保证输入的字符串的长度在指定的范围内，如果不在指定的范围内，则一直提示用户输入，直到输入合法长度的字符串为止。

程序的示例运行如下：

请输入一个字符串(长度为[3..5]个字符)：a

请输入一个字符串(长度为[3..5]个字符)：ab

请输入一个字符串(长度为[3..5]个字符)：abcdef

请输入一个字符串(长度为[3..5]个字符)：abc

你输入的字符串为：abc

#include <stdio.h>

void getString(char \*str, int min, int max)

{

int len;

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

while (len < min || len > max)

{

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

}

}

int main(int argc, char \*argv[])

{

char string[100];

getString(string, 3, 5);

printf("你输入的字符串为：%s\n", string);

return 0;

}

Q1273.(10分)

该程序的作用是将一个字符串中的元音字母复制到另一个字符串，然后输出。

#include<stdio.h>

void Cpy(char s[], char c[]);

main()

{

char str1[80], str2[80];

printf("输入一个字符串\n");

gets(str1);

Cpy(str1, str2); //改为：Cpy(str1, str2)

printf("输出一个字符串\n");

puts(str2);

}

void Cpy(char s[], char c[])

{

int i, j = 0; //改为：j=0;

for (i = 0; s[i] != '\0'; i++)

{

if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||

s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||

s[i] == 'u' || s[i] == 'U')

//将=改为：==

{

c[j] = s[i]; //改为:c[j] = s[i];

j++;

}

}

c[j] = '\0';

}

Q1113.(10分)输入一个字符串到数组S将S的前9个位数字的数存储到T，并将其转换为整数!例如：用户输入字符串1243abc3,则将12433取出以整数形式输出。

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int \*k)

{

int i;

\*k = 0;

for (i = 0; s[i]; i++)

if ('0' <= s[i] && s[i] <= '9')

{

t[\*k] = s[i];

\*k += 1;

}

}

main()

{

char str[100], tOrigin[9];

long int n = 0L;

int i, k;

printf("\nPlease enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

printf("The result is: %d", n);

}

Q1118.(10分)输入一个字符，如果此字符在已知的字符串中出现，则删除此字符，如删除成功就返回被删字符，否则返回'\0'。

#include <string.h>

#include <stdio.h>

char sh(char \*a, char x)

{

int i = 0, k, j;

char q;

while (a[i] != '\0') i++;

for (j = 0; j < i; j++)

{

if (a[j] == x)

{

q = a[j];

for (k = j; k < i; k++)

a[k] = a[k + 1];

return q;

}

}

return '\0';

}

main()

{

char a[6] = "seven", x, ch;

printf("The Original string is:");

puts(a);

printf("Enter x: ");

scanf("%c", &x);

ch = sh(a, x);

if (ch == '\0')

printf("%c not found!!!", x);

else

{

printf("The deleted charactor is :");

putchar(x);

}

printf("\nThe Original string changes to :");

puts(a);

}

Q114.(20分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q1160.(10分)

下面程序的功能是删除字符串中第一次出现的a字符。

#include <stdio.h>

#include <string.h>

void fun(char \*x,int n,int \*t)

{ int i,k=0;

x[n]='a';x[n+1]='\0';

while(x[k]!='a') k++;

if(k==n){\*t=0;}

else

{ for(i=k;i<n;i++)

x[i]=x[i+1];

x[i-1]='\0';

}

}

main()

{

char x[20];

int len,t;

gets(x);

puts(x);

len=strlen(x);

fun(x,len,&t);

if(t==0) printf("Not exist!\n");

else puts(x);

}

Q124.(20分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能。

要求：

（1）按如下函数原型进行编程：

int Mystrlen(char str[]);

（2）在主函数中：

任意输入一个字符串（长度不超过80个字符）;

调用函数Mystrlen()计算输入字符串的实际长度;

然后打印计算结果。

\*\*要求输入提示信息为："Please input the string:\n"

\*\*要求输出格式为:"%d\n"

注意：

（1）字符串中的空格计入字符串实际长度，

但是字符串结束标志不计入字符串实际长度。

（2）必须使用函数编程。否则没有成绩。

#include <stdio.h>

int Mystrlen(char str[]); //2

main()

{

char str[80];

int len;

printf ("Please input the string:\n"); //2

gets(str); //2

len = Mystrlen(str); //2

printf("%d\n", len); //2

}

int Mystrlen(char str[])

{

int len = 0, i;

for (i=0; str[i]!=0; i++) //2

{

len++; //2

}

return len; //2

}

Q286.(20分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能。

要求：

（1）按如下函数原型进行编程：

int Mystrlen(char str[]);

（2）在主函数中：

任意输入一个字符串（长度不超过80个字符）;

调用函数Mystrlen()计算输入字符串的实际长度;

然后打印计算结果。

\*\*要求输入提示信息为："Please input the string:\n"

\*\*要求输出格式为:"%d\n"

注意：

（1）字符串中的空格计入字符串实际长度，

但是字符串结束标志不计入字符串实际长度。

（2）必须使用函数编程。否则没有成绩。

#include <stdio.h>

int Mystrlen(char str[]); //2

main()

{

char str[80];

int len;

printf ("Please input the string:\n"); //2

gets(str); //2

len = Mystrlen(str); //2

printf("%d\n", len); //2

}

int Mystrlen(char str[])

{

int len = 0, i;

for (i=0; str[i]!=0; i++) //2

{

len++; //2

}

return len; //2

}

Q156.(15分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能，即在主函数中任意输入

一个字符串（长度不超过80个字符），调用该函数计算

输入字符串的实际长度，然后打印计算结果。

要求按如下函数原型编程实现计算字符串长度的功能。

int Mystrlen(char str[]);

要求必须按照题目要求和用函数编程，否则不给分。

\*\*要求输入提示信息格式为："Enter a string:"

\*\*输出格式为："The length of the string is:%d\n"

#include <stdio.h>

int Mystrlen(char str[]);

main()

{

char str[80];

int len;

printf("Enter a string:");

gets(str); //1

len = Mystrlen(str); //1

printf("The length of the string is:%d\n",len);

}

int Mystrlen(char str[]) //1

{

int len = 0, i;

for (i=0; str[i]!='\0'; i++) //1

{

len++; //1

}

return len; //1

}

AddFYJ

Q1616.(50分)按如下函数原型编写程序，从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

void Swap(int \*x, int \*y);

输入提示信息：

"Input n(n<=10):"

"Input %d numbers:"

输入格式："%d"

输出提示信息：

"Exchange results:"

输出格式："%5d"

参考答案

|  |
| --- |
| #include  <stdio.h>  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void  MaxMinExchang(int a[], int n);  void Swap(int \*x, int \*y);  int main()  {      int  a[10], n;      printf("Input n(n<=10):");      scanf("%d", &n);      printf("Input %d numbers:", n);      ReadData(a, n);      MaxMinExchang(a, n);      printf("Exchange results:");      PrintData(a, n);      return 0;  }  /\* 函数功能：输入数组a的n个元素值 \*/  void ReadData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          scanf("%d", &a[i]);      }  }  /\* 函数功能：输出数组a的n个元素值 \*/  void PrintData(int a[], int n)  {      int i;      for (i = 0; i < n; i++)      {          printf("%5d", a[i]);      }      printf("\n");  }  /\* 函数功能：将数组a中的最大数与最小数位置互换 \*/  void  MaxMinExchang(int a[], int n)  {      int  maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;      int  i;      for (i = 1; i < n; i++)      {          if (a[i] > maxValue)          {              maxValue = a[i];              maxPos = i;          }          if (a[i] < minValue)          {              minValue = a[i];              minPos = i;          }      }      Swap(&a[maxPos], &a[minPos]);  }  /\* 函数功能：两整数值互换 \*/  void  Swap(int \*x, int \*y)  {      int  temp;      temp = \*x;      \*x = \*y;      \*y = temp;  } |

Q500.(10分)利用指针能直接操作内存的特点来编程输出字符串中每个字符在内存中的存储编码（字符串中可以包含英文数字和汉字）。

程序的运行示例1如下：

请输入一个字符串，长度小于等于100：abcABC012中国人

该字符串的内存编码为： 61 62 63 41 42 43 30 31 32 d6 d0 b9 fa c8 cb

程序的运行示例2如下：

请输入一个字符串，长度小于等于100：12345上山打老虎

该字符串的内存编码为： 31 32 33 34 35 c9 cf c9 bd b4 f2 c0 cf bb a2

参考答案

|  |
| --- |
| #include <stdio.h>    int main(int argc, char \*argv[])  {      char str[101] = {'\0'};      char \*p1, \*p2;        printf("请输入一个字符串，长度小于等于100：");      scanf("%s", str);        p1 = (char \*)(str + strlen(str));      p2 = (char \*)str;        printf("该字符串的内存编码为： ");      while (p1 > p2)      {          printf("%x ", (unsigned char)\*p2 & 0xff);          p2++;      }        return 0;  } |

Q607.(10分)请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);

\*\*输入格式要求：无

\*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身

程序运行的输入输出样例1：

输入：

abc

输出：

No!

程序运行的输入输出样例2：

输入：

12321

输出：

### 12321参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  int  Judger( char  a[ ]);  int main()  {      char a[80];      int flag;      gets(a);      flag = Judger(a);      if (flag == 1)      {          puts(a);      }      else      {          printf("No!");      }      return 0;  }  int  Judger( char  a[ ])  {      char \*pStar, \*pEnd;      int i, j, flag = 1;      j = strlen(a);      pStar = &a[0];      pEnd = &a[j - 1];      for (i = 0; i < (j / 2); i++)      {          if (\*pStar != \*pEnd)          {              flag = 0;          }          pStar++;          pEnd--;      }      return flag;  } |

Q614.(10分)有n个整数，使其前面各数顺序向后移m个位置，最后m个数变成最前面的m个数。

\*\*输入格式要求："%d" 提示信息："the total numbers is:" "back m:"

\*\*输出格式要求："%d," "%d"

程序运行示例1如下：

the total numbers is:5

back m:2

1 2 3 4 5

4,5,1,2,3

程序运行示例2如下：

the total numbers is:6

back m:3

1 2 3 4 5 6

4,5,6,1,2,3

参考答案

|  |
| --- |
| int main()  {      int number[20], n, m, i;      printf("the total numbers is:");      scanf("%d", &n);      printf("back m:");      scanf("%d", &m);      for (i = 0; i < n - 1; i++)          scanf("%d,", &number[i]);      scanf("%d", &number[n - 1]);      move(number, n, m);      for (i = 0; i < n - 1; i++)          printf("%d,", number[i]);      printf("%d", number[n - 1]);  }  move(array, n, m)  int n, m, array[20];  {      int \*p, array\_end;      array\_end = \*(array + n - 1);      for (p = array + n - 1; p > array; p--)          \*p = \*(p - 1);      \*array = array\_end;      m--;      if (m > 0) move(array, n, m);  } |

Q594.(10分)编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #define N 100  int getline(char \*s, int len);  int main()  {      char line[N], b[20], \*ptr1, \*ptrb;      int n = 0, j, a[50];      printf("Please enter a string:");      getline(line, N);      ptr1 = line;      while (\*ptr1 != '\0')      {          ptrb = b;          for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)          {              \*ptrb = \*ptr1;          }          if (j)          {              \*ptrb = '\0';              \*(a + n) = atoi(b);              n++;          }          if (\*ptr1 != '\0')ptr1++;      }      printf("the result of output:\n");      for (j = 0; j < n; j++)      {          printf("%10d\n", \*(a + j));      }      return 0;  }    int getline(char \*s, int len)  {      int c;      char \*p = s;      while (--len > 0 && (c = getchar()) != '\n')      {          \*s++ = c;      }      \*s = '\0';      return s - p;  } |

Q1326.(10分)有n个人围成一圈，顺序排号。从第一人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

程序的运行示例如下：

请输入人数：30

### 30个人中最后留下的是第29号。参考答案

|  |
| --- |
| /\* n个人循环报数逢3退出求最后一人是第几号 \*/    #include <stdio.h>    #define     NMAX    50 /\* 定义最多人数为50人 \*/    int main(void)  {      int i, k, m, n, num[NMAX], \*p;        printf("请输入人数：");      scanf("%d", &n);        p = num;      for (i = 0; i < n; i++)      {          \*(p + i) = i + 1; /\* 以1至n为序给每个人编号 \*/      }      i = 0;      k = 0; /\* k为按1,2,3报数时的计数变量 \*/      m = 0; /\* m为退出人数 \*/      while (m < n - 1)      {          if (\*(p + i) != 0)          {              k++;          }          if (k == 3)          {              \*(p + i) = 0; /\* 对退出人员的编号置为0 \*/              k = 0;              m++;          }          i++;          if (i == n)          {              i = 0; /\* 报数到尾后i恢复为0 \*/          }      }      while (\*p == 0)      {          p++;      }      printf("%d个人中最后留下的是第%d号。", n, \*p);        return 0;  } |

Q1368.(10分)以下程序的功能是计算十个数据的平均值。找出其中的错误并改正之。

|  |
| --- |
| #include <stdio.h>  void main(void)  {      int i, sum;      float  avg;      int    sc[10], \*p = sc;        for (i=0, i<10, i++)      {              scanf("%d", p);              p++;              sum += \*p;      }      avg = sum / 10;      printf("avg=%f\n", avg);  } |

参考答案

|  |
| --- |
| #include <stdio.h>  void main(void)  {      int i, sum=0;  float avg;  int sc[10], \*p=sc;    for (i=0; i<10; i++)  {          scanf("%d", p);          p++;          sum+= \*(p-1);  }  avg = (float)sum/ 10;  printf("avg=%f", avg);  } |

Q1159.(10分)

|  |
| --- |
| 下面程序的功能是将数组元素倒置 ，找出其中的4处错误并改正之。  #include "stdio.h"  #define M 5;  main()  {   int a[M]={1,2,3,4,5};      int i,j,\*t;      i=0;j=M-1;      while(i)      {          t=\*(a+i);          \*(a+i)=\*(a+j);          \*(a+j)=t;          i++;j++      }      for(i=0;i<M;i++) printf("%d",(a+i));  } |

参考答案

|  |
| --- |
| #include "stdio.h"  #define M 5  main()  {   int a[M]={1,2,3,4,5};      int i,j,t;      i=0;j=M-1;      while(i<=j)      {          t=\*(a+i);          \*(a+i)=\*(a+j);          \*(a+j)=t;          i++;j--;      }      for(i=0;i<M;i++) printf("%d",\*(a+i));  } |

Q601.(10分)在一个字符串中找出最大的字符并放在第一个位置上，并将该字符前的原字符往后顺序移动，如：cbyab变成ycbab。

\*\*输入格式要求：用gets读入

\*\*输出格式要求：用puts输出

程序运行的输入输出样例：

输入为：

cbyab

输出为：

### ycbab参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char str[80], \*p = str, \*q, max;        gets(p);      max = \*(p++);        while (\*p != 0)      {          if (max < \*p)          {              max = \*p;              q = p;          }          p++;      }        p = q;        while (p > str)      {          \*p = \*(p - 1);          p--;      }        \*p = max;      puts(p);  } |

Q568.(40分)把如下两个无序数组a和b从小到大排序后，在按照从小到大的顺序一次存放到新的数组中。

int a[5]={9,78,33,12,23};

int b[8]={1,34,63,10,5,94,39,27};

要求输出在主函数中实现，排序和数组合并分别用函数实现，函数原型如下：

void sort(int a[], int n);

void addins(int a[], int b[], int c[], int n, int m);

\*\*输出格式要求："%4d"

程序运行示例为：

9 12 23 33 78

1 5 10 27 34 39 63 94

1 5 9 10 12 23 27 33 34 39 63 78 94

#include <stdio.h>

void sort(int a[], int n)

{

    int i, j, t, \*p;

    for (i = 0; i < n - 1; i++)

    {

        p = a + i;

        for (j = i + 1; j < n; j++)

        {

            if (\*p > \*(a + j))

                p = a + j;

        }

        if (p != (a + i))

        {

            t = \*(a + i);

            \*(a + i) = \*p;

            \*p = t;

        }

    }

}

void addins(int a[], int b[], int c[], int n, int m)

{

    int i, j, k, l;

    int \*p1 = a, \*p2 = b , \*p3 = c;

    i = j = k = 0;

    while (i < n && j < m)

    {

        if (\*(p1 + i) <= \*(p2 + j))

        {

            \*(p3 + k) = \*(p1 + i);

            i++;

        }

        else

        {

            \*(p3 + k) = \*(p2 + j);

            j++;

        }

        k++;

    }

    if (i == n)

    {

        for (l = j; l < m; l++)

        {

            \*(p3 + k) = \*(p2 + l);

            k++;

        }

    }

    if (j == m)

    {

        for (l = i; l < n; l++)

        {

            \*(p3 + k) = \*(p1 + l);

            k++;

        }

    }

}

void main()

{

    int a[5] = {9, 78, 33, 12, 23};

    int b[8] = {1, 34, 63, 10, 5, 94, 39, 27};

    int c[13];

    int i;

    int \*p1 = a, \*p2 = b, \*p3 = c;

    sort(a, 5);

    for (i = 0; i < 5; i++)

    {

        printf("%4d", \*(p1 + i));

    }

    printf("\n");

    sort(b, 8);

    for (i = 0; i < 8; i++)

    {

        printf("%4d", \*(p2 + i));

    }

    printf("\n");

    addins(b, a, c, 8, 5);

    for (i = 0; i < 13; i++)

    {

        printf("%4d", \*(p3 + i));

    }

    printf("\n");

}

Q1120.(10分)指针和字符数组的应用,用五种方法输出字符串“China”。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

int i = 0;

char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[5];

while (str[i] != '\0')

{

putchar(\*str);

str++;

}

scanf("%s", ptr);

puts(ptr);

ptr1 = str;

puts(ptr1);

ptr2 = "China";

puts(ptr2);

str2 = "China";

printf("%s", str2);

}

我是神奇的分割线，请在以下刷新题

Q578.(10分)分别按如下函数原形编程计算并输出求m\*m阶矩阵的转置矩阵。期中m的值由用户从键盘输入。已知m的值都不超过10。

输入函数原形：void InputMatrix(int a[][N], int m);

求转置矩阵函数原形：void Transpose(int a[][N], int at[][M], int m);

输出函数原形：void PrintMatrix(int at[][M], int m);

输入输出要求:

输入提示信息和格式信息：

"Input m:"

"%d "

"Input %d\*%d matrix:\n"

"%d"

输出提示信息和格式信息：

"The transposed matrix is:\n"

"%8d"

程序运行示例如下：

Input m:5

Input 5\*5 matrix:

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

16 17 18 19 20

21 22 23 24 25

The transposed matrix is:

1 6 11 16 21

2 7 12 17 22

3 8 13 18 23

4 9 14 19 24

5 10 15 20 25

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #define N 10  #define M 10  void InputMatrix(int a[][N], int m);  void Transpose(int a[][N], int at[][M], int m);  void PrintMatrix(int at[][M], int m);  int main()  {      int a[N][N], at[M][M];      int m;      printf("Input m:");      scanf("%d", &m);      printf("Input %d\*%d matrix:\n", m, m);      InputMatrix(a, m);      Transpose(a, at, m);      printf("The transposed matrix is:\n");      PrintMatrix(at, m);      return 0;  }  void InputMatrix(int a[][N], int m)  {      int i, k;      for (i = 0; i < m; i++)      {          for (k = 0; k < m; k++)          {              scanf("%d", &a[i][k]);          }      }  }  void Transpose(int a[][N], int at[][M], int m)  {      int i, j;      for (i = 0; i < m; i++)      {          for (j = 0; j < m; j++)          {              at[i][j] = a[j][i];          }      }  }  void PrintMatrix(int at[][M], int m)  {      int i, j;      for (i = 0; i < m; i++)      {          for (j = 0; j < m; j++)          {              printf("%8d", at[i][j]);          }          printf( "\n" );      }  } |

Q449.(10分)超长正整数的加法

请设计一个算法完成两个超长正整数的加法。

\*\*输出格式要求：" s1=" " s2=" "s1+s2="

程序运行示例如下：

3488934387589

374849389

s1=3488934387589

s2=374849389

s1+s2=3489309236978

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define HUNTHOU 10000  typedef struct node  { int data;    struct node \*next;  } NODE;  NODE \*insert\_after (u,num)     NODE \*u;     int  num;   { NODE \*v;     v=(NODE \*)malloc(sizeof(NODE));     v->data = num;     u->next = v;     return(v);   }  NODE \* addint(p,q)     NODE \*p,\*q;  {  NODE \*pp,\*qq,\*r,\*s,\*t;     int total,number,carry;     pp=p->next;  qq=q->next;     s=(NODE \*)malloc(sizeof(NODE));     s->data=-1;     t=s; carry=0;     while(pp->data!=-1 && qq->data!=-1)     {    total =pp->data + qq->data + carry;          number = total % HUNTHOU;          carry = total / HUNTHOU;          t=insert\_after(t,number);          pp=pp->next;          qq=qq->next;     }      r=(pp->data!=-1)? pp : qq ;      while (r->data!=-1)      {          total = r->data+carry;          number = total % HUNTHOU;          carry = total / HUNTHOU;          t = insert\_after(t,number);          r = r->next;      }      if (carry) t=insert\_after(t,1);      t->next=s;      return(s);   }   NODE \*inputint(void)   {  NODE \*s, \*ps, \*qs;      struct number {int num;         struct number \*np;      }  \*p,\*q;      int i,j,k;      long sum;      char c;      p=NULL;      while((c=getchar())!='\n' )          if(c>='0'&&c<='9')          {q=(struct number \*)malloc(sizeof(struct number));           q->num=c-'0';           q->np=p;           p=q;          }      s=(NODE \*)malloc(sizeof(NODE));      s->data=-1;      ps=s;      while(p!=NULL)      {          sum=0;          i=0;          k=1;          while(i<4 && p!=NULL)          {   sum=sum+k\*(p->num);              i++; p=p->np; k=k\*10;          }          qs=(NODE \*)malloc(sizeof(NODE));          qs->data=sum;          ps->next=qs;          ps=qs;      }      ps->next=s;      return(s);   }   printint(s)      NODE \*s;   { if(s->next->data!=-1 )      { printint(s->next);        if (s->next->next->data==-1)          printf("%d",s->next->data);        else        { int i,k=HUNTHOU;          for(i=1;i<=4;i++,k/=10)              putchar('0'+s->next->data%(k)/(k/10));        }      }   }   main()   { NODE \*s1,\*s2,\*s;     NODE \*inputint(),\*addint(),\*insert\_after();     s1=inputint();     s2=inputint();     printf("   s1=");     printint(s1);  putchar('\n');     printf("   s2=");     printint(s2);  putchar('\n');     s=addint(s1,s2);     printf("s1+s2=");     printint(s); putchar('\n');   } |

Q1724.(10分)检验并打印幻方矩阵。在下面的5×5阶幻方矩阵中，每一行、每一列、每一对角线上的元素之和都是相等的，试编写程序将这些幻方矩阵中的元素读到一个二维整型数组中，然后检验其是否为幻方矩阵，并将其按如下格式显示到屏幕上。

17 24 1 8 15

23 5 7 14 16

4 6 13 20 22

10 12 19 21 3

11 18 25 2 9

输出提示信息："It is a magic square!\n"

输出格式："%4d"（换行使用"\n"）

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #define   N  5  int main(void)  {      int  i, j;      int  x[N][N] = {{17, 24, 1, 8, 15}, {23, 5, 7, 14, 16},          {4, 6, 13, 20, 22}, {10, 12, 19, 21, 3}, {11, 18, 25, 2, 9}      };      int  rowSum[N], colSum[N], diagSum1, diagSum2;      int  flag = 1;      for (i = 0; i < N; i++)      {          rowSum[i] = 0;          for (j = 0; j < N; j++)          {              rowSum[i] = rowSum[i] + x[i][j];          }      }      for (j = 0; j < N; j++)      {          colSum[j] = 0;          for (i = 0; i < N; i++)          {              colSum[j] = colSum[j] + x[i][j];          }      }      diagSum1 = 0;      for (j = 0; j < N; j++)      {          diagSum1 = diagSum1 + x[j][j];      }      diagSum2 = 0;      for (j = 0; j < N; j++)      {          diagSum2 = diagSum2 + x[j][N - 1 - j];      }      if (diagSum1 != diagSum2)      {          flag = 0;      }      else      {          for (i = 0; i < N; i++)          {              if ((rowSum[i] != diagSum1) || (colSum[i] != diagSum1))                  flag = 0;          }      }      if (flag)      {          printf("It is a magic square!\n");          for (i = 0; i < N; i++)          {              for (j = 0; j < N; j++)              {                  printf("%4d", x[i][j]);              }              printf("\n");          }      }      else      {          printf("It is not a magic square!\n");      }      return 0;  } |

Q278.(20分)编程计算2×3阶矩阵A和3×2阶矩阵B之积C。

矩阵相乘的基本方法是：

矩阵A的第i行的所有元素同矩阵B第j列的元素对应相乘，

并把相乘的结果相加，最终得到的值就是矩阵C的第i行第j列的值。

要求：

(1)从键盘分别输入矩阵A和B，

输出乘积矩阵C

(2)

\*\*输入提示信息为：

输入矩阵A之前提示："Input 2\*3 matrix a:\n"

输入矩阵B之前提示："Input 3\*2 matrix b:\n"

\*\*输入矩阵中每个值的格式为："%d"

\*\*输出格式为：

输出矩阵C之前提示："Results:\n"

输出矩阵C中每个值的格式："%6d"

输出矩阵C各行结束时换行

例如：

Input 2\*3 matrix a:

1 2 3

0 1 2

Input 3\*2 matrix b:

1 -2

2 3

1 3

Results:

8 13

4 9

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  ROW 2  #define  COL 3  void main()  {      int a[ROW][COL], b[COL][ROW], c[ROW][ROW], i, j,k;        printf("Input 2\*3 matrix a:\n");      for (i=0; i<ROW ;i++)//1      {          for (j=0; j<COL; j++)//1          {              scanf("%d", &a[i][j]);//1          }      }      printf("Input 3\*2 matrix b:\n");      for (i=0; i<COL; i++)//1      {          for (j=0; j<ROW; j++) //1          {              scanf("%d",  &b[i][j]);//1          }      }      for (i=0; i<ROW; i++)//1      {          for (j=0; j<ROW; j++)//1          {              c[i][j] =   0 ;//2              for (k=0; k<COL; k++)//2              {                  c[i][j] = c[i][j]+a[i][k]\*b[k][j] ;//4              }          }      }      printf("Results:\n");      for (i=0; i<ROW; i++)//1      {          for (j=0; j<ROW; j++) //1          {              printf("%6d", c[i][j]);//1          }          printf("\n") ;//1      }  } |

Q387.(15分)请按给定要求编写程序，编程打印100~10000之间的所有完数。所谓完数是指一个数的所有因子（除其自身）之和恰好等于其自身。如6就是一个完数，因为6=1+2+3。判断一个数是否是完数请按给定的函数原型编程实现，若是完数函数返回值为1，否则返回值为0。函数原型如下：

int fun(int n);

\*\*\*输入提示信息\*\*\*：无

\*\*\*输入数据格式\*\*\*：无

\*\*\*输出数据格式\*\*\*："%d\n"

注：不按给定的函数原型编写程序,即使程序运行结果正确也不给分。

### 参考答案

|  |
| --- |
| #include<stdio.h>  int fun(int n);  main()  {      int i, flag;        for (i = 100; i <= 10000; i++)     //1      {          flag = fun(i);                 //2          if (flag)          {              printf("%d\n", i);         //1          }      }  }  int fun(int n)                        //2  {      int i, flag = 0, sum = 0;        //1        for (i = 1; i <= n / 2; i++)    //1      {          if (n % i == 0)             //1          {              sum = sum + i;          //1          }      }      if (sum == n)                  //1      {          flag = 1;      }        return flag;                    //1  } |

Q252.(15分)挑战类型表示的极限——大数的存储问题。编程计算并输出1~40之间的所有数的阶乘。提示：用一个50维的数组存储一个大数（n!），每一个数组元素存储大数中的一位数字。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："%d! = "

程序运行示例如下：

Input n:40

1! = 1

2! = 2

3! = 6

4! = 24

5! = 120

6! = 720

7! = 5040

8! = 40320

9! = 362880

10! = 3628800

11! = 39916800

12! = 479001600

13! = 6227020800

14! = 87178291200

15! = 1307674368000

16! = 20922789888000

17! = 355687428096000

18! = 6402373705728000

19! = 121645100408832000

20! = 2432902008176640000

21! = 51090942171709440000

22! = 1124000727777607680000

23! = 25852016738884976640000

24! = 620448401733239439360000

25! = 15511210043330985984000000

26! = 403291461126605635584000000

27! = 10888869450418352160768000000

28! = 304888344611713860501504000000

29! = 8841761993739701954543616000000

30! = 265252859812191058636308480000000

31! = 8222838654177922817725562880000000

32! = 263130836933693530167218012160000000

33! = 8683317618811886495518194401280000000

34! = 295232799039604140847618609643520000000

35! = 10333147966386144929666651337523200000000

36! = 371993326789901217467999448150835200000000

37! = 13763753091226345046315979581580902400000000

38! = 523022617466601111760007224100074291200000000

39! = 20397882081197443358640281739902897356800000000

40! = 815915283247897734345611269596115894272000000000

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #define SIZE   51  int main()  {      int data[SIZE];          /\* 存储50位数，为了方便记忆，不使用data[0] \*/      int index = 1;           /\* 数组元素个数，表示阶乘值的位数 \*/      int n;                     /\* 准备计算的阶乘中的最大数 \*/      int i, j, k;      for (i = 0; i < SIZE; i++)      {          data[i] = 0;       /\* 每个数组元素存储大数的每一位数字，全部初始化为0 \*/      }      data[1] = 1;            /\* 初始化，令1！=1 \*/      printf("Input n:");      scanf("%d", &n);      for (i = 1; i <= n; i++) /\* 计算从1到n之间所有数的阶乘值 \*/      {          for (j = 1; j <= index; j++) /\* 计算阶乘i! \*/          {              data[j] = data[j] \* i;/\* 每一位数字都乘以i \*/          }          for (k = 1; k < index; k++)          {              if (data[k] >= 10) /\* 阶乘值的每位数字应在0~9之内，若>=10，则进位 \*/              {                  data[k + 1] = data[k + 1] + data[k] / 10; /\* 当前位向前进位 \*/                  data[k] = data[k] % 10;  /\* 进位之后的值 \*/              }          }          /\* 单独处理最高位，若计算之后的最高位>=10，则位数index加1 \*/          while (data[index] >= 10 && index <= SIZE - 1)          {              data[index + 1] = data[index] / 10; /\* 向最高位进位 \*/              data[index] = data[index] % 10;    /\* 进位之后的值 \*/              index++;                               /\* 位数index加1 \*/          }          if (index <= SIZE - 1)      /\* 检验数组是否溢出，若未溢出，则打印阶乘值 \*/          {              printf("%d! = ", i);              for (j = index; j > 0; j--) /\* 从最高位开始打印每一位阶乘值 \*/              {                  printf("%d", data[j]);              }              printf("\n");          }          else                            /\* 若大于50，数组溢出，则提示错误信息 \*/          {              printf("Over flow!\n");              exit(1);          }      }      return 0;  } |

Q1312.(10分)打印魔方阵，所谓魔方阵是指这样的方阵，它的每一行、每一列和对角线之和均相等。例如，三阶魔方阵为：

8 1 6

3 5 7

4 9 2

要求打印出由1到n2的自然数构成的魔方阵（n为奇数）。

魔方阵中各数的排列规律如下：

（1）将1放在第一行中间一列；

（2）从2开始直到n×n，各数依次按下列规律存放：每一个数存放的行比前一个数的行数减1，列数加1；

（3）如果上一个数的行数为1，则下一个数的行数为n（指最下一行）；

（4）当上一个数的列数为n时，下一个数的列数应为1，行数减1；

（5）如果按上面规则确定的位置上已有数，或上一个数是第1行第n列时，则把下一个数放在上一个数的下面。

\*\*输出格式要求："%3d"

程序的运行示例如下：

请输入n（0<n<=15，n是奇数）：5

矩阵阶数是：5

17 24 1 8 15

23 5 7 14 16

4 6 13 20 22

10 12 19 21 3

11 18 25 2 9

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int a[16][16], i, j, k, p, m, n;        /\* 初始化 \*/      p = 1;      while (p == 1)      {          printf("请输入n（0<n<=15，n是奇数）：\n");          scanf("%d", &n);          if ((n != 0) && (n <= 15) && (n % 2 != 0))          {              printf("矩阵阶数是：%d\n", n);              p = 0;          }      }        for (i = 1; i <= n; i++)      {          for (j = 1; j <= n; j++)          {              a[i][j] = 0;          }      }      /\* 建立魔方 \*/      j = n / 2 + 1;      a[1][j] = 1;      for (k = 2; k <= n \* n; k++)      {          i = i - 1;          j = j + 1;          if ((i < 1) && (j > n))          {              i = i + 2;              j = j - 1;          }          else          {              if (i < 1)              {                  i = n;              }              if (j > n)              {                  j = 1;              }          }          if (a[i][j] == 0)          {              a[i][j] = k;          }          else          {              i = i + 2;              j = j - 1;              a[i][j] = k;          }      }        /\* 输出 \*/      for (i = 1; i <= n; i++)      {          for (j = 1; j <= n; j++)          {              printf("%3d", a[i][j]);          }          printf("\n");      }        return 0;  } |

Q1701.(10分)在下面使用指针数组的程序中存在一个错误，试分析这个程序，并上机运行，观察运行结果，找到这个错误，并分析出错的原因。

|  |
| --- |
| #include <stdio.h>  void  Print(char \*arr[], int len);  int main(void)  {      char  \*pArray[] = {"Fred","Barrey","Wilma","Betty"};      int    num = sizeof(pArray) / sizeof(char);      printf("Total string numbers = %d\n", num);  Print(pArray, num);  return 0;  }  void  Print(char \*arr[], int len)  {      int  i;      for (i=0; i<len; i++)      {          printf("%s,", arr[i]);  }  printf("\n");  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  void  Print(char \*arr[], int len);  int main(void)  {      char  \*pArray[] = {"Fred", "Barrey", "Wilma", "Betty"};      int    num = sizeof(pArray) / sizeof(char\*);      printf("Total string numbers = %d\n", num);      Print(pArray, num);      return 0;  }  void  Print(char \*arr[], int len)  {      int  i;      for (i = 0; i < len; i++)      {          printf("%s,", arr[i]);      }      printf("\n");  } |

Q356.(10分)求矩阵A(2\*3)的转置B(3\*2)。如

A=[1 4  2 5  3 6  ]

B=⎡ ⎣ 1 2 3  4 5 6  ⎤ ⎦

\*\*输出格式要求："%d "

程序运行示例如下：

1 4

2 5

3 6

### 参考答案

|  |
| --- |
| #include <stdio.h>  int a[2][3]={{1,2,3},{4,5,6}};  main()  {      int i,j,b[3][2];      for(i=0;i<=1;i++)          for(j=0;j<=2;j++)              b[j][i]=a[i][j];      for(i=0;i<=2;i++,printf("\n"))          for(j=0;j<=1;j++)              printf("%d ",b[i][j]);  } |

Q1336.(10分)定义一个长度为30的整型数组，按顺序赋予从2开始的偶数，然后按顺序每5个数求出一个平均值，放在另一个数组中，输出这个数组。

\*\*输出格式要求："%6d"

程序的运行示例如下：

6 16 26 36 46 56

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int a[30], b[6];      int i, j = 0, n = 1, sum = 0;      for (i = 0; i < 30; i++, n++)      {          a[i] = 2 \* n;          sum += a[i];          if (n % 5 == 0)          {              b[j++] = sum / 5;              sum = 0;          }      }      for (i = 0; i < 6; i++)      {          printf("%6d", b[i]);      }      return 0;  } |

Q1294.(10分)

|  |
| --- |
| 程序改错。编写一个函数Inverse()，实现将字符数组中的字符串逆序存放的功能。（4个错误）  #include<string.h>  #include<stdio.h>    #define ARR\_SIZE = 80;    void Inverse(char str[]);    main()  {      char a[ARR\_SIZE] ;        printf("Please enter a string: ");      gets(a);      Inverse(char a[]);      printf("The inversed string is: ");      puts(a);  }    void Inverse(char str[])  {      int   len, i, j;      char  \*temp;        len = strlen(str);      for (i=0; j=len-1; i<j; i++; j--)        {      temp = str[i];          str[i] = str[j];          str[j] = temp;      }  } |

### 参考答案

|  |
| --- |
| #include<string.h>  #include<stdio.h>    #define ARR\_SIZE 80    void Inverse(char str[]);    main()  {      char a[ARR\_SIZE] ;        printf("Please enter a string: ");      gets(a);      Inverse(a);      printf("The inversed string is: ");      puts(a);  }    void Inverse(char str[])  {      int   len, i, j;      char  temp;        len = strlen(str);      for (i=0, j=len-1; i<j; i++, j--)      {      temp = str[i];          str[i] = str[j];          str[j] = temp;      }  } |

Q1384.(10分)下面的函数MyStrcmp()用于实现函数strcmp()的功能，将两个字符串s和t进行比较，要求将两个字符串中第一个不相同字符的ASCII码值之差作为MyStrcmp()函数的返回值。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  int MyStrcmp(char s[], char t[]);  main()  {      char  str1[20],str2[20];        printf("Input string:");      gets(str1);      printf("Input another string:");      gets(str2);        if (MyStrcmp(str1[],str2[]) > 0)      {          printf("str1 大于 str2\n");      }      else if (MyStrcmp(str1[],str2[]) < 0)      {          printf("str1 小于 str2\n");      }      else      {          printf("str1 等于 str2\n");      }  }    int MyStrcmp(char s[], char t[])  {      int i;        for (i=0; s[i] == t[i]; i++);      {              if (s[i] = '\0' )  return 0 ;      }      return (s[i] - t[i]);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  int MyStrcmp(char s[], char t[]);  main()  {      char  str1[20],str2[20];      printf("Input string:");      gets(str1);      printf("Input another string:");      gets(str2);      if (MyStrcmp(str1,str2) > 0)      {          printf("str1 大于 str2\n");      }      else if (MyStrcmp(str1,str2) < 0)      {          printf("str1 小于 str2\n");      }      else      {          printf("str1 等于 str2\n");      }  }    int MyStrcmp(char s[], char t[])  {      int i;        for (i=0; s[i] == t[i]; i++)      {              if (s[i] == '\0' )  return 0 ;      }      return (s[i] - t[i]);  } |

Q168.(10分)

（附加题）找出一个2\*6的二维数组中的鞍点，即该位置上的元素是该行上的最大值，是该列上的最小值。二维数组也可能没有鞍点。

输入：二维数组中的元素；

输出：二维数组中的鞍点，若没有鞍点输出"no saddle point"

样例输入：

12 45 33 3 5 54

23 49 58 22 7 55

样例输出：

54

### 参考答案

|  |
| --- |
| #include<stdio.h>  main( )  {      int n,spoint,i,j,max,pos;      int find=0;      int arry[2][6];      for(i=0;i<2;++i)      {              for(j=0;j<6;j++)  //1              {                  scanf("%d",&arry[i][j]);              }      }      for(i=0;i<2;++i)      {          max=arry[i][0];          for(j=0;j<6;++j)  //1          {              if(max<arry[i][j])  //1              {                  max=arry[i][j];                  pos = j;              }          }          if(arry[(i+1)%2][pos]>max)  //1          {              printf("%d",max);   //1              find = 1;          }      }      if(!find)          printf("no saddle point");  } |

Q1559.(10分)1.输出helloword

### 参考答案

|  |
| --- |
| #include "stdio.h"    int main()  {      printf("Hello World!");      return 0;    } |

Q355.(10分)在一个二维数组中形成并输出如下矩阵:

A=⎡ ⎣ ⎢ ⎢ ⎢ ⎢ 1 2 3 4 5  1 1 2 3 4  1 1 1 2 3  1 1 1 1 2  1 1 1 1 1  ⎤ ⎦ ⎥ ⎥ ⎥ ⎥

程序运行示例如下：

1 1 1 1 1

2 1 1 1 1

3 2 1 1 1

4 3 2 1 1

5 4 3 2 1

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      int i,j,a[5][5];      for(i=0;i<=4;i++)          for(j=0;j<=4;j++)              if(i<=j) a[i][j]=1;              else a[i][j]=i-j+1;      for(i=0;i<=4;i++)      {          for(j=0;j<=4;j++)              printf("%d ",a[i][j]);          printf("\n");      }  } |

Q1385.(10分)程序改错，实现从键盘任意输入10个整数，用函数编程实现将其中最大数与最小数的位置对换后，再输出调整后的数组。

|  |
| --- |
| #include  <stdio.h>  #define ARRSIZE 10;    void  MaxMinExchang(int a[], int n)  {      int  maxValue=a[0], minValue=a[0], maxPos, minPos;      int  i, temp;        for (i=1; i<n; i++)      {          if (a[i] > maxValue)          {                  maxValue = a[i];                  maxPos = i;          }          else if (a[i] < minValue)          {                  minValue = a[i];                  minPos = i;          }      }      temp = a[maxPos];      a[maxPos] = a[minPos];      a[minPos] = temp;  }    main()  {      int a[n], i, n;        printf("Input n(n<=10):");      scanf("%d", &n) ;      printf("Input %d Numbers:\n", n);      for (i=0; i<n; i++)      {          scanf("%d", &a[i]);      }        MaxMinExchang(a[], n);        printf("After MaxMinExchange:\n");      for (i=0; i<n; i++)      {              printf("%4d", a[i]);      }      printf("\n");  } |

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #define ARRSIZE 10   //去掉末尾的;    void  MaxMinExchang(int a[], int n)  {      int  maxValue=a[0], minValue=a[0], maxPos=0, minPos=0;      int  i, temp;        for (i=1; i<n; i++)      {          if (a[i] > maxValue)          {                  maxValue = a[i];                  maxPos = i;          }          else if (a[i] < minValue)          {                  minValue = a[i];                  minPos = i;          }      }      temp = a[maxPos];      a[maxPos] = a[minPos];      a[minPos] = temp;      }    main()  {      int a[ARRSIZE], i, n;        printf("Input n(n<=10):");      scanf("%d", &n) ;      printf("Input %d Numbers:\n", n);      for (i=0; i<n; i++)      {          scanf("%d", &a[i]);      }        MaxMinExchang(a, n);        printf("After MaxMinExchange:\n");      for (i=0; i<n; i++)      {              printf("%4d", a[i]);      }      printf("\n") ;  } |

Q1242.(10分)

|  |
| --- |
| 下面程序用于将输入的两个字符串连接起来，然后打印连接后的字符串。找出其中错误并改正之。  #include <stdio.h>  #include <string.h>    main()  {      char    first[80];      char    second[80];        printf("Input the first string:");      gets(first);      printf("Input the second string:");      gets(second);      MyStrCat(first, second);      printf("The result is : %s\n", first);  }    void MyStrCat(char \*dest, char \*source);  {      int i = 0, j = 0;        while (\*(dest+i)!='\0')      {          i++;      }      for (; \*(source+i)!='\0'; i++,j++)      {          \*(dest+j) = \*(source+i);      }      \*(dest+j) = '\0';  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>  void MyStrCat(char \*dest, char \*source);  main()  {      char    first[80];      char    second[80];      printf("Input the first string:");      gets(first);      printf("Input the second string:");      gets(second);      MyStrCat(first, second);      printf("The result is : %s\n", first);  }    void MyStrCat(char \*dest, char \*source)  {      int i = 0, j = 0;        while (\*(dest+i)!='\0')      {          i++;      }      for (; \*(source+j)!='\0'; i++,j++)      {          \*(dest+i) = \*(source+j);        }      \*(dest+i) = '\0';  } |

Q1113.(10分)输入一个字符串到数组S将S的前9个位数字的数存储到T，并将其转换为整数!例如：用户输入字符串1243abc3,则将12433取出以整数形式输出。阅读程序，找出其中的错误，并改正之。

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int k)

{

int i;

k = 0;

for (i = 0; s[i]; i++)

if (0 <= s[i] && s[I] <= 9)

{

t[\*k] = s[i];

\*k += 1;

}

}

void main()

{

char str[100], tOrigin[9];

long int n;

int i, k;

printf("\nPlease enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

printf("The result is: %d", n);

}

### 参考答案

|  |
| --- |
| #include <math.h>  #include <stdio.h>    void fun(char \*s, char \*t, int \*k)  {      int i;      \*k = 0;      for (i = 0; s[i]; i++)          if ('0' <= s[i] && s[i] <= '9')          {              t[\*k] = s[i];              \*k += 1;          }  }    main()  {      char str[100], tOrigin[9];      long int n = 0L;      int i, k;      printf("\nPlease enter string s:");      scanf("%s", str);      fun(str, tOrigin, &k);      for (i = 0; i < k; i++)          n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);      printf("The result is: %d", n);  } |

Q179.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define SIZE 80  int MyStrcmp(char s[],char t[]);  main()  {      char s[SIZE],t[SIZE],i;      printf("Input s\n");      gets(s);//1      printf("Input t\n");      gets(t);//1      i=MyStrcmp(s,t);//2      if(i>0)//1          printf("string s>string t.\n");      else if(i<0)//1          printf("string s<string t.\n");      else//1          printf("string s=string t.\n");  }  int MyStrcmp(char s[], char t[])  {      int i;      for (i=0; s[i] == t[i]; i++ ) //6      {              if (s[i] ==  '\0' )  //2              return 0 ;//2      }      return (s[i]-t[i]); //2  } |

Q1733.(10分)计算1-1/2+1/3-1/4+...+1/99-1/100+...，直到最后一项的绝对值小于10e-4为止。

\*\*输出格式要求："sum = %f\n"

### 参考答案

|  |
| --- |
| #include  <stdio.h>  #include  <math.h>    main()  {      int n = 1;      float term = 1, sign = 1, sum = 0;      while (fabs(term) >= 1e-4)       //判断末项大小      {          term = sign / n;            //求出累加项          sum = sum + term;           //累加          sign = -sign;               //改变项的符号          n++;                            //分母加1      }      printf("sum = %f\n", sum);  } |

Q1230.(10分)

|  |
| --- |
| 下面程序用于从键盘输入3\*4矩阵的元素，通过调用函数FindMax，求出这3\*4矩阵元素中的最大值及其所在行列位置，然后输出这个最大值及其所在行列位置。（找出其中的4处错误，并改正之）  #include <stdio.h>  int FindMax(int x[3][4],int \*pRow,int \*pCol);  main()  {      int a[3][4], max, i, j, row, col;        for (i=0; i<3; i++)      {          for (j=0; j<4; j++)          {              scanf("%d",a[i][j]);          }      }      max = FindMax(a,&row,&col);      printf("max=%d,row=%d,col=%d\n",max,row,col);  }  int FindMax(int x[3][4],int \*pRow,int \*pCol)  {      int max, i, j;      max = x[0][0];      for (i=0; i<3; i++)      {          for (j=0; j<4; j++)          {              if (x[i][j] > max);              {                  max = x[i][j];                  \*pRow = i;                  \*pCol = j;              }          }      }      return max;  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  int FindMax(int x[3][4],int \*pRow,int \*pCol);  main()  {      int a[3][4], max, i, j, row, col;        for (i=0; i<3; i++)      {          for (j=0; j<4; j++)          {              scanf("%d",&a[i][j]);          }      }      max = FindMax(a,&row,&col);      printf("max=%d,row=%d,col=%d\n",max,row,col);  }  int FindMax(int x[3][4],int \*pRow,int \*pCol)  {      int max, i, j;      max = x[0][0];      \*pRow = 0;      \*pCol = 0;      for (i=0; i<3; i++)      {          for (j=0; j<4; j++)          {              if (x[i][j] > max)              {                  max = x[i][j];                  \*pRow = i;                  \*pCol = j;              }          }      }      return max;  } |

Q448.(10分)素数幻方

求四阶的素数幻方。即在一个4\*4的矩阵中，每一个格填入一个数字，使每一行、每一列和两条对角线上的四个数字所组成的四位数，均为可逆素数。

\*\*输入格式要求："%d" 提示信息："There are magic aquares with invertable primes as follow:\n"

\*\*输出格式要求："No.%d\n" "%6d"

程序运行示例如下：

There are magic aquares with invertable primes as follow:

No.1

1 1 9 3

1 0 0 9

9 2 2 1

3 1 9 1

No.2

1 1 9 3

1 0 2 1

9 0 2 9

3 9 1 1

......

### 参考答案

|  |
| --- |
| #include <math.h>  #include <stdio.h>  int number[210][5];  int select [110];  int array[4][5];  int count;  int selecount;  int larray[2][200];  int lcount[2];  main()  {      int i, k, flag, cc = 0, i1, i4;      printf("There are magic aquares with invertable primes as follow:\n");      for (i = 1001; i < 9999; i += 2)      {          k = i / 1000;          if (k % 2 != 0 && k != 5 && num(i))          {              number[count][0] = i;              process(count++);              if (number[count - 1][2] % 2 != 0 && number[count - 1][3] % 2 != 0                      && number[count - 1][2] != 5 && number[count - 1][3] != 5)                  select[selecount++] = count - 1;          }      }      larray[0][lcount[0]++] = number[0][0] / 100;      larray[1][lcount[1]++] = number[0][0] / 10;      for (i = 1; i < count; i++)      {          if (larray[0][lcount[0] - 1] != number[i][0] / 100)              larray[0][lcount[0]++] = number[i][0] / 100;          if (larray[1][lcount[1] - 1] != number[i][0] / 10)              larray[1][lcount[1]++] = number[i][0] / 10;      }      for (i1 = 0; i1 < selecount; i1++)      {          array[0][0] = select[i1];          copy\_num(0);          for (array[1][0] = 0; array[1][0] < count; array[1][0]++)          {              copy\_num(1);              if (!comp\_num(2))                  continue;              for (array[2][0] = 0; array[2][0] < count; array[2][0]++)              {                  copy\_num(2);                  if (!comp\_num(3))                      continue;                  for (i4 = 0; i4 < selecount; i4++)                  {                      array[3][0] = select[i4];                      copy\_num(3);                      for (flag = 1, i = 1; flag && i <= 4; i++)                          if (!find1(i)) flag = 0;                      if (flag && find2())                      {                          printf("No.%d\n", ++cc);                          p\_array();                      }                  }              }            }        }  }  num(int number)  {      int j;      if (!ok(number)) return(0);      for (j = 0; number > 0; number /= 10)          j = j \* 10 + number % 10;      if (!ok(j)) return(0);      return(1);  }  ok (int number)  {      int i, j;      if (number % 2 == 0) return(0);      j = sqrt((double)number) + 1;      for (i = 3; i <= j; i += 2)          if (number % i == 0)return(0);      return(1);  }  process(int i)  {      int j, num;      num = number[i][0];      for (j = 4; j >= 1; j--, num /= 10)          number[i][j] = num % 10;  }  copy\_num(int i)  {      int j;      for (j = 1; j <= 4; j++)          array[i][j] = number[array[i][0]][j];  }  comp\_num (int n)  {      static int ii;      static int jj;      int i, num, k, \*p;      int \*pcount;      switch (n)      {      case 2:          pcount = &lcount[0];          p = ⅈ              break;      case 3:          pcount = &lcount[1];          p = &jj;          break;      default:          return(0);      }      for (i = 1; i <= 4; i++)      {          for (num = 0, k = 0; k < n; k++)              num = num \* 10 + array[k][i];          if ( num <= larray[n - 2][\*p] )              for (; \*p >= 0 && num < larray[n - 2][\*p]; (\*p)--)                  ;          else              for (; \*p < \*pcount && num > larray[n - 2][\*p]; (\*p)++);          if ( \*p < 0 || \*p >= \*pcount )          {              \*p = 0;              return(0);          }          if ( num != larray[n - 2][\*p] )              return(0);        }      return(1);  }  find1 (int i)  {      int num, j;      for (num = 0, j = 0; j < 4; j++ )          num = num \* 10 + array[j][i];      return (find0(num));  }  find2(void)  {      int num1, num2, j, i;      for (num1 = 0, j = 0; j < 4; j++)          num1 = num1 \* 10 + array[j][j + 1];      for (num2 = 0, j = 0, i = 4; j < 4; j++, i--)          num2 = num2 \* 10 + array[j][i];      if (find0(num1))          return(find0(num2));      else return(0);  }  find0 (int num)  {      static int j;      if (num <= number[j][0])          for (; j >= 0 && num < number[j][0]; j--);      else          for (; j < count && num > number[j][0]; j++);      if (j < 0 || j >= count)      {          j = 0;          return(0);      }      if (num == number[j][0])          return(1);      else return(0);  }  p\_array(void)  {      int i, j;      for (i = 0; i < 4; i++)      {          for (j = 1; j <= 4; j++)              printf("%6d", array[i][j]);          printf("\n");        }  } |

Q1163.(10分)

|  |
| --- |
| 以下程序有若干语法错误。请找出并改正之。  #include <stdio.h>  struct date  {      int  y;m;d;  }  struct stu  {   char n[10];      struct date b;      int a;  }s={“Wang”,{1980,11,11},30};  main()  {      printf(“%c,%d,%d\n”,s.n,s.d,s.a);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  struct date  {      int  y,m,d;  };  struct stu  {   char n[10];      struct date b;      int a;  }s={"Wang",{1980,11,11},30};  main()  {      printf("%s,%d,%d\n",s.n,s.b.d,s.a);  } |

Q1562.(10分)从键盘任意输入三角形的三边长a、b、c，编程判断a、b、c是否构成三角形，若构成三角形则计算并输出三角形的面积，否则提示不构成三角形。计算三角形面积的公式为：

area=sqrt(s(s-a)(s-b)(s-c))

s=1/2(a+b+c)

其中sqrt为开平方计算。

\*\*输入提示信息格式: "Input a,b,c:\n";

\*\*输入数据格式要求: %f,%f,%f

\*\*输出数据格式要求:

若a,b,c构成三角形，则输出: "area=%.2f\n"

若a,b,c不构成三角形，输出: "No triangle!\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>                      //1  int main()  {      float a, b, c;      float s, area;        printf("Input a,b,c:\n");      scanf("%f,%f,%f", &a, &b, &c);      //1        if ( (a + b) > c && (a + c) > b && (b + c) > a ) //2      {          s = (float) 1.0 / 2 \* (a + b + c);      //1          area = (float) sqrt(s \* (s - a) \* (s - b) \* (s - c)); //1          printf("area=%.2f\n", area);     //1      }      else          printf("No triangle!\n");         //1        return 0;  } |

Q1625.(10分)按如下函数原型，采用梯形法编程实现，在积分区间[a,b]内计算函数y 1 =∫ 1 0 (1+x 2 )dx

和y 2 =∫ 3 0 x 1+x 2   dx

的定积分。其中，指向函数的指针变量f用于接收被积函数的入口地址。

|  |
| --- |
| Integral(float (\*f)(float), float a, float b); |

\*\*输出格式要求："y1=%f\ny2=%f\n"

### 参考答案

|  |
| --- |
| #include <stdio.h>  float Fun1(float x);  float Fun2(float x);  float Integral(float (\*f)(float), float a, float b);  int main()  {      float y1, y2;      y1 = Integral(Fun1, 0.0, 1.0);      y2 = Integral(Fun2, 0.0, 3.0);      printf("y1=%f\ny2=%f\n", y1, y2);      return 0;  }  /\*  函数功能：计算函数1+x\*x的函数值 \*/  float Fun1(float x)  {      return  1 + x \* x;  }  /\*  函数功能：计算函数x/(1+x\*x)的函数值 \*/  float Fun2(float x)  {      return x / (1 + x \* x);  }  /\*    函数功能：用梯形法计算函数的定积分 \*/  float Integral(float (\*f)(float), float a, float b)  {      float s, h;      int n = 100, i;      s = ((\*f)(a) + (\*f)(b)) / 2;      h = (b - a) / n;      for (i = 1; i < n; i++)      {          s += (\*f)(a + i \* h);      }      return s \* h;  } |

Q364.(10分)给定某年某月某日，将其转换成这一年的第几天并输出。

\*\*输入格式要求："%d%d%d"

\*\*输出格式要求："%d"

程序运行示例如下：

2012 11 8

313

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      static int day\_tab[2][13]={          {0,31,28,31,30,31,30,31,31,30,31,30,31},          {0,31,29,31,30,31,30,31,31,30,31,30,31}};      int y,m,d;      scanf("%d%d%d",&y,&m,&d);      printf("%d\n",day\_of\_year(day\_tab,y,m,d));  }  day\_of\_year(day\_tab,year,month,day)      int day\_tab[][13];      int year,month,day;  {      int i,j;      i=(year%4==0&&year%100!=0)||year%400==0;      for(j=1;j<month;j++)          day+=day\_tab[i][j];      return day;  } |

Q258.(30分)编程实现如下功能：在字符串中删除与某字符相同的字符。

要求输入和输出在主函数中实现，

在字符串中删除与某字符相同的字符用函数实现

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

### 参考答案

|  |
| --- |
| #include <stdio.h>  void  Squeeze(char s[], char c);  int main()  {      char  str[20], ch;      printf("Input a string:");      gets(str);      printf("Input a character:");      ch = getchar();      Squeeze(str, ch);      printf("Results:%s\n", str);      return 0;  }  void  Squeeze(char s[], char c)  {      int i, j;      for (i = j = 0; s[i] != '\0'; i++)      {          if (s[i] != c)          {              s[j] = s[i];              j++;          }      }      s[j] = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/  } |

Q388.(15分)请按给定要求编程计算方阵的主对角线之和。其中方阵的输入请在Input函数实现，计算主对角之和请在Compute函数实现，计算结果请在主函数打印，且给定的方阵最大为10x10，方阵实际大小请在主函数读入。Input函数和Compute函数的函数原型如下：

void Input(int a[][10],int n);

int Compute(int a[][10],int n);

\*\*\*输入提示信息\*\*\*："Input n(n<=10):\n"

\*\*\*输入数据格式\*\*\*："%d"

\*\*\*输入提示信息\*\*\*："Input matrix:\n"

\*\*\*输入数据格式\*\*\*："%d"

\*\*\*输出数据格式\*\*\*："sum=%d\n"

注：

1、请按行读入数据

2、请按给定的函数原型编写程序，否则即使程序运行结果正确，也不给分。

### 参考答案

|  |
| --- |
| #include<stdio.h>  void Input(int a[][10], int n);  int Compute(int a[][10], int n);  main()  {      int n, a[10][10];      int sum;        printf("Input n(n<=10):\n");      scanf("%d", &n);                 //1        Input(a, n);                     //2      sum = Compute(a, n);             //2        printf("sum=%d\n", sum);         //1  }  void Input(int a[][10], int n)  {      int i, j;        printf("Input matrix:\n");      for (i = 0; i < n; i++)           //1      {          for (j = 0; j < n; j++)       //1          {              scanf("%d", &a[i][j]);    //1          }      }  }  int Compute(int a[][10], int n)  {      int i;      int sum = 0;        for (i = 0; i < n; i++)          //1      {          sum = sum + a[i][i];         //1      }        return sum;                      //1  } |

Q1320.(10分)

输入2个学生5门课的成绩，分别用函数求：（1）每个学生的平均分；（2）每门课的平均分；（3）找出最高的分数所对应的学生和课程；（4）求出平均分方差：

(1/n)(∑xi2) - （（∑xi）/ n）2。

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define     N   2  #define     M   5    float score[N][M];  float a\_stu[N], a\_cor[M];    int main(void)  {      int i, j, r, c;      float h;      float s\_diff(void);      float highest(int \*, int \*);        r = 0;      c = 1;      input\_stu();      avr\_stu();      avr\_cor();      printf("\n 序号     课程1     2     3     4     5     平均分\n");      for (i = 0; i < N; i++)      {          printf("\n NO%2d", i + 1);          for (j = 0; j < M; j++)          {              printf("%8.2f", score[i][j]);          }          printf("%8.2f", a\_stu[i]);      }        printf("\n课平均");      for (j = 0; j < M; j++)      {          printf("%8.2f", a\_cor[j]);      }      h = highest(&r, &c);      printf("\n\n最高分%8.2f是%d号学生的第%d门课\n", h, r, c);      printf("\n方差 %8.2f", s\_diff());        return 0;  }    input\_stu(void)  {      int i, j;        for (i = 0; i < N; i++)      {          printf("\n输入学生%4d的5个成绩：\n", i + 1);          for (j = 0; j < M; j++)          {              scanf("%f", &score[i][j]);          }      }  }    avr\_stu(void)  {      int i, j;      float s;        for (i = 0; i < N; i++)      {          for (j = 0, s = 0; j < M; j++)          {              s += score[i][j];          }          a\_stu[i] = s / 5.0;      }  }    avr\_cor(void)  {      int i, j;      float s;        for (j = 0; j < M; j++)      {          s = 0;          for (i = 0; i < N; i++)          {              s += score[i][j];          }          a\_cor[j] = s / (float)N;      }  }    float highest(int \*r, int \*c)  {      float high;      int i, j;        high = score[0][0];      for (i = 0; i < N; i++)      {          for (j = 0; j < M; j++)          {              if (score[i][j] > high)              {                  high = score[i][j];                  \*r = i + 1;                  \*c = j + 1;              }          }      }      return high;  }    float s\_diff(void)  {      int i, j;      float sumx, sumxn;        sumx = 0.0;      sumxn = 0.0;        for (i = 0; i < N; i++)      {          sumx += a\_stu[i] \* a\_stu[i];          sumxn += a\_stu[i];      }      return (sumx / N - (sumxn / N) \* (sumxn / N));  } |

Q603.(10分)模拟文曲星上的猜数游戏，先由计算机随机生成一个各位相异的4位数字，由用户来猜，根据用户猜测的结果给出提示：xAyB其中，A前面的数字表示有几位数字不仅数字猜对了，而且位置也正确，B前面的数字表示有几位数字猜对了，但是位置不正确。最多允许用户猜的次数由用户从键盘输入。如果猜对，则提示“Congratulations!”；如果在规定次数以内仍然猜不对，则给出提示“Sorry, you haven't guess the right number!”。程序结束之前，在屏幕上显示这个正确的数字。

\*\*输入格式要求："%d"

\*\*输出格式要求："%d"

程序运行示例如下：

请输入最大猜测次数：5

4

8

6

5

3A0B

4

8

7

5

Congratulations!

### 参考答案

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  #include<time.h>  int main()  {      int i, k, a1, b1, c1, d1, a2, b2, c2, d2, m, x, y;      srand(time(NULL));      do      {          a1 = rand() % 9 + 0;          b1 = rand() % 9 + 0;          c1 = rand() % 9 + 0;          d1 = rand() % 9 + 0;      }      while (a1 == b1 || a1 == c1 || a1 == d1 || b1 == c1 || b1 == d1 || c1 == d1);      m = a1 \* 1000 + b1 \* 100 + c1 \* 10 + d1;      scanf("%d", &k);      for (i = 1; i <= k; i++)      {          scanf("%d%d%d%d", &a2, &b2, &c2, &d2);          x = 0;          y = 0;          if (a1 == a2)              x++;          if (b1 == b2)              x++;          if (c1 == c2)              x++;          if (d1 == d2)              x++;          if (a2 == b1 || a2 == c1 || a2 == d1)              y++;          if (b2 == a1 || b2 == c1 || b2 == d1)              y++;          if (c2 == a1 || c2 == b1 || c2 == d1)              y++;          if (d2 == a1 || d2 == b1 || d2 == c1)              y++;          if (x == 4 && y == 0)          {              printf("Congratulations!");              return 0;          }          else              printf("%dA%dB\n", x, y);      }      printf("Sorry,you haven't guess the right number!");      printf("%d", m);      return 0;  } |

Q600.(10分)从键盘为3\*3的矩阵输入数据，找出主对角线上最大的元素，以及所在的行号。

\*\*输入格式要求："%d"

\*\*输出格式要求："max=%d ,row=%d"

程序运行示例如下：

1 2 3

4 5 6

7 8 9

max=9 ,row=2

### 参考答案

|  |
| --- |
| #include<stdio.h>  main()  {      int i, j, row = 0, max;      int a[3][3];      for (i = 0; i < 3; i++)      {          for (j = 0; j < 3; j++)          {              scanf("%d", &a[i][j]);          }      }      max = a[0][0];      for (i = 0; i < 3; i++)      {          if (max < a[i][i])          {              max = a[i][i];              row = i;          }      }      printf("max=%d ,row=%d", max, row);  } |

Q459.(10分)写一个3×5矩阵的转置程序（a[i][j] = 5 \* i + j + 1），输出其原矩阵的值和转置以后的结果。

\*\*输入格式要求："%-4d" 提示信息："转置前：\n"

\*\*输出格式要求："------------------\n" "转置后：\n" ""%-4d""

程序运行结果如下：

转置前：

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

16 17 18 19 20

21 22 23 24 25

------------------

转置后：

1 6 11 16 21

2 7 12 17 22

3 8 13 18 23

4 9 14 19 24

5 10 15 20 25

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define ROW 5  #define COLUMN 5    int main()  {      int a[ROW][COLUMN];      int i, j, k;        for (i = 0; i < ROW ; i++)          for (j = 0; j < COLUMN ; j++ )              a[i][j] = 5 \* i + j + 1;  /\*对矩阵赋值\*/      printf("转置前：\n");      for (i = 0; i < ROW ; i++)      {          for (j = 0; j < COLUMN ; j++ )              printf("%-4d", a[i][j]);         /\*输出转置前的矩阵\*/          printf("\n");      }      printf("------------------\n");      for (i = 0; i < ROW ; i++ )      {          for (j = 0; j < i ; j++ )          {              k = a[i][j];              a[i][j] = a[j][i];              a[j][i] = k;          }      }      printf("转置后：\n");      for (i = 0; i < ROW ; i++ )      {          for (j = 0; j < COLUMN ; j++ )              printf("%-4d", a[i][j]);      /\*输出后转置后的矩阵\*/          printf("\n");        }  } |

Q1313.(10分)找出一个二维数组中的鞍点，即该位置上的元素在该行最大，在该列上最小。也可能没有鞍点。

\*\*输出格式要求："%5d"

程序的运行示例1如下：

输入行数：3

输入列数：3

第0行？

1 2 3

第1行？

4 5 6

第2行？

7 8 9

1 2 3

4 5 6

7 8 9

第0行，第2列的3是鞍点

程序的运行示例2如下：

输入行数：2

输入列数：2

第0行？

1 2

第1行？

4 1

1 2

4 1

矩阵中无鞍点！

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define     N   10  #define     M   10    int main()  {      int i, j, k, m, n, flag1, flag2, a[N][M], max, maxi, maxj;        printf("\n输入行数：");      scanf("%d", &n);      printf("\n输入列数：");      scanf("%d", &m);        for (i = 0; i < n; i++)      {          printf("第%d行？\n", i);          for (j = 0; j < m; j++)          {              scanf("%d", &a[i][j]);          }      }      for (i = 0; i < n; i++)      {          for (j = 0; j < m; j++)          {              printf("%5d", a[i][j]);          }          printf("\n");      }      flag2 = 0;      for (i = 0; i < n; i++)      {          max = a[i][0];          for (j = 0; j < m; j++)          {              if (a[i][j] > max)              {                  max = a[i][j];                  maxj = j;              }          }          for (k = 0, flag1 = 1; k < n && flag1; k++)          {              if (max > a[k][maxj])              {                  flag1 = 0;              }          }          if (flag1)          {              printf("\n第%d行，第%d列的%d是鞍点\n", i, maxj, max);              flag2 = 1;          }      }      if (!flag2)      {          printf("\n矩阵中无鞍点！\n");      }        return 0;  } |

Q1712.(10分)编程将下列矩阵中的元素向右移动一列，最右一列移至第一列。

1 4 6

8 10 12

输入格式：无

输出格式："%6d"（换行使用"\n"）

### 参考答案

|  |
| --- |
| #include <stdio.h>    #define  ROW 2  #define  COL 3    int main(void)  {      int a[ROW][COL] = {1, 4, 6, 8, 10, 12};      int i, j, temp;        for (i = 0; i < ROW; i++)      {          temp = a[i][COL - 1];           //将当前行最后一列暂存          for (j = COL - 2; j >= 0; j--)          {              a[i][j + 1] = a[i][j];  //将当前行其他列后移          }          a[i][0] = temp;             //将暂存数据赋予当前行0列      }      for (i = 0; i < ROW; i++)      {          for (j = 0; j < COL; j++)          {              printf("%6d", a[i][j]);          }          printf("\n");      }        return 0;  } |

Q359.(10分)编写程序实现将字符串str2拷贝到字符串str1。

\*\*输入格式要求："%s" 提示信息："Enter string 2:"

\*\*输出格式要求："Output string 1:%s\n"

程序运行的输入输出样例：

输入为

Enter string 2:good morning

输出为

Output string 1:good morning

### 参考答案

|  |
| --- |
| #include <stdio.h>  main()  {      char str1[100],str2[100];      int i;      printf("Enter string 2:");      gets(str2);      for(i=0;(str1[i]=str2[i])!='\0';i++)          ;      printf("Output string 1:%s\n",str1);  } |

Q350.(10分)写一个函数，计算x的n次方。

\*\*输入格式要求："%d%d" 提示信息："Enter x and n:"

\*\*输出格式要求："X power N is %ld.\n"

程序运行示例如下：

Enter x and n:2 8

X power N is 256.

### 参考答案

|  |
| --- |
| #include <stdio.h>  long power(x,n)  int x,n;  {      long p;      for(p=1;n>0;--n)          p\*=x;      return p;  }  main()  {      int x,n;      printf("Enter x and n:");      scanf("%d%d",&x,&n);      printf("X power N is %ld.\n",power(x,n));  } |

Q1565.(30分)从键盘输入任意一个3位整数，编程计算并输出它的逆序数（忽略整数前的正负号）。例如，输入-123，则忽略负号，由123分离出百位1、十位2、个位3，然后计算3\*100+2\*10+1=321，输出321.

题目要求： 输入提示为Input x:

输出格式为y=%d\n

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      int x = 0, y = 0, b0 = 0, b1 = 0, b2 = 0;      printf("Input x:");      scanf("%d", &x);      x = (int)fabs(x);      b2 = x / 100;      b1 = (x - b2 \* 100) / 10;      b0 = x % 10;      y = b0 \* 100 + b1 \* 10 + b2;      printf("y=%d\n", y);      return 0;  } |

Q453.(10分)请根据输入的学生成绩给出成绩等级的判断。判断规则如下：

如果输入的成绩大于等于90，则输出优秀；

如果输入的成绩小于90、大于等于80，则输出良好；

如果输入的成绩小于80、大于等于70，则输出中等；

如果输入的成绩小于70、大于等于60，则输出及格；

其它输出不及格。

\*\*输入格式要求："%d" 提示信息："Please input the score\n"

\*\*输出格式要求：" 优秀\n" " 良好\n" " 中等\n" " 及格\n" " 不及格\n"

程序运行示例如下：

Please input the score

96

优秀

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int iScore;      int iGrade;      printf("Please input the score\n");      scanf("%d", &iScore);      iGrade = iScore / 10;      switch (iGrade)      {      case 9:          printf(" 优秀\n");          break;      case 8:          printf(" 良好\n");          break;      case 7:          printf(" 中等\n");          break;      case 6:          printf(" 及格\n");          break;      default:          printf(" 不及格\n");      }  } |

Q480.(10分)编程求输入的整数的绝对值。

\*\*输入格式要求："%d%d" 提示信息："input the value of x:"

\*\*输出格式要求："|x|=%d\n"

程序运行示例如下：

input the value of x:68

|x|=68

### 参考答案

|  |
| --- |
| #include <stdio.h>    int main()  {      int x, y;        printf("input the value of x:");      scanf("%d", &x);        y = x;      if (x < 0)          y = -x;      printf("|x|=%d\n", y);        return 0;  } |

Q332.(10分)\*猜数游戏。由计算机“想”一个数请人猜，如果人猜对了，则结束游戏，否则计算机给出提示，告诉人所猜的数是太大还是太小，直到人猜对为止。计算机记录人猜的次数，以此可以反映出猜数者猜的水平。

\*\*输入格式要求："%d" 提示信息："guess the magic number:"

\*\*输出格式要求："("\*\*\*\*Wrong\*\*\*\*too hight\n" "\*\*\*\*Wrong\*\*\*\*too low\n" "\*\*\*\*Right\*\*\*\*\n" "guess counter is %d\n"

程序运行示例如下：

guess the magic number:50

\*\*\*\*Wrong\*\*\*\*too hight

guess the magic number:30

\*\*\*\*Wrong\*\*\*\*too low

guess the magic number:40

\*\*\*\*Right\*\*\*\*

guess counter is 3

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  main()  {      int magic;      int guess;      int counter;      magic=rand();      guess=magic-1;      counter=0;      while(magic!=guess)      {          printf("guess the magic number:");          scanf("%d",&guess);          counter++;          if(guess>magic)              printf("\*\*\*\*Wrong\*\*\*\*too hight\n");          else if(guess<magic)              printf("\*\*\*\*Wrong\*\*\*\*too low\n");      }      printf("\*\*\*\*Right\*\*\*\*\n");      printf("guess counter is %d\n",counter);  } |

Q116.(20分)编程计算2×3阶矩阵A和3×2阶矩阵B之积C。

矩阵相乘的基本方法是：

矩阵A的第i行的所有元素同矩阵B第j列的元素对应相乘，

并把相乘的结果相加，最终得到的值就是矩阵C的第i行第j列的值。

要求：

(1)从键盘分别输入矩阵A和B，

输出乘积矩阵C

(2)

\*\*输入提示信息为：

输入矩阵A之前提示："Input 2\*3 matrix a:\n"

输入矩阵B之前提示："Input 3\*2 matrix b:\n"

\*\*输入矩阵中每个值的格式为："%d"

\*\*输出格式为：

输出矩阵C之前提示："Results:\n"

输出矩阵C中每个值的格式："%6d"

输出矩阵C各行结束时换行

例如：

Input 2\*3 matrix a:

1 2 3

0 1 2

Input 3\*2 matrix b:

1 -2

2 3

1 3

Results:

8 13

4 9

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define  ROW 2  #define  COL 3  void main()  {      int a[ROW][COL], b[COL][ROW], c[ROW][ROW], i, j,k;        printf("Input 2\*3 matrix a:\n");      for (i=0; i<ROW ;i++)//1      {          for (j=0; j<COL; j++)//1          {              scanf("%d", &a[i][j]);//1          }      }      printf("Input 3\*2 matrix b:\n");      for (i=0; i<COL; i++)//1      {          for (j=0; j<ROW; j++) //1          {              scanf("%d",  &b[i][j]);//1          }      }      for (i=0; i<ROW; i++)//1      {          for (j=0; j<ROW; j++)//1          {              c[i][j] =   0 ;//2              for (k=0; k<COL; k++)//2              {                  c[i][j] = c[i][j]+a[i][k]\*b[k][j] ;//4              }          }      }      printf("Results:\n");      for (i=0; i<ROW; i++)//1      {          for (j=0; j<ROW; j++) //1          {              printf("%6d", c[i][j]);//1          }          printf("\n") ;//1      }  } |

Q478.(10分)判断二维空间中的点，是否在圆内（输出：该点在圆内、该点在圆上、该点在圆外）。

\*\*输入格式要求："%f,%f" 提示信息："请输入圆的圆心坐标：" "请输入圆的半径：" "请输入要判断的点的坐标(x,y)："

\*\*输出格式要求："该点在圆内\n" "该点不在圆内\n" "该点在圆上\n"

程序示例运行如下：

请输入圆的圆心坐标：3.5,4.5

请输入圆的半径：6

请输入要判断的点的坐标(x,y)：5.5,7.8

该点在圆内

### 参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>    int main()  {      float x, y;      float ox, oy, radius, d;        printf("请输入圆的圆心坐标：");      scanf("%f,%f", &ox, &oy);      printf("请输入圆的半径：");      scanf("%f", &radius);        printf("请输入要判断的点的坐标(x,y)：");      scanf("%f,%f", &x, &y);        d = sqrt((ox - x ) \* (ox - x) + (oy - y) \* (oy - y));      if (d < radius)      {          printf("该点在圆内\n");      }      else if (d > radius)      {          printf("该点不在圆内\n");      }      else      {          printf("该点在圆上\n");      }        return 0;  } |

Q1162.(10分)

|  |
| --- |
| 在下面给出的程序中，共有5处错误（包括语法错误和逻辑错误），请找出其中的错误，并改正之。本程序的功能是对a数组中的元素按从大到小的顺序排序。  #include <stdio.h>  #define N 10;  main()  {      int a[N]={2,17,8,3,24,53,82,1,29,101};      int i,j,k,t;      for(i=0;i<9;i++)      {       k=i;              for(j=i+1;j<10;j++)                  if(a[j]<a[k])    k=j;              if(k=i)              {                  t=a[j];                  a[j]=a[i];                  a[i]=t;                          }      }          for(i=0;i<9;i++)                  printf("%d ",a[i]);  } |

### 参考答案

|  |
| --- |
| #include <stdio.h>  #define N 10  main()  {      int a[N]={2,17,8,3,24,53,82,1,29,101};      int i,j,k,t;      for(i=0;i<9;i++)      {       k=i;              for(j=i+1;j<10;j++)                  if(a[j]>a[k])    k=j;              if(k!=i)              {                  t=a[k];                  a[k]=a[i];                  a[i]=t;              }      }      for(i=0;i<9;i++)              printf("%d ",a[i]);  } |

Q1740.(10分)编程打印以下图案。

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

参考答案

#include <stdio.h>

main()

{

int i , j, k ;

for (i = 1; i <= 4; i++) // 控制行数

{

for (j = 1; j <= 8 - i; j++) // 随着行数的增加，输出递减数目的空格

{

printf(" ");

}

for (k = 1; k <= (2 \* i - 1); k++)

{

printf("\*");

}

printf("\n");

}

}

Q1728.(10分)编写一个程序，打印所有的“水仙花数”。

所谓“水仙花数”，是指一个三位数，

其各位数字的立方和等于该数本身。

例如，153是“水仙花数”，因为153=13+33+53。

\*\*要求输入提示信息为：无输入提示信息和输入数据

\*\*要求输出格式为："%d\n"

参考答案

|  |
| --- |
| #include<stdio.h>  main()  {    int i,j,k,n;    for(n=100;n<1000;n++)    {      i=n/100;//1      j=(n-i\*100)/10;//1      k=n-i\*100-j\*10;//1      if(i\*i\*i+j\*j\*j+k\*k\*k==n)        printf("%d\n",n);      }    } |

Q547.(10分)从键盘输入一个字符串a，将字符串a复制到字符串b中，再输出字符串b，即编程实现字符串处理函数strcpy()的功能，但要求不能使用字符串处理函数strcpy()。

\*\*输入格式要求："%s" 提示信息："Input a string:"

\*\*输出格式要求："The copy is:"

程序的运行示例如下：

Input a string:Hello China

The copy is:Hello China

#include  <stdio.h>

#define N 80

void  MyStrcpy(char dstStr[], char srcStr[]);

int main()

{

    char  a[N], b[N];

    printf("Input a string:");

    gets(a);                      /\* 输入字符串 \*/

    MyStrcpy(b, a);              /\* 将字符数组a中的字符串拷贝到b中 \*/

    printf("The copy is:");

    puts(b);                       /\* 输出复制后的字符串 \*/

    return 0;

}

/\* 函数功能：用字符数组作为函数参数，字符串拷贝 \*/

void  MyStrcpy(char dstStr[], char srcStr[])

{

    int  i = 0;                   /\* 数组下标初始化为0 \*/

    while (srcStr[i] != '\0')  /\* 若当前取出的字符不是字符串结束标志 \*/

    {

        dstStr[i] = srcStr[i];/\* 复制字符 \*/

        i++;                      /\* 移动下标 \*/

    }

    dstStr[i] = '\0';            /\* 在字符串dstStr的末尾添加字符串结束标志 \*/

}

Q429.(10分)满足特异条件的数列。输入m和n（20≥m≥n≥0），求出满足以下方程式的正整数数列i1,i2,…,in，使得i1+i2+…+in=m，且i1≥i2≥…≥in。例如：

当n=4，m=8时，将得到如下5个数列：

5 1 1 1 4 2 1 1 3 3 1 1 3 2 2 1 2 2 2 2

\*\*输入格式要求："%d" 提示信息："Please enter requried terms (<=10):"

\*\*输出格式要求："There are following possible series:\n" "[%d]:"

参考答案

|  |
| --- |
| #define NUM 10  #include <stdio.h>  int i[NUM];  main()  { int sum, n,total,k,flag,count=0;    printf("Please enter requried terms (<=10):");    scanf("%d",&n);    printf("                             their sum:");    scanf("%d",&total);    sum=0;    k=n;    i[n]=1;    printf("There are following possible series:\n");    while(1)    {      if(sum+i[k]<total)          if(k<=1)          {i[1]=total-sum;flag=1;}          else          {sum+=i[k--];           i[k]=i[k+1];           continue;           }      else if(sum+i[k]>total||k!=1)          {sum-=i[++k];          flag=0;          }          else          flag=1;      if(flag)      {          printf("[%d]:",++count);          for(flag=1;flag<=n;++flag)              printf("%d",i[flag]);          printf("\n");      }      if(++k>n)          break;      sum-=i[k];      i[k]++;      }  } |

Q1741.(10分)

按如下公式递归计算矩阵行列式的值：

\*\*输入格式要求："%d" "%lf" 提示信息："Please enter matrix size n(1<=n<%d):"  "Please input matrix line by line:\n"  "matrix a:\n"  
\*\*输出格式要求："Submatrix:\n"  "DValue of the Submatrix is %6.1f\n"  "%6.1f"  "result = %f\n"

程序运行结果如下：

Please enter matrix size n(1<=n<20):3  
Please input matrix line by line:

1 2 3  
4 5 7  
8 9 11  
matrix a:  
   1.0     2.0     3.0  
   4.0     5.0     7.0  
   8.0     9.0    11.0

Submatrix:  
   5.0     7.0  
   9.0    11.0  
DValue of the Submatrix is   -8.0  
Submatrix:  
   4.0     7.0  
   8.0    11.0  
DValue of the Submatrix is  -12.0  
Submatrix:  
   4.0     5.0  
   8.0     9.0  
DValue of the Submatrix is   -4.0  
result = 4.000000

参考答案

|  |
| --- |
| #include <math.h>  #include <stdio.h>    #define  CONST 1e-6  #define  SIZE 20    void InputMatrix (double a[][SIZE], int n);  double DeterminantValue(double a[][SIZE], int n);  void SubMatrix(double a[][SIZE], double b[][SIZE], int n, int row, int col);  void PrintMatrix(double a[][SIZE], int n);  int main(void)  {      double  a[SIZE][SIZE];      int     n;      double  result;      printf("Please enter matrix size n(1<=n<%d):", SIZE);      scanf("%d", &n);      printf("Please input matrix line by line:\n");      InputMatrix(a, n);      printf("matrix a:\n");      PrintMatrix(a, n);      printf("\n");      result = DeterminantValue(a, n);      printf("result = %f\n", result);      return 0;  }  //  函数功能： 输入一个n×n矩阵的元素  void InputMatrix (double a[][SIZE], int n)  {      int i, j;      for (i = 0; i < n; i++)      {          for (j = 0; j < n; j++)          {              scanf("%lf", &a[i][j]);          }      }  }  //  函数功能： 计算n×n矩阵的行列式的值  double DeterminantValue(double a[][SIZE], int n)  {      int    i = 0, j = 0;      double temp, result, b[SIZE][SIZE];      if (n == 1)      {          result = a[0][0];      }      else if (n == 2)      {          result = a[0][0] \* a[1][1] - a[0][1] \* a[1][0];      }      else      {          result = 0.0;          for (j = 0; j < n; j++)          {              SubMatrix(a, b, n, i, j);              printf("Submatrix:\n");              PrintMatrix(b, n - 1);              temp = DeterminantValue(b, n - 1);              result += pow(-1, i + j) \* a[0][j] \* temp;              printf("DValue of the Submatrix is %6.1f\n", temp);          }      }      return result;  }  //  函数功能： 计算n×n矩阵a中第row行col列元素的(n-1)×(n-1)子矩阵b  void SubMatrix(double a[][SIZE], double b[][SIZE], int n, int row,                 int col)  {      int i, j, ii = 0, jj = 0;      for (i = 0; i < n; i++)      {          jj = 0;          for (j = 0; j < n; j++)          {              if (i != row && j != col)              {                  b[ii][jj] = a[i][j];                  jj++;              }          }          if (i != row && j != col)          {              ii++;          }      }  }  //  函数功能： 输出一个n×n矩阵的元素  void PrintMatrix(double a[][SIZE], int n)  {      int i, j;      for (i = 0; i < n; i++)      {          for (j = 0; j < n; j++)          {              printf("%6.1f\t", a[i][j]);          }          printf("\n");      }  } |

Q608.(10分)请使用short int型的变量计算两个7位十进制整数的和，要求使用尽可能少的变量。

\*\*输入格式要求："a = %3hd%4hd, b = %3hd%4hd"

\*\*输出格式要求："a + b = %d"

程序运行示例如下：

a = 1234567,b = 7654321

输出

a + b = 8888888

参考答案

|  |
| --- |
| #include <stdio.h>  int main()  {      short int d, m, c, n;      scanf("a = %3hd%4hd, b = %3hd%4hd", &d, &m, &c, &n);      printf("a + b = %d", (d + c) \* 10000 + m + n);      return 0;  } |

Q1294.(10分)

|  |
| --- |
| 程序改错。编写一个函数Inverse()，实现将字符数组中的字符串逆序存放的功能。（4个错误）  #include<string.h>  #include<stdio.h>    #define ARR\_SIZE = 80;    void Inverse(char str[]);    main()  {      char a[ARR\_SIZE] ;        printf("Please enter a string: ");      gets(a);      Inverse(char a[]);      printf("The inversed string is: ");      puts(a);  }    void Inverse(char str[])  {      int   len, i, j;      char  \*temp;        len = strlen(str);      for (i=0; j=len-1; i<j; i++; j--)        {      temp = str[i];          str[i] = str[j];          str[j] = temp;      }  } |

参考答案

|  |
| --- |
| #include<string.h>  #include<stdio.h>    #define ARR\_SIZE 80    void Inverse(char str[]);    main()  {      char a[ARR\_SIZE] ;        printf("Please enter a string: ");      gets(a);      Inverse(a);      printf("The inversed string is: ");      puts(a);  }    void Inverse(char str[])  {      int   len, i, j;      char  temp;        len = strlen(str);      for (i=0, j=len-1; i<j; i++, j--)      {      temp = str[i];          str[i] = str[j];          str[j] = temp;      }  } |

Q781.(5分)改错题：下面程序用于输入三角形的三条边a,b,c，

判断它们能否构成三角形，若能则指出是何种三角形：

等腰三角形、直角三角形、等腰直角三角形，一般三角形。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

\*\*输入输出格式要求：不要改变以下代码的输入输出格式。

|  |
| --- |
| #include <stdio.h>  main()  {      float   a, b, c;      int flag;      scanf("%f,%f,%f", &a, &b , &c);      if (a + b > c && b + c > a && a + c > b)      {          if (a = b | b = c | c = a)          {              printf("等腰");              flag = 0;          }          else if (a \* a + b\*b = c \* c | a \* a + c\*c = b \* b | c \* c + b\*b = a \* a)          {              printf("直角");              flag = 0;          }          else if (!flag)          {              printf("一般");          }          printf("三角形\n");      }      else      {          printf("不是三角形\n");      }  } |

Q781.(5分)改错题：下面程序用于输入三角形的三条边a,b,c，

判断它们能否构成三角形，若能则指出是何种三角形：

等腰三角形、直角三角形、等腰直角三角形，一般三角形。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

\*\*输入输出格式要求：不要改变以下代码的输入输出格式。

|  |
| --- |
| #include <stdio.h>  main()  {      float   a, b, c;      int flag;      scanf("%f,%f,%f", &a, &b , &c);      if (a + b > c && b + c > a && a + c > b)      {          if (a = b | b = c | c = a)          {              printf("等腰");              flag = 0;          }          else if (a \* a + b\*b = c \* c | a \* a + c\*c = b \* b | c \* c + b\*b = a \* a)          {              printf("直角");              flag = 0;          }          else if (!flag)          {              printf("一般");          }          printf("三角形\n");      }      else      {          printf("不是三角形\n");      }  } |

参考答案

|  |
| --- |
| #include <stdio.h>  #include <math.h>  main()  {      float a, b, c;      int flag = 0;        scanf("%f,%f,%f", &a, &b , &c);        if ((a + b) > c && ((b + c) > a) && ((a + c) > b))      {          if ((fabs(a - b) <= 0.1  || fabs(b - c) <= 0.1  || fabs(c - a) <= 0.1) && (fabs(a \* a + b \* b - c \* c) <= 0.1 || fabs(a \* a + c \* c - b \* b) <= 0.1 || fabs(c \* c + b \* b - a \* a) <= 0.1))          {              printf("等腰直角");              flag = 1;          }          else if (fabs(a - b) <= 0.1  || fabs(b - c) <= 0.1  || fabs(c - a) <= 0.1)          {              printf("等腰");              flag = 1;          }          else if (fabs(a \* a + b \* b - c \* c) <= 0.1 || fabs(a \* a + c \* c - b \* b) <= 0.1 || fabs(c \* c + b \* b - a \* a) <= 0.1)          {              printf("直角");              flag = 1;          }          else          {              printf("一般");          }          printf("三角形\n");      }      else      {          printf("不是三角形\n");      }  } |

Q557.(20分)设有a[n],b[n]。求a[0]\*b[0]+a[1]\*b[1]+...+a[n-1]\*b[n-1] 。其中n，及数组a和b的元素值由键盘输入。

程序运行示例如下：

Input n

3

Input array a

1 2 3

Input array b

4 5 6

sum=32.000000

#include <stdio.h>

float product(float\*pa, float \*pb, int n)

{

int i;

float s;

for (i = 0, s = 0; i < n; i++)

{

s = s + (\*pa \* \*pb);

pa++;

pb++;

}

return s;

}

int main(int argc, char\* argv[])

{

float \*pa, \*pb, sum;

int i, n;

printf("Input n\n");

scanf("%d", &n);

pa = (float\*) malloc(n \* sizeof(float));

pb = (float\*) malloc(n \* sizeof(float));

printf("Input array a\n");

for (i = 0; i < n; i++)

{

scanf("%f", pa);

pa++;

}

printf("Input array b\n");

for (i = 0; i < n; i++)

{

scanf("%f", pb);

pb++;

}

pa = pa - n;

pb = pb - n;

sum = product(pa, pb, n);

printf("\nsum=%f\n", sum);

return 0;

}

Q525.(10分)编程从键盘输入n值（10≥n≥3），然后计算并输出1! + 2! + 3! + … + n!。

\*\*输入格式要求："%d" 提示信息："Input n:"

\*\*输出格式要求："1!+2!+…+%d! = %ld\n"

程序运行示例如下：

Input n:10

1!+2!+…+10! = 4037913

#include <stdio.h>

main()

{

int i, j, n;

long term, sum = 0; /\* 累加求和变量sum初始化为0 \*/

printf("Input n:");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

term = 1; /\* 累乘求积变量term赋初值为1 \*/

for (j = 1; j <= i; j++)

{

term = term \* j; /\* 累乘求积 \*/

}

sum = sum + term; /\* 累加求和 \*/

}

printf("1!+2!+…+%d! = %ld\n", n, sum);

}

Q1383.(10分)以下程序的功能是从键盘输入10个整型数据，放入数组a 中，求其最大值、最小值及其所在元素的下标位置并输出。

#include <stdio.h>

main()

{ int a[10], n, max, min, maxPos, minPos;

for (n=0; n<10; n++)

{

scanf("%d", &a[n]);

}

max = min = a[0]; //应初始化为数组中的某一元素

maxPos = minPos = 0;

for (n=1; n<10; n++)

{

if (a[n] > max)

{

max = a[n];

maxPos = n;

}

else if (a[n] < min)

{

min = a[n];

minPos = n;

}

}

printf("max=%d, pos=%d\n",max, maxPos);

printf("min=%d, pos=%d\n",min, minPos);

}

Q361.(10分)编写一个C函数，实现保留小数点后第N-1位，从第N位四舍五入。用该函数对数组a[5]中的各元素从小数点后第2位开始四舍五入。

a[5]={2.33,2.56,2.65,2.66,2.30}

\*\*输出格式要求："Old array:\n" "a[%d]=%.3f\t" "\nnew array:\n" "a[%d]=%.2f\t"

程序运行示例如下：

Old array:

a[0]=2.330 a[1]=2.560 a[2]=2.650 a[3]=2.660 a[4]=2.300

new array:

a[0]=2.30 a[1]=2.60 a[2]=2.70 a[3]=2.70 a[4]=2.30

#include <stdio.h>

#include <math.h>

float a[5] = {2.33, 2.56, 2.65, 2.66, 2.30};

float t(float x,int n)

{

float x1, x2;

int i;

for (x1 = x, i = 0; i < n - 1; i++)

x1 \*= 10.0;

x2 = floor((double)x1);

x1 = floor((double)(x1 - x2) \* 10);

if (x1 >= 5)

x2 += 1;

for (i = 0; i < n - 1; i++)

x2 /= 10.0;

return x2;

}

main()

{

float t();

int i;

printf("Old array:\n");

for (i = 0; i < 5; i++)

printf("a[%d]=%.3f\t", i, a[i]);

printf("\nnew array:\n");

for (i = 0; i < 5; i++)

printf("a[%d]=%.2f\t", i, t(a[i], 2));

}

Q345.(10分)求两个整数中的较大值。

\*\*输入格式要求："%d%d" 提示信息："input a and b="

\*\*输出格式要求："The max of a and b :%d\n"

程序运行示例如下：

input a and b=5 6

The max of a and b :6

#include <stdio.h>

main()

{

int a,b,max;

printf("input a and b=");

scanf("%d%d",&a,&b);

if(a>b) max=a;

else max=b;

printf("The max of a and b :%d\n",max);

}

Q1309.(10分)用二分法求方程2x^3-4x^2+3x-6=0在(-10, 10)之间的根。

\*\*输入格式要求："%f,%f" 提示信息："请输入x1，x2的值："

\*\*输出格式要求："方程的根=%6.2f\n"

程序的运行示例如下：

请输入x1，x2的值：-10,10

方程的根= 2.00

#include <stdio.h>

#include <math.h>

int main()

{

float x0, x1, x2, fx0, fx1, fx2;

do

{

printf("请输入x1，x2的值：");

scanf("%f,%f", &x1, &x2);

fx1 = x1 \* ((2 \* x1 - 4) \* x1 + 3) - 6;

fx2 = x2 \* ((2 \* x2 - 4) \* x2 + 3) - 6;

}

while (fx1 \* fx2 > 0);

do

{

x0 = (x1 + x2) / 2;

fx0 = x0 \* ((2 \* x0 - 4) \* x0 + 3) - 6;

if (fx0 \* fx1 < 0)

{

x2 = x0;

fx2 = fx0;

}

else

{

x1 = x0;

fx1 = fx0;

}

}

while (fabs(fx0) >= 1e-5);

printf("方程的根=%6.2f\n", x0);

return 0;

}

Q592.(10分)编写一个程序，将一个字符串s2插入到字符串s1中，其起始插入位置为n。

\*\*输入格式要求："%d" 提示信息："main string:" "sub string:" "site of begining:(<=%d)"

\*\*输出格式要求："After instert:%s\n"

程序运行示例如下：

main string:hello,world!

sub string:c

site of begining:(<=12)6

After instert:hello,c world!

#include <stdio.h>

#include <string.h>

#define N 100

int main()

{

int n, i, j, k, len;

char s1[N], s2[N], s3[2 \* N];

printf("main string:");

gets(s1);

printf("sub string:");

gets(s2);

len = strlen(s1);

do

{

printf("site of begining:(<=%d)", len);

scanf("%d", &n);

}

while (n > len);

for (i = 0; i < n; i++)

{

s3[i] = s1[i];

}

for (j = 0; s2[j] != '\0'; j++)

{

s3[i + j] = s2[j];

}

for (k = n; s1[k] != '\0'; k++)

{

s3[j + k] = s1[k];

}

s3[j + k] = '\0';

printf("After instert:%s\n", s3);

return 0;

}

Q342.(10分)输入10个整数，求其中正数的个数及平均值，精确到小数点后两位。

\*\*输入格式要求："%d" 提示信息："Input integer:"

\*\*输出格式要求："Plus number:%d,average value:%.2f" "Plus number:0,average value: 0"

程序运行示例如下：

Input integer:1

Input integer:2

Input integer:-9

Input integer:8

Input integer:-3

Input integer:3

Input integer:81

Input integer:3

Input integer:0

Input integer:5

Plus number:7,average value:14.71

#include <stdio.h>

main()

{

int i,count,j,sum;

sum=count=0;

for(i=1;i<=10;i++)

{

printf("Input integer:");

scanf("%d",&j);

if(j<=0) continue;

count++;

sum+=j;

}

if(count)

printf("Plus number:%d,average value:%.2f",count,1.0\*sum/count);

else printf("Plus number:0,average value: 0");

}

Q429.(10分)满足特异条件的数列。输入m和n（20≥m≥n≥0），求出满足以下方程式的正整数数列i1,i2,…,in，使得i1+i2+…+in=m，且i1≥i2≥…≥in。例如：

当n=4，m=8时，将得到如下5个数列：

5 1 1 1 4 2 1 1 3 3 1 1 3 2 2 1 2 2 2 2

\*\*输入格式要求："%d" 提示信息："Please enter requried terms (<=10):"

\*\*输出格式要求："There are following possible series:\n" "[%d]:"

#define NUM 10

#include <stdio.h>

int i[NUM];

main()

{ int sum, n,total,k,flag,count=0;

printf("Please enter requried terms (<=10):");

scanf("%d",&n);

printf(" their sum:");

scanf("%d",&total);

sum=0;

k=n;

i[n]=1;

printf("There are following possible series:\n");

while(1)

{

if(sum+i[k]<total)

if(k<=1)

{i[1]=total-sum;flag=1;}

else

{sum+=i[k--];

i[k]=i[k+1];

continue;

}

else if(sum+i[k]>total||k!=1)

{sum-=i[++k];

flag=0;

}

else

flag=1;

if(flag)

{

printf("[%d]:",++count);

for(flag=1;flag<=n;++flag)

printf("%d",i[flag]);

printf("\n");

}

if(++k>n)

break;

sum-=i[k];

i[k]++;

}

}

Q330.(10分)输入年份year和月month，求该月有多少天。判断是否为闰年，可用如下C语言表达式：year%4==0&&year%100!=0||year%4==0。若表达式成立，则year为闰年；否则，表达式不成立，为平年。

\*\*输入格式要求："%d%d" 提示信息："Enter year and month:"

\*\*输出格式要求："%d,%d is %d days\n"

程序运行示例如下：

Enter year and month:2012 11

2012,11 is 30 days

#include <stdio.h>

main()

{

int year,month,days;

printf("Enter year and month:");

scanf("%d%d",&year,&month);

switch(month)

{

case 1: case 3: case 5: case 7:

case 8: case 10: case 12:

days=31;break;

case 4: case 6: case 9: case 11:

days=30;break;

case 2:

if(year%4==0&&year%100!=0||year%4==0)

days=29;

else days=28;

break;

default:

printf("Input error\n");

days=0;

}

if(days!=0)

printf("%d,%d is %d days\n",year,month,days);

}

Q511.(10分)从键盘输入一个大写英文字母，将其转换为小写英文字母后，将转换后的小写英文字母及其十进制的ASCII码值显示到屏幕上。

\*\*输入格式要求：提示信息："Press a key and then press Enter:"

\*\*输出格式要求："%c, %d\n"

程序运行示例如下：

Press a key and then press Enter:B

b, 98

#include <stdio.h>

main()

{

char ch;

printf("Press a key and then press Enter:");

ch = getchar();

ch = ch + 32;

printf("%c, %d\n", ch, ch); /\* 分别输出变量ch中的字符及其ASCII码值 \*/

}

Q333.(10分)从键盘中读入一系列字符，原样输出到屏幕上，直到输入字母a时才停止。

程序运行示例1如下：

dbca （输入）

dbc （输出）

程序运行示例2如下：

first （输入）

first （输出）

second （输入）

second （输出）

a （输入）

#include <stdio.h>

main()

{

char ch;

while ((ch = getchar()) != 'a')

{

putchar(ch);

}

}

Q454.(10分)已知xyz+yzz=532，其中x，y，z都是数字（0-9），编写一个程序求出x，y，z分别代表什么数字。

\*\*输出格式要求："x=%d,y=%d,z=%d"

#include <stdio.h>

int main()

{

int x, y, z;

for (x = 1; x < 6; x++)

{

for (y = 0; y < 4; y++)

{

for (z = 0; z < 4; z++)

{

if ((x \* 100 + y \* 10 + z) +

(y \* 100 + z \* 10 + z) == 532)

{

printf("x=%d,y=%d,z=%d", x, y, z);

}

}

}

}

}

Q386.(15分)编程计算：3+33+333+...+33...3(n个3)的值，n值由键盘输入。

\*\*\*输入提示信息\*\*\*："Input n:\n"

\*\*\*输入数据格式\*\*\*："%d"

\*\*\*输出数据格式\*\*\*："sum=%ld\n"

注：不允许使用goto语句

#include<stdio.h>

main()

{

int i, n;

long term, sum = 0; //1

printf("Input n:\n");

scanf("%d", &n); //1

for (i = 1, term = 3; i <= n; i++) //4

{

sum = sum + term; //2

term = term \* 10 + 3; //2

}

printf("sum=%ld\n", sum); //1

}

Q388.(15分)请按给定要求编程计算方阵的主对角线之和。其中方阵的输入请在Input函数实现，计算主对角之和请在Compute函数实现，计算结果请在主函数打印，且给定的方阵最大为10x10，方阵实际大小请在主函数读入。Input函数和Compute函数的函数原型如下：

void Input(int a[][10],int n);

int Compute(int a[][10],int n);

\*\*\*输入提示信息\*\*\*："Input n(n<=10):\n"

\*\*\*输入数据格式\*\*\*："%d"

\*\*\*输入提示信息\*\*\*："Input matrix:\n"

\*\*\*输入数据格式\*\*\*："%d"

\*\*\*输出数据格式\*\*\*："sum=%d\n"

注：

1、请按行读入数据

2、请按给定的函数原型编写程序，否则即使程序运行结果正确，也不给分。

#include<stdio.h>

void Input(int a[][10], int n);

int Compute(int a[][10], int n);

main()

{

int n, a[10][10];

int sum;

printf("Input n(n<=10):\n");

scanf("%d", &n); //1

Input(a, n); //2

sum = Compute(a, n); //2

printf("sum=%d\n", sum); //1

}

void Input(int a[][10], int n)

{

int i, j;

printf("Input matrix:\n");

for (i = 0; i < n; i++) //1

{

for (j = 0; j < n; j++) //1

{

scanf("%d", &a[i][j]); //1

}

}

}

int Compute(int a[][10], int n)

{

int i;

int sum = 0;

for (i = 0; i < n; i++) //1

{

sum = sum + a[i][i]; //1

}

return sum; //1

}

Q1513.(20分)从键盘输入一个小写英文字母,将其转换成大写英文字母后,输出大写英文字母及其对应的十进制ASCII码值.

\*\*输入格式要求：无任何信息提示；

\*\*输出格式要求："%c,%d\n"

#include<stdio.h>

int main()

{

char ch;

ch = getchar();

ch = ch - 32;

printf("%c,%d\n", ch, ch);

return 0;

}

Q444.(10分)选美比赛

在选美大奖赛的半决赛现场，有一批选手参加比赛，比赛的规则是最后得分越高，名次

越低。当半决赛结束时，要在现场按照选手的出场顺序宣布最后得分和最后名次，获得

相同分数的选手具有相同的名次，名次连续编号，不用考虑同名次的选手人数。例如：

选手序号： 1，2，3，4，5，6，7

选手得分： 5，3，4，7，3，5，6

则输出名次为：3，1，2，5，1，3，4

请编程帮助大奖赛组委会完成半决赛的评分排名工作。

\*\*输出格式要求：" Player\_NO Score Rank\n" " %3d %4d %4d\n"

程序运行示例如下：

Player\_NO Score Rank

1 5 3

2 3 1

3 4 2

4 7 5

5 3 1

6 5 3

7 6 4

#include <stdio.h>

#define NUM 7

int a[NUM+1]={0,5,3,4,7,3,5,6};

int m[NUM+1],l[NUM+1];

main()

{

int i,n,smallest,num,k,j;

num=1;

for(i=1;i<=NUM;i++)

if(m[i]==0)

{

smallest=a[i];

k=1;

l[k]=i;

for(j=i+1;j<=NUM;j++)

if(m[j]==0)

if(a[j]<smallest)

{

smallest=a[j];

k=0;

l[++k]=j;

}

else if(a[j]==smallest)

l[++k]=j;

for(j=1;j<=k;j++)

m[l[j]]=num;

num++;

i=0;

}

printf(" Player\_NO Score Rank\n");

for(j=1;j<=NUM;j++)

printf(" %3d %4d %4d\n",j,a[j],m[j]);

}

Q186.(15分)两个正整数的最大公约数是能够整除这两个整数的最大整数。

请按如下函数原型编写计算最大公约数的函数Gcd()。

int Gcd(int a, int b);

在主函数中调用该函数计算，并输出从键盘任意输入的两整数的最大公约数。

\*\*要求输入提示信息为: "Input a,b:\n"

\*\*要求输入格式为: "%d,%d"

\*\*要求输出格式为

如果a或b不是正整数，输出: "Input number should be positive!\n"

否则输出: "Greatest Common Divisor of %d and %d is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include <stdio.h>

int Gcd(int a, int b);

main()

{

int a, b, c;

printf("Input a,b:\n");

scanf("%d,%d", &a, &b); //1

c = Gcd(a, b); //1

if (c != -1) //1

{

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c); //1

}

else

{

printf("Input number should be positive!\n"); //1

}

}

int Gcd(int a, int b) //1

{

int i, t;

if (a <= 0 || b <= 0) //1

{

return -1; //1

}

t = a < b ? a : b; //2

for (i = t; i > 0; i--)//2

{

if (a % i == 0 && b % i == 0)//1

return i; //1

}

return 1; //1

}

Q1117.(10分)函数fun功能：由小到大输出两实数。

#include <stdio.h>

fun()

{

float t;

float a, b;

scanf("%f%f", &a, &b);

if (a < b)

{

t = a;

a = b;

b = t;

}

printf("%5.2f,%5.2f\n", a, b);

}

main()

{

fun();

}

Q155.(15分)从键盘任意输入一个整数n（如果n<0，则要求提示"Input Error!\n"）,用函数编程计算这个整数的阶乘值。

要求按如下函数原型编程。

long fac(int n);

在主函数中调用fac计算n!。

要求必须用函数编程，否则不给分。

\*\*输入提示信息格式要求为："Input an integer:\n"；

\*\*输出格式要求为"%d! = %ld\n"，其中第一个整数为输入数值，第二个整数位其阶乘。

#include <stdio.h>

long fac(int n);

main()

{

int n;

long res;

printf("Input an integer:\n");

scanf("%d",&n); //1

if (n < 0) printf("Input Error!\n");

else

{

res = fac(n); //1

printf("%d! = %ld\n", n, res);

}

}

long fac(int n)

{

int i;

long res = 1; //1

for(i=1; i<=n; i++) //1

{

res \*= i; //1

}

return res;

}

Q478.(10分)判断二维空间中的点，是否在圆内（输出：该点在圆内、该点在圆上、该点在圆外）。

\*\*输入格式要求："%f,%f" 提示信息："请输入圆的圆心坐标：" "请输入圆的半径：" "请输入要判断的点的坐标(x,y)："

\*\*输出格式要求："该点在圆内\n" "该点不在圆内\n" "该点在圆上\n"

程序示例运行如下：

请输入圆的圆心坐标：3.5,4.5

请输入圆的半径：6

请输入要判断的点的坐标(x,y)：5.5,7.8

该点在圆内

#include <stdio.h>

#include <math.h>

int main()

{

float x, y;

float ox, oy, radius, d;

printf("请输入圆的圆心坐标：");

scanf("%f,%f", &ox, &oy);

printf("请输入圆的半径：");

scanf("%f", &radius);

printf("请输入要判断的点的坐标(x,y)：");

scanf("%f,%f", &x, &y);

d = sqrt((ox - x ) \* (ox - x) + (oy - y) \* (oy - y));

if (d < radius)

{

printf("该点在圆内\n");

}

else if (d > radius)

{

printf("该点不在圆内\n");

}

else

{

printf("该点在圆上\n");

}

return 0;

}

Q363.(10分)给定一维数组a，求a中各元素的平均值。

a[10]={1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9}

\*\*输出格式要求："The average of array is %f\n"

程序运行示例如下：

The average of array is 1.450000

main()

{

float average();

static float a[10]={1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9};

printf("The average of array is %f\n",average(a,10));

}

float average(a,n)

int n;

float a[];

{

int i;

float sum=0;

for(i=0;i<n;i++)

sum+=a[i];

return sum/n;

}

Q195.(30分)编程计算: 1!+3!+5!+...+(2n-1)!，要求阶乘计算调用fun函数实现，

数据输入及打印结果在主函数实现。阶乘计算fun函数原型为:

long fun(int m);

参数说明:

参 数: m是要进行阶乘计算的数；

返回值: 数m的阶乘

\*\*\*\*输入提示信息格式: "Input n:\n"

\*\*\*\*输入数据格式要求: %d

\*\*\*\*输出数据格式要求: "sum=%ld\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include<stdio.h>

long fun(int m);

main()

{

int i,n;

long sum=0,p; //1

printf("Input n:\n");

scanf("%d",&n); //1

for(i=1;i<=2\*n-1;i+=2) //2

{

p=fun(i); //2

sum=sum+p; //1

}

printf("sum=%ld\n",sum); //1

}

long fun(int m)

{

long p=1; //1

int i;

for(i=1;i<=m;i++) //2

{

p=p\*i; //1

}

return p; //2

}

Q586.(10分)编写一个程序，输入年份和月份，判断该年是否是闰年，并根据给出的月份判断是什么季节和该月有多少天？（闰年的条件是年份能被4整除但不能被100整除，或者能被400整除；规定3~5月为春季，6~8月为夏季，9~11月为秋季，1、2和12月为冬季）。

\*\*输入格式要求："%d,%d" 提示信息："Please enter year,month:"

\*\*输出格式要求："%d is leap year\n" "%d is not leap year\n" "The season is spring/summer/autumn/winter" "The number of days of this month is %d\n"

程序运行示例如下：

Please enter year,month:2012,11

2012 is leap year

The season is autumn

The number of days of this month is 30

#include <stdio.h>

int main()

{

int y, m, leap, season, days;

printf("Please enter year,month:");

scanf("%d,%d", &y, &m);

if ((y % 4 == 0 && y % 100 != 0) || (y % 400 == 0))

leap = 1;

else

leap = 0;

if (m >= 3 && m <= 5)

season = 1;

else if (m >= 6 && m <= 8)

season = 2;

else if (m >= 9 && m <= 11)

season = 3;

else

season = 4;

switch (m)

{

case 1:

case 3:

case 5:

case 7:

case 8:

case 10:

case 12:

days = 31;

break;

case 4:

case 6:

case 9:

case 11:

days = 30;

break;

case 2:

if (leap == 1)

days = 29;

else

days = 28;

}

printf("%d %s leap year\n", y, (leap == 1 ? "is" : "is not"));

printf("The season is ");

switch (season)

{

case 1:

printf("spring\n");

break;

case 2:

printf("summer\n");

break;

case 3:

printf("autumn\n");

break;

case 4:

printf("winter\n");

}

printf("The number of days of this month is %d\n", days);

return 0;

}

Q283.(10分)编程计算分段：

y= x x<1

y= 2x-1 1<=x<10

y= 3x-11 x>=10

从键盘输入一个单精度实数x，打印出y值。

\*\*输入提示信息格式要求为："Please input x:"；

\*\*输出格式要求为"y = %.2f\n"。

#include <stdio.h>

main()

{

float x,y;

printf("Please input x:");

scanf("%f",&x); //1

if(x<1) //1

y=x;

else if(x>=1 && x<10) //1

y=2\*x-1;

else if(x>=10) //1

y=3\*x-11;

printf("y = %.2f\n",y); //1

}

Q344.(10分)编程判断输入整数x的正负性和奇偶性。

\*\*输入格式要求："%d" 提示信息："Enter a integer:"

\*\*输出格式要求："The integer is a plus even number\n" "This integer is a plus odd number\n"

"This integer is a minus even number\n" "This integer is a minus odd number\n"

"This integer is zero.\n"

程序运行示例如下：

Enter a integer:-9

This integer is a minus odd number

#include <stdio.h>

main()

{

int x;

printf("Enter a integer:");

scanf("%d",&x);

if(x!=0)

{

if(x>0)

{

x=x%2;

if(x==0) printf("The integer is a plus even number\n");

else printf("This integer is a plus odd number\n");

}

else

{

x=x%2;

if(x==0) printf("This integer is a minus even number\n");

else printf("This integer is a minus odd number\n");

}

}

else printf("This integer is zero.\n");

}

Q452.(10分)编写一个程序完成输入一个整数，输出它的符号。

\*\*输入格式要求："%d" 提示信息："Enter n:"

\*\*输出格式要求："+\n" "-\n" "this number has no sign\n"

程序运行示例1如下：

Enter n:90

+

程序运行示例2如下：

Enter n:0

this number has no sign

#include <stdio.h>

int main()

{

int n;

printf("Enter n:");

scanf("%d", &n);

if (n > 0)

{

printf("+\n");

}

else if (n < 0)

{

printf("-\n");

}

else

{

printf("this number has no sign\n");

}

}

Q552.(10分)如果整数A的全部因子（包括1，不包括A本身）之和等于B；且整数B的全部因子（包括1，不包括B本身）之和等于A，则A和B称为亲密数。求10000以内的亲密数。

\*\*输出格式要求："(%4d,%4d)\n"

程序运行示例如下：

( 220, 284)

(1184,1210)

(2620,2924)

(5020,5564)

(6232,6368)

#include <stdio.h>

int main()

{

int a, b, i, n;

for (a = 1; a < 10000; a++)

{

b = 0;

for (i = 1; i <= a / 2; i++)

{

if (a % i == 0)

b += i;

}

n = 0;

for (i = 1; i <= b / 2; i++)

{

if (b % i == 0)

n += i;

}

if (n == a && a < b)

printf("(%4d,%4d)\n", a, b);

}

return 0;

}

Q1567.(25分)水仙花是指各位数字的立方和等于该数本身的三位数。例如153是水仙花数，因为153=1\*1\*1+5\*5\*5+3\*3\*3.请编程计算并输出所有的水仙花数。

要求输出格式为"%6d"，且所有的水仙花数字在同一行。

#include <stdio.h>

int main()

{

int i, j, k, n;

for (n = 100; n < 1000; n++)

{

i = n / 100;

j = (n - i \* 100) / 10;

k = n % 10;

if (n == i \* i \* i + j \* j \* j + k \* k \* k)

{

printf("%6d", n);

}

}

printf("\n");

return 0;

}

Q485.(10分)求解一元二次方程。

\*\*输入格式要求："%lf %lf %lf" 提示信息："请分别输入二次项、一次项、常数项系数a,b,c："

\*\*输出格式要求："方程%.1lfx^2" "%.1lfx" "%.1lf=0" "无解.\n" "有两个根：x1=%.1lf,x2=%.1lf\n" "有一个根：x=%.1lf\n"

程序示例运行1如下：

请分别输入二次项、一次项、常数项系数a,b,c：1 2 1

方程1.0x^2+2.0x+1.0=0有一个根：x=-1.0

程序示例运行2如下：

请分别输入二次项、一次项、常数项系数a,b,c：1 5 6

方程1.0x^2+5.0x+6.0=0有两个根：x1=-2.0,x2=-3.0

程序示例运行3如下：

请分别输入二次项、一次项、常数项系数a,b,c：1 2 3

方程1.0x^2+2.0x+3.0=0无解.

#include <stdio.h>

#include <math.h>

int main()

{

double a, b, c, x1, x2;

printf("请分别输入二次项、一次项、常数项系数a,b,c：");

scanf("%lf %lf %lf", &a, &b, &c);//输入方程的系数

//打印方程

printf("方程%.1lfx^2", a);

if (b > 0)

printf("+");

printf("%.1lfx", b);

if (c > 0)

printf("+");

printf("%.1lf=0", c);

if (b \* b - 4 \* a \* c < 0)

{

printf("无解.\n");

}

else

{

x1 = (-b + sqrt(b \* b - 4 \* a \* c)) / (2 \* a);

x2 = (-b - sqrt(b \* b - 4 \* a \* c)) / (2 \* a);

if (b \* b - 4 \* a \* c > 0)

printf("有两个根：x1=%.1lf,x2=%.1lf\n", x1, x2);

else

printf("有一个根：x=%.1lf\n", x1);

}

return 0;

}

学生成绩排序结构体

#include<stdio.h>

struct student

{

int account;

char name[20];

int grade;

};

typedef struct student Student;

void sort(Student a[],int number);

int main(void)

{

int i;

int number;

Student a[100];

printf("请输入学生人数");

scanf("%d",&number);

for (i=0; i<number; i++)

{

printf("请输入学号");

scanf("%d",&a[i].account);

printf("请输入姓名");

getchar();

gets(a[i].name);

printf("请输入成绩");

scanf("%d",&a[i].grade);

}

sort(a,number);

}

void sort(Student a[],int number)

{

int i,j,max;

Student y;

for(i=0; i<number; i++)

{

max=i;//以第一名为例，假设第一个同学就是最高分即max为1

for(j=i+1; j<number; j++)//利用循环和别的同学进行比较

{

if(a[max].grade <a[j].grade)

{

max=j;//如果有人比他高，则用max变量记录下那个同学是第几个

}

}

if(max!=i)//结束比较后，将max是不是1,即看第一个同学是不是最高分，如果不是

{

y=a[i];

a[i]=a[max];

a[max]=y;//通过换位，将最高分的同学换到第一个，同理可得第二高，第三高。。。。。

}

}

printf("成绩由高到低排名为:\n");

for (i=0; i<number; i++)

{

printf("%d %d %s %d\n",i+1,a[i].account,a[i].name,a[i].grade);//将排序好的同学连学号，姓名打印出来

}

}

一、

编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

一、

#include  <stdio.h>

#include  <string.h>

#define N 80

void main()

{

    int    n;

    char  str[N], min[N];

    printf("Input five countries' names:\n");

    gets(str);//2

    strcpy(min, str);  //2

    for (n=1; n<5; n++)//3

    {

        gets(str); //2

        if (strcmp(str, min) < 0)    //4

        {

            strcpy(min, str);  //2

        }

    }

    printf("The minimum is:%s\n", min);//2

}

二、

从键盘输入10个整型数据，放入数组a 中，求其最大值，并输出。

\*\*要求输入提示信息为：无输入提示信息

\*\*要求输入格式为："%d"

\*\*要求输出格式为："max=%d\n"

二、

#include <stdio.h>

main()

{

    int a[10],i,max;

    for(i=0;i<10;i++)

    {

        scanf("%d",&a[i]);  //2

    }

    max=a[0];  //2

    for(i=1;i<10;i++)  //2

    {

        if(max<a[i])  //2

        {

            max=a[i];  //2

        }

    }

    printf("max=%d\n",max);  //2

}

三、

输入n×n阶矩阵，用函数编程计算并输出其两条对角线上的各元素之和。

\*\*输入格式要求："%d" 提示信息："Input n:" "Input %d\*%d matrix:\n"

\*\*输出格式要求："sum = %d\n" "%6d"

程序运行示例如下：

Input n:5

Input 5\*5 matrix:

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7

4 5 6 7 8

5 6 7 8 9

sum = 45

三、

#include <stdio.h>

#define  N   10

void InputMatrix(int a[N][N], int n);

int AddDiagonal(int a[N][N], int n);

int main()

{

    int a[N][N], n, sum;

    printf("Input n:");

    scanf("%d", &n);

    InputMatrix(a, n);

    sum = AddDiagonal(a, n);

    printf("sum = %d\n", sum);

    return 0;

}

/\*  函数功能： 输入n×n矩阵的元素值，存于数组a中 \*/

void InputMatrix(int a[N][N], int n)

{

    int i, j;

    printf("Input %d\*%d matrix:\n", n, n);

    for (i = 0; i < n; i++)

    {

        for (j = 0; j < n; j++)

        {

            scanf("%d", &a[i][j]);

        }

    }

}

/\*  函数功能： 计算n×n矩阵中两条对角线上的元素之和 \*/

int AddDiagonal(int a[N][N], int n)

{

    int i, j, sum = 0;

    for (i = 0; i < n; i++)

    {

        for (j = 0; j < n; j++)

        {

            if (i == j || i + j == n - 1)   sum = sum + a[i][j];

        }

    }

    return sum;

}

四、

按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。

其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息要求："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%d\t"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数

使用void main,不用写return。

四、

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

void main()

{

int s[N][N], n;

printf("Input n:\n");

scanf("%d", &n);

InputMatrix(s, n);//1

Transpose(s, n);//1

printf("The transposed matrix is:\n");

PrintMatrix(s, n);//1

}

void Transpose(int a[][N], int n)

{

int i, j, temp;

for (i = 0; i < n; i++) //1

{

for (j = i; j < n; j++)//1

{

temp = a[i][j];//1

a[i][j] = a[j][i];//1

a[j][i] = temp;//1

}

}

}

void InputMatrix(int a[][N], int n) {

int i, j;

printf("Input %d\*%d matrix:\n", n, n);

for (i = 0; i < n; i++) //1

{

for (j = 0; j < n; j++)//1

{

scanf("%d", &a[i][j]);//1

}

}

}

void PrintMatrix(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j < n; j++) //1

{

printf("%d\t", a[i][j]);//1

}

printf("\n");//1

}

}

五、

输入2个学生5门课的成绩，分别用函数求：（1）每个学生的平均分；（2）每门课的平均分；（3）找出最高的分数所对应的学生和课程；（4）求出平均分方差：

(1/n)(∑xi2) - （（∑xi）/ n）2。

五、

#include <stdio.h>

#define     N   2

#define     M   5

float score[N][M];

float a\_stu[N], a\_cor[M];

int main(void)

{

    int i, j, r, c;

    float h;

    float s\_diff(void);

    float highest(int \*, int \*);

    r = 0;

    c = 1;

    input\_stu();

    avr\_stu();

    avr\_cor();

    printf("\n 序号     课程1     2     3     4     5     平均分\n");

    for (i = 0; i < N; i++)

    {

        printf("\n NO%2d", i + 1);

        for (j = 0; j < M; j++)

        {

            printf("%8.2f", score[i][j]);

        }

        printf("%8.2f", a\_stu[i]);

    }

    printf("\n课平均");

    for (j = 0; j < M; j++)

    {

        printf("%8.2f", a\_cor[j]);

    }

    h = highest(&r, &c);

    printf("\n\n最高分%8.2f是%d号学生的第%d门课\n", h, r, c);

    printf("\n方差 %8.2f", s\_diff());

    return 0;

}

input\_stu(void)

{

    int i, j;

    for (i = 0; i < N; i++)

    {

        printf("\n输入学生%4d的5个成绩：\n", i + 1);

        for (j = 0; j < M; j++)

        {

            scanf("%f", &score[i][j]);

        }

    }

}

avr\_stu(void)

{

    int i, j;

    float s;

    for (i = 0; i < N; i++)

    {

        for (j = 0, s = 0; j < M; j++)

        {

            s += score[i][j];

        }

        a\_stu[i] = s / 5.0;

    }

}

avr\_cor(void)

{

    int i, j;

    float s;

    for (j = 0; j < M; j++)

    {

        s = 0;

        for (i = 0; i < N; i++)

        {

            s += score[i][j];

        }

        a\_cor[j] = s / (float)N;

    }

}

float highest(int \*r, int \*c)

{

    float high;

    int i, j;

    high = score[0][0];

    for (i = 0; i < N; i++)

    {

        for (j = 0; j < M; j++)

        {

            if (score[i][j] > high)

            {

                high = score[i][j];

                \*r = i + 1;

                \*c = j + 1;

            }

        }

    }

    return high;

}

float s\_diff(void)

{

    int i, j;

    float sumx, sumxn;

    sumx = 0.0;

    sumxn = 0.0;

    for (i = 0; i < N; i++)

    {

        sumx += a\_stu[i] \* a\_stu[i];

        sumxn += a\_stu[i];

    }

    return (sumx / N - (sumxn / N) \* (sumxn / N));

}

六、

编程实现一个输入指定范围内的整数的函数getint，其完整的函数原型为：

int getint(int min, int max);，它负责接收用户的输入，进行验证，保证接收的一定是一个介于min和max之间（[min, max]区间内）的一个整数并最后返回该整数。如果用户输入不合法，则会提示继续输入，直到输入合法时为止。要求编写完整的程序并测试你所写的getint函数。

程序的运行示例如下：

请输入介于3和100之间的一个整数：0

请输入[3..100]间的一个整数：102

请输入[3..100]间的一个整数：200

请输入[3..100]间的一个整数：36

你输入的整数为：36

六、

#include <stdio.h>

int getint(int min, int max)

{

int n;

printf("请输入[%d..%d]间的一个整数：\n", min, max);

while ((scanf("%d", &n) != 1) || (n < min) || (n > max))

{

printf("请输入[%d..%d]间的一个整数：\n", min, max);

}

return n;

}

int main()

{

int m;

m = getint(3, 100);

printf("你输入的整数为：%d\n", m);

return 0;

}

七、

按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。

其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息要求："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%5d"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程

七、

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

int main()

{

    int s[N][N], n;

    printf("Input n:\n");

    scanf("%d", &n);

    InputMatrix(s, n);//1

    Transpose(s, n);//1

    printf("The transposed matrix is:\n");

    PrintMatrix(s, n);//1

    return 0;

}

void Transpose(int a[][N], int n)

{

    int i, j, temp;

    for (i = 0; i < n; i++) //1

    {

        for (j = i; j < n; j++)//1

        {

            temp = a[i][j];//1

            a[i][j] = a[j][i];//1

            a[j][i] = temp;//1

        }

    }

}

void InputMatrix(int a[][N], int n)

{

    int i, j;

    printf("Input %d\*%d matrix:\n", n, n);

    for (i = 0; i < n; i++)

    {

        for (j = 0; j < n; j++)

        {

            scanf("%d", &a[i][j]); //1

        }

    }

}

void PrintMatrix(int a[][N], int n)

{

    int i, j;

    for (i = 0; i < n; i++) //1

    {

        for (j = 0; j < n; j++)  //1

        {

            printf("%5d", a[i][j]); //1

        }

        printf("\n");    //1

    }

}

八、

按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。

其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息要求："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%5d"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

八、

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

int main()

{

    int s[N][N], n;

    printf("Input n:\n");

    scanf("%d", &n);

    InputMatrix(s, n);//1

    Transpose(s, n);//1

    printf("The transposed matrix is:\n");

    PrintMatrix(s, n);//1

    return 0;

}

void Transpose(int a[][N], int n)

{

    int i, j, temp;

    for (i = 0; i < n; i++) //1

    {

        for (j = i; j < n; j++)//1

        {

            temp = a[i][j];//1

            a[i][j] = a[j][i];//1

            a[j][i] = temp;//1

        }

    }

}

void InputMatrix(int a[][N], int n)

{

    int i, j;

    for (i = 0; i < n; i++)

    {

        for (j = 0; j < n; j++)

        {

            scanf("%d", &a[i][j]); //1

        }

    }

}

void PrintMatrix(int a[][N], int n)

{

    int i, j;

    for (i = 0; i < n; i++) //1

    {

        for (j = 0; j < n; j++)  //1

        {

            printf("%5d", a[i][j]); //1

        }

        printf("\n");    //1

    }

}

九、

按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。

其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息要求："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%d\t"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

九、

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

main()

{

    int s[N][N], n;

    printf("Input n:\n");

    scanf("%d", &n);

    InputMatrix(s, n);//1

    Transpose(s, n);//1

    printf("The transposed matrix is:\n");

    PrintMatrix(s, n);//1

}

void Transpose(int a[][N], int n)

{

    int i, j, temp;

    for (i = 0; i < n; i++) //1

    {

        for (j = i; j < n; j++)//1

        {

            temp = a[i][j];//1

            a[i][j] = a[j][i];//1

            a[j][i] = temp;//1

        }

    }

}

void InputMatrix(int a[][N], int n) {

    int i, j;

    printf("Input %d\*%d matrix:\n", n, n);

    for (i = 0; i < n; i++) //1

    {

        for (j = 0; j < n; j++)//1

        {

            scanf("%d", &a[i][j]);//1

        }

    }

}

void PrintMatrix(int a[][N], int n)

{

    int i, j;

    for (i = 0; i < n; i++) //1

    {

        for (j = 0; j < n; j++) //1

        {

            printf("%d\t", a[i][j]);//1

        }

        printf("\n");//1

    }

}

十、

有一个3\*4的矩阵，求其中的最大元素的值。矩阵为：

{{1,3,5,7},{2,4,6,8},{15,17,34,12}};

\*\*输出格式要求："max value is %d\n"

程序运行示例如下：

max value is 34

十、

#include <stdio.h>

maxvalue(array)

    int array[][4];

{

    int i,j,k,max;

    max=array[0][0];

    for(i=0;i<3;i++)

        for(j=0;j<4;j++)

            if(array[i][j]>max)

                max=array[i][j];

    return max;

}

main()

{

    static int a[3][4]={{1,3,5,7},{2,4,6,8},{15,17,34,12}};

    printf("max value is %d\n",maxvalue(a));

}

十一、

阅读以下程序，找出其中的5处错误，并改正之。

#include <math.h>

int isprime(int m)

main()

{

    int n,flag;

    printf(“Input n:”);

    scanf(“%d”, n);

    flag=isprime(n);

    if(flag)

        printf(Yes!\n”)

    else

        printf(“No!\n”);

}

/\* 函数名：  isprime

函数功能：判断m是否为素数

入口参数：整型数m

返回值：  返回值为1时，表示m是素数；

返回值为0时，表示m不是素数 \*/

int isprime(int \*m)

{

    int i;

    if(m=1)

        return 0;  /\*1不是素数，所以返回0值\*/

    for(i=2; i<=sqrt(m); i++)

    {

    if(m%i==0)  return 0;

    }

    return 1;

}

十一、

int isprime(int m);

main()

{

int n,flag;

printf("Input n:");

scanf("%d",&n);

flag=isprime(n);

if(flag)

printf("Yes!\n");

else

printf("No!\n");

}

/\* 函数名： isprime

函数功能：判断m是否为素数

入口参数：整型数m

返回值： 返回值为1时，表示m是素数；

返回值为0时，表示m不是素数 \*/

int isprime(int m)

{

int i;

if(m==1)

return 0; /\*1不是素数，所以返回0值\*/

for(i=2; i<=sqrt(m); i++)

{

if(m%i==0) return 0;

}

return 1;

}

十二、

有5个人围坐在一起，问第5个人多大年纪，他说比第4个人大2岁；问第4个人，他说比第3个人大2岁；问第3个人，他说比第2个人大2岁；问第2个人，他说比第1个人大2岁。第1个人说自己10岁，请利用递归法编程计算并输出第5个人的年龄。

\*\*输出格式要求："The 5th person's age is %d\n"

十二、

#include  <stdio.h>

unsigned int ComputeAge(unsigned int n);

main()

{

    unsigned int n = 5;

    printf("The 5th person's age is %d\n", ComputeAge(n));

}

/\* 函数功能：用递归算法计算年龄 \*/

unsigned int ComputeAge(unsigned int n)

{

    unsigned int age;

    if (n == 1)

    {

        age = 10;

    }

    else

    {

        age = ComputeAge(n - 1) + 2;

    }

    return age;

}

十三、

(20分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能。

要求：

（1）按如下函数原型进行编程：

int Mystrlen(char str[]);

（2）在主函数中：

任意输入一个字符串（长度不超过80个字符）;

调用函数Mystrlen()计算输入字符串的实际长度;

然后打印计算结果。

\*\*要求输入提示信息为："Please input the string:\n"

\*\*要求输出格式为:"%d\n"

注意：

（1）字符串中的空格计入字符串实际长度，

但是字符串结束标志不计入字符串实际长度。

（2）必须使用函数编程。否则没有成绩。

十三、

#include <stdio.h>

int Mystrlen(char str[]);  //2

main()

{

    char str[80];

    int len;

        printf ("Please input the string:\n");  //2

    gets(str);  //2

    len = Mystrlen(str);  //2

    printf("%d\n", len);  //2

}

int Mystrlen(char str[])

{

    int len = 0, i;

    for (i=0; str[i]!=0; i++)  //2

    {

        len++;  //2

    }

    return len;  //2

}

十四、

不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

十四、

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

    char s[SIZE],t[SIZE],i;

    printf("Input s\n");

    gets(s);//1

    printf("Input t\n");

    gets(t);//1

    i=MyStrcmp(s,t);//2

    if(i>0)//1

        printf("string s>string t.\n");

    else if(i<0)//1

        printf("string s<string t.\n");

    else//1

        printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

    int i;

    for (i=0; s[i] == t[i]; i++ ) //6

    {

            if (s[i] ==  '\0' )  //2

            return 0 ;//2

    }

    return (s[i]-t[i]); //2

}

十五、

用字符数组作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

十五、

#include <stdio.h>

void Squeeze(char s[], char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char s[], char c)

{

int i, j;

for (i = j = 0; s[i] != '\0'; i++)

{

if (s[i] != c)

{

s[j] = s[i];

j++;

}

}

s[j] = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

十六、

程序改错。编写一个函数Inverse()，实现将字符数组中的字符串逆序存放的功能。（4个错误）

#include<string.h>

#include<stdio.h>

#define ARR\_SIZE = 80;

void Inverse(char str[]);

main()

{

    char a[ARR\_SIZE] ;

    printf("Please enter a string: ");

    gets(a);

    Inverse(char a[]);

    printf("The inversed string is: ");

    puts(a);

}

void Inverse(char str[])

{

    int   len, i, j;

    char  \*temp;

    len = strlen(str);

    for (i=0; j=len-1; i<j; i++; j--)

    {

    temp = str[i];

        str[i] = str[j];

        str[j] = temp;

    }

}

十六、

#include<string.h>

#include<stdio.h>

#define ARR\_SIZE 80

void Inverse(char str[]);

main()

{

char a[ARR\_SIZE] ;

printf("Please enter a string: ");

gets(a);

Inverse(a);

printf("The inversed string is: ");

puts(a);

}

void Inverse(char str[])

{

int len, i, j;

char temp;

len = strlen(str);

for (i=0, j=len-1; i<j; i++, j--)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

}

}

十七、

不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

十七、

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

十八、

调用GetInput()函数从键盘任意输入一个字符串，然后分别在GetInput()和main()中输出该字符串。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，

修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

|  |
| --- |
| #include  <stdio.h>  void GetInput(char\* s);    int main(void)  {     char \*ptr = NULL;     GetInput(ptr);     puts(ptr);     return 0;  }    void GetInput(char\* s)  {      scanf("%s", s);      puts(s);  } |

十八、

#include <stdio.h>

void GetInput(char\* s);

int main(void)

{

char str[80];

char \*ptr = str;

GetInput(ptr);

puts(ptr);

return 0;

}

void GetInput(char\* s)

{

scanf("%s", s);

puts(s);

}

十九、

)从键盘输入一个字符串a，将字符串a复制到字符串b中，再输出字符串b，即编程实现字符串处理函数strcpy()的功能，但要求不能使用字符串处理函数strcpy()。

\*\*输入格式要求："%s" 提示信息："Input a string:"

\*\*输出格式要求："The copy is:"

程序的运行示例如下：

Input a string:Hello China

The copy is:Hello China

十九、

#include <stdio.h>

#define N 80

void MyStrcpy(char dstStr[], char srcStr[]);

int main()

{

char a[N], b[N];

printf("Input a string:");

gets(a); /\* 输入字符串 \*/

MyStrcpy(b, a); /\* 将字符数组a中的字符串拷贝到b中 \*/

printf("The copy is:");

puts(b); /\* 输出复制后的字符串 \*/

return 0;

}

/\* 函数功能：用字符数组作为函数参数，字符串拷贝 \*/

void MyStrcpy(char dstStr[], char srcStr[])

{

int i = 0; /\* 数组下标初始化为0 \*/

while (srcStr[i] != '\0') /\* 若当前取出的字符不是字符串结束标志 \*/

{

dstStr[i] = srcStr[i];/\* 复制字符 \*/

i++; /\* 移动下标 \*/

}

dstStr[i] = '\0'; /\* 在字符串dstStr的末尾添加字符串结束标志 \*/

}

二十、

对输入的字符串进行长度验证，保证输入的字符串的长度在指定的范围内，如果不在指定的范围内，则一直提示用户输入，直到输入合法长度的字符串为止。

程序的示例运行如下：

请输入一个字符串(长度为[3..5]个字符)：a

请输入一个字符串(长度为[3..5]个字符)：ab

请输入一个字符串(长度为[3..5]个字符)：abcdef

请输入一个字符串(长度为[3..5]个字符)：abc

你输入的字符串为：abc

二十、

#include <stdio.h>

void getString(char \*str, int min, int max)

{

int len;

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

while (len < min || len > max)

{

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

}

}

int main(int argc, char \*argv[])

{

char string[100];

getString(string, 3, 5);

printf("你输入的字符串为：%s\n", string);

return 0;

}

二十一、

写一个函数，将一个字符串中的元音字母复制到另一个字符串，然后输出。

程序的运行示例如下：

输入字符串：Hello,world!

字符串中的元音字母是eoo

二十一、

#include <stdio.h>

int main()

{

char str[80], c[80];

void cpy();

printf("\n输入字符串：");

gets(str);

printf("%s", str);

cpy(str, c);

printf("\n字符串中的元音字母是%s", c);

return 0;

}

void cpy(char s[], char c[])

{

int i, j;

for (i = 0, j = 0; s[i] != '\0'; i++)

{

if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||

s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||

s[i] == 'u' || s[i] == 'U')

{

c[j] = s[i];

j++;

}

}

c[j] = '\0';

}

二十二、

请按给定的函数原型编程实现将字符数组中的字符串的第m个字符开始的n个字符逆序存放。要求在主函数读入字符串，且逆序存放后的字符串也在主函数打印。函数原型：

void inverse(char str[],int m,int n);

输入要求: 输入的一行字符串，应包含字母，数字以及空格字符

友情提示:

在执行输入字符串的函数之前,请用getchar();把输入缓冲区中的换行符读出!!!

\*\*\*\*输入提示信息和格式要求为：

"input m,n:"

"%d,%d"

"input the string:"

\*\*\*\*输出提示信息为:"the inverse string:"

二十二、

#include <stdio.h>

#define N 100

void inverse(char str[],int m,int n);

main()

{

    char str[N];

    int m,n;

    printf("input m,n:");   //1

    scanf("%d,%d",&m,&n);

    getchar();

    printf("input the string:");    //1

    gets(str);

        inverse(str,m,n);   //2

    printf("the inverse string:%s",str);    //1

}

void inverse(char str[],int m,int n)

{

    int i;

    char a[N];

    for(i=0;i<n;i++)

    {

        a[i]=str[m-1+i];    //1

    }

    for(i=0;i<n;i++)

    {

        str[m+i-1]=a[n-1-i];    //1

    }

}

二十三、

自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能。

要求：

（1）按如下函数原型进行编程：

int Mystrlen(char str[]);

（2）在主函数中：

任意输入一个字符串（长度不超过80个字符）;

调用函数Mystrlen()计算输入字符串的实际长度;

然后打印计算结果。

\*\*要求输入提示信息为："Please input the string:\n"

\*\*要求输出格式为:"%d\n"

注意：

（1）字符串中的空格计入字符串实际长度，

但是字符串结束标志不计入字符串实际长度。

（2）必须使用函数编程。否则没有成绩。

二十三、

#include <stdio.h>

int Mystrlen(char str[]); //2

main()

{

char str[80];

int len;

printf ("Please input the string:\n"); //2

gets(str); //2

len = Mystrlen(str); //2

printf("%d\n", len); //2

}

int Mystrlen(char str[])

{

int len = 0, i;

for (i=0; str[i]!=0; i++) //2

{

len++; //2

}

return len; //2

}

二十四、

按如下函数原型编写程序，用字符数组作函数参数编程实现字符串逆序存放功能。

void Inverse(char str[]);

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

二十四、

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

int main()

{

char a[N];

printf("Input a string:");

gets(a);

Inverse(a);

printf("Inversed results:%s\n", a);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char str[])

{

int len, i, j;

char temp;

len = strlen(str);

for (i = 0, j = len - 1; i < j; i++, j--)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

}

}

二十五、

该程序的作用是将一个字符串中的元音字母复制到另一个字符串，然后输出。改正错误。

#include<stdio.h>

main()

{

    char str1[80], str2[80];

    printf("输入一个字符串\n");

    gets(str1);

    Cpy(str1[], str2[]);

    printf("输出一个字符串\n");

    puts(str2);

}

void Cpy(char s[], char c[])

{

    int i, j;

    for(i=0; s[i] != '\0'; i++)

    {

        if(s[i]= 'a'||s[i]= 'A'||s[i]= 'e'||s[i]= 'E'||

          s[i]= 'i'||s[i]= 'I'||s[i]= 'o'||s[i]= 'O'||s[i]= 'u'||s[i]= 'U')

        {

            s[i]=c[j];

            j++;

        }

    }

}

二十五、

#include<stdio.h>

void Cpy(char s[], char c[]);

main()

{

char str1[80], str2[80];

printf("输入一个字符串\n");

gets(str1);

Cpy(str1, str2); //改为：Cpy(str1, str2)

printf("输出一个字符串\n");

puts(str2);

}

void Cpy(char s[], char c[])

{

int i, j = 0; //改为：j=0;

for (i = 0; s[i] != '\0'; i++)

{

if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||

s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||

s[i] == 'u' || s[i] == 'U')

//将=改为：==

{

c[j] = s[i]; //改为:c[j] = s[i];

j++;

}

}

c[j] = '\0';

}

二十六、

输入一个字符，如果此字符在已知的字符串中出现，则删除此字符，如删除成功就返回被删字符，否则返回'\0'。请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

#include <stdio.h>

char sh(char \*a, char x)

{

int i = 0, k;

char q;

while (a[i] != '\0') i++;

for (int j = 0; j < i; j++);

{

if (a[j] = x)

{

q = a[j];

for (k = j; k <= i; k++)

a[k] = a[k + 1];

return q;

}

}

return '\0';

}

main()

{

char a[6] = "seven", x, ch;

printf("The Original string is:");

puts(a);

printf("Enter x: ");

scanf("%d", &x);

ch = sh(a, x);

if (ch = '\0')

printf("%c not found!!!", x);

else

{

printf("The deleted charactor is :");

puts(x);

}

printf("\nThe Original string changes to :");

puts(a);

}

二十六、

#include <string.h>

#include <stdio.h>

char sh(char \*a, char x)

{

int i = 0, k, j;

char q;

while (a[i] != '\0') i++;

for (j = 0; j < i; j++)

{

if (a[j] == x)

{

q = a[j];

for (k = j; k < i; k++)

a[k] = a[k + 1];

return q;

}

}

return '\0';

}

main()

{

char a[6] = "seven", x, ch;

printf("The Original string is:");

puts(a);

printf("Enter x: ");

scanf("%c", &x);

ch = sh(a, x);

if (ch == '\0')

printf("%c not found!!!", x);

else

{

printf("The deleted charactor is :");

putchar(x);

}

printf("\nThe Original string changes to :");

puts(a);

}

二十七、

不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

二十七、

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);//1

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])//1

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

二十八、

不使用字符串处理函数strcat()编程实现strcat()的功能，即任意输入两个字符串，然后连接这两个字符串，返回连接后字符串的首地址。

程序的运行示例如下：

Input the first string:Hello

Input the second string:China

The result is: HelloChina

二十八

#include <stdio.h>

#define N 80

char \*MyStrcat(char \*dstStr, char \*srcStr);

int main()

{

char first[2 \* N]; /\* 这个数组应该足够大，以便存放连接后的字符串 \*/

char second[N];

char \*result = NULL;

printf("Input the first string:");

gets(first);

printf("Input the second string:");

gets(second);

result = MyStrcat(first, second);

printf("The result is: %s\n", result);

return 0;

}

/\* 函数功能：连接两个字符串 \*/

char \*MyStrcat(char \*dstStr, char \*srcStr)

{

char \*pStr = dstStr; /\* 保存字符串首地址指针 \*/

/\* 将指针移到字符串末尾 \*/

while (\*dstStr != '\0')

{

dstStr++;

}

/\* 将字符串2内容复制到字符串1的后面 \*/

for (; \*srcStr != '\0'; dstStr++, srcStr++)

{

\*dstStr = \*srcStr;

}

\*dstStr = '\0'; /\* 在连接后的字符串的末尾添加字符串结束标志 \*/

return pStr; /\* 返回连接后的字符串的指针 \*/

}

二十九、

不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

二十九、

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

三十、

自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能，即在主函数中任意输入

一个字符串（长度不超过80个字符），调用该函数计算

输入字符串的实际长度，然后打印计算结果。

要求按如下函数原型编程实现计算字符串长度的功能。

int Mystrlen(char str[]);

要求必须按照题目要求和用函数编程，否则不给分。

\*\*要求输入提示信息格式为："Enter a string:"

\*\*输出格式为："The length of the string is:%d\n"

三十、

#include <stdio.h>

int Mystrlen(char str[]);

main()

{

char str[80];

int len;

printf("Enter a string:");

gets(str); //1

len = Mystrlen(str); //1

printf("The length of the string is:%d\n",len);

}

int Mystrlen(char str[]) //1

{

int len = 0, i;

for (i=0; str[i]!='\0'; i++) //1

{

len++; //1

}

return len; //1

}

三十一、

不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

三十一、

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

三十二、

下面程序的Squeeze函数的功能是删除字符串s中所出现的与变量c相同的字符。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

void Squeeze(char s[], char c);

main()

{

char a[80], c;

scanf("%s", a);

scanf("%c", &c);

Squeeze(char a[], char c);

printf("%s\n", a);

}

void Squeeze(char s[], char c);

{

int i, j;

for (i = 0; s[i] != '\0'; i++);

{

if (s[i] != 'c')

{

s[j] = s[i];

j++;

}

}

s[i] = '\0';

}

三十二、

#include <stdio.h>

void Squeeze(char s[], char c);

main()

{

char a[80], c;

gets(a);

scanf("%c", &c);

Squeeze(a, c);

printf("%s\n", a);

}

void Squeeze(char s[], char c)

{

int i, j;

for (i = 0, j = 0; s[i] != '\0'; i++)

{

if (s[i] != c)

{

s[j] = s[i];

j++;

}

}

s[j] = '\0';

}

三十三、

调用GetInput()函数从键盘任意输入一个字符串，然后分别在GetInput()和main()中输出该字符串。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，

修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

|  |
| --- |
| #include  <stdio.h>  void GetInput(char\* s);    int main(void)  {     char \*ptr = NULL;     GetInput(ptr);     puts(ptr);     return 0;  }    void GetInput(char\* s)  {      scanf("%s", s);      puts(s);  } |

三十三、

#include <stdio.h>

void GetInput(char\* s);

int main(void)

{

char str[80];

char \*ptr = str;

GetInput(ptr);

puts(ptr);

return 0;

}

void GetInput(char\* s)

{

scanf("%s", s);

puts(s);

}

三十四、

利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

三十四、

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

三十五、

下面程序的功能是删除字符串中第一次出现的a字符。找出其中错误，并改正之。

#include <stdio.h>

#include <string.h>

void fun(char \*x,int n,\*t);

{   int i,k=0;

    x[n]='a';x[n+1]='\0';

    while(x[k]!=’a’) k++;

    if(k==n){\*t=0;break;}

    else

    {   for(i=k;i<n;i++)

            x[i]=x[i+1];

        x[i]='\0';

    }

}

main()

{

    char x[20];

    int len,t;

    gets(x);

    puts(x);

    len=strlen(x);

    fun(x,len,t);

    if(t==0) printf("Not exist!\n");

    else    puts(x);

}

三十五、

#include <stdio.h>

#include <string.h>

void fun(char \*x,int n,int \*t)

{ int i,k=0;

x[n]='a';x[n+1]='\0';

while(x[k]!='a') k++;

if(k==n){\*t=0;}

else

{ for(i=k;i<n;i++)

x[i]=x[i+1];

x[i-1]='\0';

}

}

main()

{

char x[20];

int len,t;

gets(x);

puts(x);

len=strlen(x);

fun(x,len,&t);

if(t==0) printf("Not exist!\n");

else puts(x);

}

三十六、

输入一行字符，用函数编程统计其中有多少单词。假设单词之间以空格分开。

\*\*输入格式要求：提示信息："Input a string:"

\*\*输出格式要求："Numbers of words = %d\n"

三十六、

#include <stdio.h>

int CountWords(char str[]);

int main()

{

char str[20];

printf("Input a string:");

gets(str);

printf("Numbers of words = %d\n", CountWords(str));

return 0;

}

int CountWords(char str[])

{

int i, num;

num = (str[0] != ' ') ? 1 : 0;

for (i = 1; str[i] != '\0'; i++)

{

if (str[i] != ' ' && str[i - 1] == ' ')

{

num++;

}

}

return num;

}

三十七、

利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

三十七、

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

三十八、

下面的程序判断是不是回文，是回文返回1，否则返回0。回文是顺读和倒读都一样。请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

#include <stdio.h>

hui(char \*a)

{

int i, k, j = 0;

for (i = 0; i < 15; i++)

if (a[i] != '\0')

j++;

if (j % 2 == 0)

return 0;

for (i = 0, k = j; i <= (j - 1) / 2; i++, k--)

if (a[i] != a[k]) return 0;

return 1;

}

main()

{

int x;

char a[15];

gets(a);

puts(a);

x = hui(a);

if (x == 1)

printf("this is huiwen!!!");

else

printf("this is not huiwen!!!");

}

三十八、

#include <string.h>

#include <stdio.h>

hui(char \*a)

{

int i, k, j = 0;

for (i = 0; i < 15; i++)

if (a[i] != '\0')

j++;

else

break;

if (j % 2 == 0)

return 0;

for (i = 0, k = j - 1; i <= (j - 1) / 2; i++, k--)

if (a[i] != a[k])

return 0;

return 1;

}

main()

{

int x;

char a[15];

gets(a);

puts(a);

x = hui(a);

if (x == 1)

printf("this is huiwen!!!");

else

printf("this is not huiwen!!!");

}

三十九、

下面程序用于将输入的两个字符串连接起来，然后打印连接后的字符串。找出其中错误并改正之。

#include <stdio.h>

#include <string.h>

main()

{

    char    first[80];

    char    second[80];

    printf("Input the first string:");

    gets(first);

    printf("Input the second string:");

    gets(second);

    MyStrCat(first, second);

    printf("The result is : %s\n", first);

}

void MyStrCat(char \*dest, char \*source);

{

    int i = 0, j = 0;

    while (\*(dest+i)!='\0')

    {

        i++;

    }

    for (; \*(source+i)!='\0'; i++,j++)

    {

        \*(dest+j) = \*(source+i);

    }

    \*(dest+j) = '\0';

}

三十九、

#include <stdio.h>

#include <string.h>

void MyStrCat(char \*dest, char \*source);

main()

{

char first[80];

char second[80];

printf("Input the first string:");

gets(first);

printf("Input the second string:");

gets(second);

MyStrCat(first, second);

printf("The result is : %s\n", first);

}

void MyStrCat(char \*dest, char \*source)

{

int i = 0, j = 0;

while (\*(dest+i)!='\0')

{

i++;

}

for (; \*(source+j)!='\0'; i++,j++)

{

\*(dest+i) = \*(source+j);

}

\*(dest+i) = '\0';

}

四十、

程序中函数 transform()的功能: 将一个由二进制数字字符组成的字符串转换为与其值相等的十进制整数。规定输入的字符串最多只能包含10位二进制数字字符。

\*\*输入格式要求：

要求对输入的二进制数字字符组成的字符串进行长度和内容确为二进制的判断。

提示信息："输入一个二进制字符串（10位）:" "错误：字符串太长\n" "错误:%c 不是一个二进制字符\n"

\*\*输出格式要求："输入的二进制字符串为" "\n%s 转换成十进制整数为：%d\n"

程序运行示例如下：

输入一个二进制字符串（10位）:10010

输入的二进制字符串为10010

10010 转换成十进制整数为：18

提示：exit(0)函数在stdlib.h库中定义。

四十、

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

int transform(char ch[])

{

int n=0,i=0;

while (ch[i] != '\0')

{

n = n \* 2 + ch[i] - '0';

i++;

}

return n;

}

void main()

{

char s[10];

int i;

int n;

printf("输入一个二进制字符串（10位）:");

gets(s);

if (strlen(s) > 10)

{

printf("错误：字符串太长\n");

exit(0);

}

for (i=0;s[i]!='\0';i++)

{

if (s[i] != '0' && s[i] != '1')

{

printf("错误:%c 不是一个二进制字符\n", s[i]);

exit(0);

}

}

printf("输入的二进制字符串为");

puts(s);

n = transform(s);

printf("\n%s 转换成十进制整数为：%d\n", s, n);

}

四十一、

下面的函数MyStrcmp()用于实现函数strcmp()的功能，将两个字符串s和t进行比较，要求将两个字符串中第一个不相同字符的ASCII码值之差作为MyStrcmp()函数的返回值。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  int MyStrcmp(char s[], char t[]);  main()  {      char  str1[20],str2[20];        printf("Input string:");      gets(str1);      printf("Input another string:");      gets(str2);        if (MyStrcmp(str1[],str2[]) > 0)      {          printf("str1 大于 str2\n");      }      else if (MyStrcmp(str1[],str2[]) < 0)      {          printf("str1 小于 str2\n");      }      else      {          printf("str1 等于 str2\n");      }  }    int MyStrcmp(char s[], char t[])  {      int i;        for (i=0; s[i] == t[i]; i++);      {              if (s[i] = '\0' )  return 0 ;      }      return (s[i] - t[i]);  } |

四十一、

#include <stdio.h>

int MyStrcmp(char s[], char t[]);

main()

{

char str1[20],str2[20];

printf("Input string:");

gets(str1);

printf("Input another string:");

gets(str2);

if (MyStrcmp(str1,str2) > 0)

{

printf("str1 大于 str2\n");

}

else if (MyStrcmp(str1,str2) < 0)

{

printf("str1 小于 str2\n");

}

else

{

printf("str1 等于 str2\n");

}

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++)

{

if (s[i] == '\0' ) return 0 ;

}

return (s[i] - t[i]);

}

四十二、

编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

四十二、

#include <stdio.h>

#include <stdlib.h>

#define N 100

int getline(char \*s, int len);

int main()

{

char line[N], b[20], \*ptr1, \*ptrb;

int n = 0, j, a[50];

printf("Please enter a string:");

getline(line, N);

ptr1 = line;

while (\*ptr1 != '\0')

{

ptrb = b;

for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)

{

\*ptrb = \*ptr1;

}

if (j)

{

\*ptrb = '\0';

\*(a + n) = atoi(b);

n++;

}

if (\*ptr1 != '\0')ptr1++;

}

printf("the result of output:\n");

for (j = 0; j < n; j++)

{

printf("%10d\n", \*(a + j));

}

return 0;

}

int getline(char \*s, int len)

{

int c;

char \*p = s;

while (--len > 0 && (c = getchar()) != '\n')

{

\*s++ = c;

}

\*s = '\0';

return s - p;

}

四十三、

编写计算组合数的程序。要求输入数据要有容错功能。

\*\*输入格式要求："%d,%d" 提示信息："Input m,k (m>=k>0):"

\*\*输出格式要求："p = %.0f\n"

程序运行示例1如下：

Input m,k (m>=k>0):3,2

p = 3

程序运行示例2如下：

Input m,k (m>=k>0):2,3

Input m,k (m>=k>0):3,3

p = 1

Input m,k (m>=k>0):-2,-4

Input m,k (m>=k>0):4,2

p = 6

四十三、

#include <stdio.h>

unsigned long Fact(unsigned int n);

int main()

{

int m, k;

double p;

do

{

printf("Input m,k (m>=k>0):");

scanf("%d,%d", &m, &k);

}

while (m < k || m < 0 || k < 0);

p = (double)Fact(m) / (Fact(k) \* Fact(m - k));

printf("p = %.0f\n", p);

return 0;

}

/\* 函数功能：用迭代法计算无符号整型变量n的阶乘，当n>=0时返回n!；否则返回-1 \*/

unsigned long Fact(unsigned int n)

{

unsigned int i;

unsigned long result = 1;

for (i = 2; i <= n; i++)

result \*= i;

return result;

}

四十四、

数值151是一个回文素数。因为151既是一个素数也是一个回文数（回文数是从前向后读和从后向前读都一样的数）。写一个程序找出所有在[a,b]区间内的回文素数(5 <= a < b < 1000)。要求判素数调用primer函数实现，判回文数调用huiwen函数实现。

素数primer函数原型为:

int primer(int number)

参数说明:

参 数: number是要进行判断的数；

回文数huiwen函数原型为:

int huiwen(int number)

参数说明:

参 数: number是要进行判断的数；

输入

第一行：两个整数，a和b

输出

符合条件的回文素数，一个数一行。

输入样例

5 500

输出样例

5

7

11

101

131

151

181

191

313

353

373

383

四十四、

#include <math.h>

#include <stdio.h>

// Judge if the palindromes that have been found are primes

int primer(int number)

{

int i;

int sqrt\_number;

sqrt\_number = sqrt(number);

for (i = 2; i <= sqrt\_number; i++)

{

if ((number % i) == 0)

return 0; //The number is not prime

}

return 1; // The number is prime

}

int huiwen(int number)

{

int g,s,b;

g=number%10;

s=number/10%10;

b=number/100;

if (number>=10&&number<100)

{ if(g!=s) return 0; }//The number is not huiwen

else

if (number>=100&&number<1000)

{ if(g!=b) return 0; }//The number is not huiwen

return 1; // The number is huiwen

}

int main()

{

int a, b; // Enter two numbers as the range of prime palindromes

int i;

scanf("%d %d", &a, &b);

for (i = a; i <= b; i++)

{

if (primer(i) && huiwen(i))

printf("%d\n", i);

}

return 0;

}

四十五、

采用穷举法，用函数编程实现计算两个正整数的最小公倍数（Least Common Multiple，LCM）的函数，在主函数中调用该函数计算并输出从键盘任意输入的两整数的最小公倍数。

\*\*输入格式要求："%d,%d" 提示信息："Input a,b:"

\*\*输出格式要求："Least Common Multiple of %d and %d is %d\n" "Input error!\n"

程序运行示例如下：

Input a,b:16,24

Least Common Multiple of 16 and 24 is 48

四十五、

#include <stdio.h>

int Lcm(int a, int b);

int main()

{

int a, b, x;

printf("Input a,b:");

scanf("%d,%d", &a, &b);

x = Lcm(a, b);

if (x != -1)

{

printf("Least Common Multiple of %d and %d is %d\n", a, b, x);

}

else

{

printf("Input error!\n");

}

return 0;

}

/\* 函数功能：计算a和b的最小公倍数，输入负数时返回-1 \*/

int Lcm(int a, int b)

{

int i;

if (a <= 0 || b <= 0)

{

return -1;

}

for (i=1; i<b; i++)

{

if (i \* a % b == 0)

return i \* a;

}

return b \* a;

}

四十六、

从键盘任意输入一个整数n，编程计算并输出1~n之间的所有素数之和

输入提示信息："Input n:"

输入格式："%d"

输出格式："sum = %d\n"

四十六、

#include <stdio.h>

#include <math.h>

int IsPrime(int x);

int main()

{

int m, n, sum = 0;

printf("Input n:");

scanf("%d", &n);

for (m = 1; m <= n; m++)

{

if (IsPrime(m)) /\* 素数判定 \*/

{

sum += m;

}

}

printf("sum = %d\n", sum);

return 0;

}

/\* 函数功能：判断x是否是素数，若函数返回0，则表示不是素数，若返回1，则代表是素数 \*/

int IsPrime(int x)

{

int i, flag = 1;

int squareRoot = (int)sqrt(x);

if (x <= 1) flag = 0; /\* 负数、0和1都不是素数 \*/

for (i = 2; i <= squareRoot && flag; i++)

{

if (x % i == 0) flag = 0; /\* 若能被整除，则不是素数 \*/

}

return flag;

}

四十七、

函数fun功能：由小到大输出两实数。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

fun();

}

fun(float a, float b)

{

float t;

scanf("%f%f", &a, &b);

if (a < b)

{

t = a;

a = b;

b = t;

}

printf("%5.2,%5.2f\n", &a, &b);

}

四十七、

#include <stdio.h>

fun()

{

float t;

float a, b;

scanf("%f%f", &a, &b);

if (a < b)

{

t = a;

a = b;

b = t;

}

printf("%5.2f,%5.2f\n", a, b);

}

main()

{

fun();

}

四十八、

按如下函数原型，采用欧几里德算法编写计算两个整数的最大公约数的函数Gcd()。欧几里德算法，也称辗转相除法。其基本思想是：对正整数a和b，连续进行求余运算，直到余数为0为止，此时非0的除数就是最大公约数。

/\* 函数功能：计算a和b的最大公约数，输入负数时返回-1 \*/

int Gcd(int a, int b)

要求如下：

（1）从键盘任意输入的两整数

主函数调用Gcd()函数，并输出两整数的最大公约数。

（2）Gcd函数原型为：

int Gcd(int a, int b);

如果输入的数不是正整数，则返回-1，

否则，返回两个数的最大公约数。

（3）\*\*输入提示信息格式要求："Input a,b:\n"

输入两个整数时用,号分隔

\*\*输出提示信息要求：

如果输入的数不是正整数，则输出"Input error!\n"

否则按如下格式输出"Gcd=%d\n"

四十八、

#include <stdio.h>

int Gcd(int a, int b);

main()

{

int a, b, c;

printf("Input a,b:\n");

scanf("%d,%d", &a, &b);//1

c = Gcd(a,b);//2

if (c != -1)//1

{

printf("Gcd=%d\n", c);

}

else//1

{

printf("Input error!\n");

}

}

int Gcd(int a, int b)//1

{

int r;

if (a <= 0 || b <= 0)//2

{

return -1;//1

}

do{

r = a % b;//1

a = b;//1

b = r;//1

}while (r != 0);//e2

return a; //1

}

四十九、

设计一个函数MaxCommonFactor()，利用欧几里德算法（也称辗转相除法）计算两个正整数的最大公约数。假设有两个整数m和n，通过连续运用求余运算，直到余数为0为止，最后非0的余数就是最大公约数。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

int MaxCommonFactor(int a, b);

main()

{

int a, b, x;

scanf("%d,%d", a, b);

x = MaxCommonFactor(a, b);

if (x < 0)

printf("Input Error!\n");

printf("%d\n", x);

}

int MaxCommonFactor(int a, b);

{

int r;

if (a <= 0 && b <= 0)

return -1;

do

{

r = a % b;

b = r;

a = b;

}

while (r != 0)

return b;

}

四十九、

#include <stdio.h>

int MaxCommonFactor(int a, int b);

main()

{

int a, b, x;

scanf("%d,%d", &a, &b);

x = MaxCommonFactor(a, b);

if (x < 0) //或者if (x==-1)

printf("Input Error!\n");

else

printf("%d\n", x);

}

int MaxCommonFactor(int a, int b)

{

int r;

if (a <= 0 || b <= 0)

return -1;

do

{

r = a % b;

a = b;

b = r;

}

while (r != 0);

return a;

}

五十、

编写一个找出大于给定整数m的最小素数的函数。所谓素数是指这个整数只能被1和自身整除。要求在主函数中输入一个整数m，调用子函数IsPrime()找出其最小素数，并输出结果。

函数原型：int IsPrime(int x)

\*\*\*\*要求输入提示信息为：Please input n:

\*\*\*\*输出格式要求为："%d\n"

五十、

#include <stdio.h>

#include <math.h>

int IsPrime(int x)

{

int i,flag=1;

int squareRoot=sqrt(x);

if(x<=1) //1

{

flag=0;

}

for(i=2; i<=squareRoot&&flag; i++) //2

{

if(x%i==0) //1

{

flag=0;

}

}

return flag;

}

main()

{

int n,i;

printf("Please input n:");

scanf("%d",&n);

i=n+1;

while(i>=n) //1

{

if(IsPrime(i)) //1

{

printf("%d\n",i); //1

break;

}

else

{

i++; //1

}

}

}

五十一、

计算两个正整数的最大公约数。请按以下给定的函数原型编程：

int MaxCommonFactor(int a, int b);

返回值：返回的是最大公约数；若输入的数据有任意一个不满足条件，返回值是-1。

程序的运行示例1如下：

45,155 （输入，用,隔开）

5 （输出）

程序的运行示例2如下：

-9,20

-1

五十一、

#include <stdio.h>

int MaxCommonFactor(int a, int b);

int main()

{

int x, y;

scanf("%d,%d", &x, &y);

printf("%d", MaxCommonFactor(x, y));

return 0;

}

int MaxCommonFactor(int a, int b)

{

int min = a, i;

if (a <= 0 || b <= 0)

return -1;

else

{

if (b < a)

{

min = b;

}

for (i = min; i >= 1; i--)

{

if (a % i == 0 && b % i == 0)

return i;

}

}

}

五十二、

编写一个C函数，实现保留小数点后第N-1位，从第N位四舍五入。用该函数对数组a[5]中的各元素从小数点后第2位开始四舍五入。

a[5]={2.33,2.56,2.65,2.66,2.30}

\*\*输出格式要求："Old array:\n" "a[%d]=%.3f\t" "\nnew array:\n" "a[%d]=%.2f\t"

程序运行示例如下：

Old array:

a[0]=2.330 a[1]=2.560 a[2]=2.650 a[3]=2.660 a[4]=2.300

new array:

a[0]=2.30 a[1]=2.60 a[2]=2.70 a[3]=2.70 a[4]=2.30

五十二、

#include <stdio.h>

#include <math.h>

float a[5] = {2.33, 2.56, 2.65, 2.66, 2.30};

float t(float x,int n)

{

float x1, x2;

int i;

for (x1 = x, i = 0; i < n - 1; i++)

x1 \*= 10.0;

x2 = floor((double)x1);

x1 = floor((double)(x1 - x2) \* 10);

if (x1 >= 5)

x2 += 1;

for (i = 0; i < n - 1; i++)

x2 /= 10.0;

return x2;

}

main()

{

float t();

int i;

printf("Old array:\n");

for (i = 0; i < 5; i++)

printf("a[%d]=%.3f\t", i, a[i]);

printf("\nnew array:\n");

for (i = 0; i < 5; i++)

printf("a[%d]=%.2f\t", i, t(a[i], 2));

}

五十三、

从键盘输入一个年份，判断是否为闰年。

题目要求：

输入提示： Input a year:

输出要求：

如果是闰年： %d is a leap year!

如果不是闰年： %d is not a leap year!

五十三、

#include <stdio.h>

int main()

{

int year;

printf("Input a year:");

scanf("%d", &year);

if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))

{

printf("%d is a leap year!\n", year);

}

else printf("%d is not a leap year!\n", year);

return 0;

}

五十四、

程序改错，实现设计一个函数MaxCommonFactor()，利用欧几里德算法（也称辗转相除法）计算两个正整数的最大公约数。该算法的基本思想是：假设有两个整数m和n，通过连续运用求余运算，直到余数为0为止，最后非0的余数就是最大公约数。

|  |
| --- |
| #include <stdio.h>  int MaxCommonFactor(int a, int b)  main()  {      int a, b, x;        printf("Input a,b:");      scanf("%d,%d",a,b);      x = MaxCommonFactor(a,b);      printf("MaxCommonFactor = %d\n", x);  }    int MaxCommonFactor(int a, int b);  {      int r;        do{          r = a % b;          a = b;          b = r;      }while (r != 0)      return      a;  } |

五十四、

#include <stdio.h>

int MaxCommonFactor(int a, int b);

main()

{ int a, b, x;

printf("Input a,b:");

scanf("%d,%d",&a,&b);

x = MaxCommonFactor(a,b);

printf("MaxCommonFactor = %d\n", x);

}

int MaxCommonFactor(int a, int b)

{

int r;

do{

r = a % b;

a = b;

b = r;

}while (r != 0);

return a;

}

五十五、

两个正整数的最大公约数是能够整除这两个整数的最大整数。

请按如下函数原型编写计算最大公约数的函数Gcd()。

int Gcd(int a, int b);

在主函数中调用该函数计算，并输出从键盘任意输入的两整数的最大公约数。

\*\*要求输入提示信息为: "Input a,b:\n"

\*\*要求输入格式为: "%d,%d"

\*\*要求输出格式为

如果a或b不是正整数，输出: "Input number should be positive!\n"

否则输出: "Greatest Common Divisor of %d and %d is %d\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

五十五、

#include <stdio.h>

int Gcd(int a, int b);

main()

{

int a, b, c;

printf("Input a,b:\n");

scanf("%d,%d", &a, &b); //1

c = Gcd(a, b); //1

if (c != -1) //1

{

printf("Greatest Common Divisor of %d and %d is %d\n", a, b, c); //1

}

else

{

printf("Input number should be positive!\n"); //1

}

}

int Gcd(int a, int b) //1

{

int i, t;

if (a <= 0 || b <= 0) //1

{

return -1; //1

}

t = a < b ? a : b; //2

for (i = t; i > 0; i--)//2

{

if (a % i == 0 && b % i == 0)//1

return i; //1

}

return 1; //1

}

五十六、

根据如下性质，设计函数MaxCommonFactor()，计算两个正整数的最大公约数。

性质1：当a>b时，计算a与b的公约数等价于计算a-b与b的公约数。

性质2：当a<b时，计算a与b的公约数等价于计算b-a与b的公约数。

性质3：当a=b时，a与b的公约数等于a或b。

请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

int MaxCommonFactor(int a, int b);

main()

{

int a, b, x;

printf("Input a,b:");

scanf("%d,%d", a, b);

x = MaxCommonFactor(int a, int b);

if (x < 0) printf("Input Error!\n");

printf("%d\n", x);

}

int MaxCommonFactor(int a, int b)

{

if (a <= 0 || b <= 0)

return -1;

while if (a != b);

{

if (a > b)

return a - b;

else if (b > a)

return b - a;

else return a;

}

}

五十六、

#include <stdio.h>

int MaxCommonFactor(int a, int b);

main()

{

int a, b, x;

printf("Input a,b:");

scanf("%d,%d", &a, &b);

x = MaxCommonFactor(a, b);

if (x < 0)

printf("Input Error!\n");

else

printf("%d\n", x);

}

int MaxCommonFactor(int a, int b)

{

if (a <= 0 || b <= 0)

return -1;

while (a != b)

{

if (a > b)

a = a - b;

else if (b > a)

b = b - a;

}

return a;

}

五十七、

请编写函数fuc（）,函数的功能是： 根据以下公式求出p的值，结果由函数值返回。m , n 是两个正整数，且 m>n. P= m!/(n!\*(n-m))!

\*\*输入格式要求："%d %d" 提示信息："请输入m,n的值( m>n )：\n"

\*\*输出格式要求："n项之和为：%lf\n"

程序运行示例如下：

请输入m,n的值( m>n )：

5 3

n项之和为：10.000000

五十七、

#include <stdio.h>

double fac(int m, int n)

{

int i;

long p = 1, q = 1, r = 1;

double d = 0.0;

for (i = 1; i <= m; i++)

{

p = p \* i;

}

for (i = 1; i <= n; i++)

{

q = q \* i;

}

for (i = 1; i <= m - n; i++)

{

r = r \* i;

}

d = (double)p / (r \* q);

return d;

}

main()

{

int n, m;

double d;

do

{

printf("请输入m,n的值( m>n )：\n");

scanf("%d %d", &m, &n);

}

while (m < n);

d = fac(m, n);

printf("n项之和为：%lf\n", d);

}

五十八、

改错题：设计一个函数MinCommonMultiple,其功能是计算两个正整数的最小公倍数。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

|  |
| --- |
| #include <stdio.h>  int MinCommonMultiple(int a,int b)  main()  {      int a,b,x;        printf("Input a,b:");      scanf("%d,%d",a,b);      x = MinCommonMultiple(a,b);      printf("MinCommonMultiple = %d\n", x);  }  int MinCommonMultiple(int a,int b);  {      int i;        for(i=1; i<a\*b; i++)      {          if(i%a == 0 || i%b == 0)              return i;      }  } |

五十八、

int MinCommonMultiple(int a,int b);

main()

{

int a,b,x;

printf("Input a,b:");

scanf("%d,%d",&a,&b);

x = MinCommonMultiple(a,b);

printf("MinCommonMultiple = %d\n", x);

}

int MinCommonMultiple(int a,int b)

{

int i;

for(i=1; i<a\*b; i++)

{

if(i%a == 0 && i%b == 0)

return i;

}

}

五十九、

波瓦松的分酒趣题。法国著名数学家波瓦松青年时代研究过一个有趣的数学问题：某人有12品脱的啤酒一瓶，想从中倒出6品脱，但他没有6品脱的容器，仅有一个8品脱和一个5品脱的容器，怎样倒才能将啤酒分为两个6品脱呢？

\*\*输入格式要求："%d%d%d%d" 提示信息："Input Full a,Empty b,c,Get i:"

\*\*输出格式要求：" a%d b%d c%d\n%4d%4d%4d\n" "%4d%4d%4d\n"

程序运行示例如下：

Input Full a,Empty b,c,Get i: 12 8 5 6

a12 b8 c5

12 0 0

4 8 0

4 3 5

9 3 0

9 0 3

1 8 3

1 6 5

6 6 0

a12 b5 c8

12 0 0

7 5 0

7 0 5

2 5 5

2 2 8

10 2 0

10 0 2

5 5 2

5 0 7

0 5 7

0 4 8

8 4 0

8 0 4

3 5 4

3 1 8

11 1 0

11 0 1

6 5 1

6 0 6

五十九、

#include <stdio.h>

int i;

main()

{ int a,y,z;

printf("Input Full a,Empty b,c,Get i:");

scanf("%d%d%d%d",&a,&y,&z,&i);

getti(a,y,z);

getti(a,z,y);

}

getti(a,y,z)

int a,y,z;

{int b=0,c=0;

printf(" a%d b%d c%d\n%4d%4d%4d\n",a,y,z,a,b,c);

while(a!=i||b!=i&&c!=i)

{if(!b)

{a-=y;b=y;}

else if(c==z)

{a+=z;c=0;}

else if(b>z-c)

{b-=(z-c);c=z;}

else {c+=b;b=0;}

printf("%4d%4d%4d\n",a,b,c);

}

}

六十、

编写一个找出大于给定整数m的最小素数的函数，要求在主函数中输入一个整数m，调用子函数IsPrime()找出其最小素数，并在主函数中输出结果。

函数原型：int IsPrime(int x)

\*\*\*\*要求输入提示信息为：Please input n:

\*\*\*\*输出格式要求为："%d\n"

六十、

#include <stdio.h>

#include <math.h>

int IsPrime(int x)

{

int i,flag=1;

int squareRoot=sqrt(x);

if(x<=1) //1

{

flag=0;

}

for(i=2; i<=squareRoot&&flag; i++) //2

{

if(x%i==0) //1

{

flag=0;

}

}

return flag;

}

main()

{

int n,i;

printf("Please input n:");

scanf("%d",&n);

i=n+1;

while(i>=n) //1

{

if(IsPrime(i)) //1

{

printf("%d\n",i); //1

break;

}

else

{

i++; //1

}

}

}

六十一、

求1898

现將不超过2000的所有素数从到大排成第一行，第二行上的每个数都等于它“右肩”上的素数与“左肩”上的素数之差。请编程求出：第二行数中是否存在这样的若干个连续的整数，它们的和恰好是1898？假如存在的话，又有几种这样的情况？

\*\*输入格式要求：提示信息："There are follwing primes in first row :\n"

\*\*输出格式要求："(%d).%3d,......,%d\n"

程序运行示例如下：

There are follwing primes in first row :

(1).101,......,1999

(2). 89,......,1987

(3). 53,......,1951

(4). 3,......,1901

六十一、

#include <stdio.h>

#include <math.h>

#define NUM 320

int number[NUM];

main()

{

int i,j,count=0;

printf("There are follwing primes in first row :\n");

for(j=0,i=3;i<=1999;i+=2)

if (fflag(i)) number[j++]=i;

for(j--;number[j]>1898;j--)

{

for(i=0;number[j]-number[i]>1898;i++);

if(number[j]-number[i]==1898)

printf ("(%d).%3d,......,%d\n",++count,number[i],number[j]);

}

}

fflag(i)

int i;

{

int j;

if (i<=1) return(0);

if (i==2) return(1);

if(!(i%2)) return(0);

for(j=3;j<=(int)( sqrt((double)i)+1);j+=2)

if (!(i%j)) return(0);

return(1);

}

六十二、

利用求阶乘函数Fact()，编程计算并输出1! + 2! + …… + n!的值。

\*\*输入格式要求："%u" 提示信息："Input n(n>0):"

\*\*输出格式要求："sum = %lu\n

六十二、

#include <stdio.h>

unsigned long Fact(unsigned int n);

int main()

{

unsigned int i, n;

unsigned long sum = 0;

printf("Input n(n>0):");

scanf("%u", &n);

for (i = 1; i <= n; i++)

{

sum = sum + Fact(i);

}

printf("sum = %lu\n", sum);

return 0;

}

/\* 函数功能：用迭代法计算无符号整型变量n的阶乘 \*/

unsigned long Fact(unsigned int n)

{

unsigned int i;

unsigned long result = 1;

for (i = 2; i <= n; i++)

result \*= i;

return result;

}

1.利用指针能直接操作内存的特点来编程输出字符串中每个字符在内存中的存储编码（字符串中可以包含英文数字和汉字）。

程序的运行示例1如下：

请输入一个字符串，长度小于等于100：abcABC012中国人

该字符串的内存编码为： 61 62 63 41 42 43 30 31 32 d6 d0 b9 fa c8 cb

程序的运行示例2如下：

请输入一个字符串，长度小于等于100：12345上山打老虎

该字符串的内存编码为： 31 32 33 34 35 c9 cf c9 bd b4 f2 c0 cf bb a2

#include <stdio.h>

int main(int argc, char \*argv[])

{

    char str[101] = {'\0'};

    char \*p1, \*p2;

    printf("请输入一个字符串，长度小于等于100：");

    scanf("%s", str);

    p1 = (char \*)(str + strlen(str));

    p2 = (char \*)str;

    printf("该字符串的内存编码为： ");

    while (p1 > p2)

    {

        printf("%x ", (unsigned char)\*p2 & 0xff);

        p2++;

    }

    return 0;

}

2.按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。

其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息要求："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%5d"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

int main()

{

    int s[N][N], n;

    printf("Input n:\n");

    scanf("%d", &n);

    InputMatrix(s, n);//1

    Transpose(s, n);//1

    printf("The transposed matrix is:\n");

    PrintMatrix(s, n);//1

    return 0;

}

void Transpose(int a[][N], int n)

{

    int i, j, temp;

    for (i = 0; i < n; i++) //1

    {

        for (j = i; j < n; j++)//1

        {

            temp = a[i][j];//1

            a[i][j] = a[j][i];//1

            a[j][i] = temp;//1

        }

    }

}

void InputMatrix(int a[][N], int n)

{

    int i, j;

    printf("Input %d\*%d matrix:\n", n, n);

    for (i = 0; i < n; i++)

    {

        for (j = 0; j < n; j++)

        {

            scanf("%d", &a[i][j]); //1

        }

    }

}

void PrintMatrix(int a[][N], int n)

{

    int i, j;

    for (i = 0; i < n; i++) //1

    {

        for (j = 0; j < n; j++)  //1

        {

            printf("%5d", a[i][j]); //1

        }

        printf("\n");    //1

    }

}

3.以下程序的功能是计算十个数据的平均值。

#include <stdio.h>

void main(void)

{

    int i, sum=0;

float avg;

int sc[10], \*p=sc;

for (i=0; i<10; i++)

{

        scanf("%d", p);

        p++;

        sum+= \*(p-1);

}

avg = (float)sum/ 10;

printf("avg=%f", avg);

}

4.编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

#include <stdio.h>

#include <stdlib.h>

#define N 100

int getline(char \*s, int len);

int main()

{

    char line[N], b[20], \*ptr1, \*ptrb;

    int n = 0, j, a[50];

    printf("Please enter a string:");

    getline(line, N);

    ptr1 = line;

    while (\*ptr1 != '\0')

    {

        ptrb = b;

        for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)

        {

            \*ptrb = \*ptr1;

        }

        if (j)

        {

            \*ptrb = '\0';

            \*(a + n) = atoi(b);

            n++;

        }

        if (\*ptr1 != '\0')ptr1++;

    }

    printf("the result of output:\n");

    for (j = 0; j < n; j++)

    {

        printf("%10d\n", \*(a + j));

    }

    return 0;

}

int getline(char \*s, int len)

{

    int c;

    char \*p = s;

    while (--len > 0 && (c = getchar()) != '\n')

    {

        \*s++ = c;

    }

    \*s = '\0';

    return s - p;

}

5.把如下两个无序数组a和b从小到大排序后，在按照从小到大的顺序一次存放到新的数组中。

int a[5]={9,78,33,12,23};

int b[8]={1,34,63,10,5,94,39,27};

要求输出在主函数中实现，排序和数组合并分别用函数实现，函数原型如下：

void sort(int a[], int n);

void addins(int a[], int b[], int c[], int n, int m);

\*\*输出格式要求："%4d"

#include <stdio.h>

void sort(int a[], int n)

{

    int i, j, t, \*p;

    for (i = 0; i < n - 1; i++)

    {

        p = a + i;

        for (j = i + 1; j < n; j++)

        {

            if (\*p > \*(a + j))

                p = a + j;

        }

        if (p != (a + i))

        {

            t = \*(a + i);

            \*(a + i) = \*p;

            \*p = t;

        }

    }

}

void addins(int a[], int b[], int c[], int n, int m)

{

    int i, j, k, l;

    int \*p1 = a, \*p2 = b , \*p3 = c;

    i = j = k = 0;

    while (i < n && j < m)

    {

        if (\*(p1 + i) <= \*(p2 + j))

        {

            \*(p3 + k) = \*(p1 + i);

            i++;

        }

        else

        {

            \*(p3 + k) = \*(p2 + j);

            j++;

        }

        k++;

    }

    if (i == n)

    {

        for (l = j; l < m; l++)

        {

            \*(p3 + k) = \*(p2 + l);

            k++;

        }

    }

    if (j == m)

    {

        for (l = i; l < n; l++)

        {

            \*(p3 + k) = \*(p1 + l);

            k++;

        }

    }

}

void main()

{

    int a[5] = {9, 78, 33, 12, 23};

    int b[8] = {1, 34, 63, 10, 5, 94, 39, 27};

    int c[13];

    int i;

    int \*p1 = a, \*p2 = b, \*p3 = c;

    sort(a, 5);

    for (i = 0; i < 5; i++)

    {

        printf("%4d", \*(p1 + i));

    }

    printf("\n");

    sort(b, 8);

    for (i = 0; i < 8; i++)

    {

        printf("%4d", \*(p2 + i));

    }

    printf("\n");

    addins(b, a, c, 8, 5);

    for (i = 0; i < 13; i++)

    {

        printf("%4d", \*(p3 + i));

    }

    printf("\n");

}

6.有n个人围成一圈，顺序排号。从第一个人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

\*\*输入格式要求："%d" 提示信息："please input the total of numbers:"

\*\*输出格式要求："%d is left\n"

#include <stdio.h>

#define nmax 50

int main()

{

    int i, k, m, n, num[nmax], \*p;

    printf("please input the total of numbers:");

    scanf("%d", &n);

    p = num;

    for (i = 0; i < n; i++ )

        \*(p + i) = i + 1;

    i = 0;

    k = 0;

    m = 0;

    while (m < n)

    {

        if (\*(p + i) != 0)

            k++;

        if (k % 3 == 0)

        {

            \*(p + i) = 0;

            m++;

        }

        i++;

        if (i == n)

            i = 0;

    }

    while (\*p == 0)

        p++;

    printf("%d is left\n", \*p);

}

7.在一个字符串中找出最大的字符并放在第一个位置上，并将该字符前的原字符往后顺序移动，如：cbyab变成ycbab。

\*\*输入格式要求：用gets读入

\*\*输出格式要求：用puts输出

#include <stdio.h>

main()

{

    char str[80], \*p = str, \*q, max;

    gets(p);

    max = \*(p++);

    while (\*p != 0)

    {

        if (max < \*p)

        {

            max = \*p;

            q = p;

        }

        p++;

    }

    p = q;

    while (p > str)

    {

        \*p = \*(p - 1);

        p--;

    }

    \*p = max;

    puts(p);

}

8.按如下函数原型编写程序，从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

void Swap(int \*x, int \*y);

输入提示信息：

"Input n(n<=10):"

"Input %d numbers:"

输入格式："%d"

输出提示信息：

"Exchange results:"

输出格式："%5d"

#include  <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void  MaxMinExchang(int a[], int n);

void Swap(int \*x, int \*y);

int main()

{

    int  a[10], n;

    printf("Input n(n<=10):");

    scanf("%d", &n);

    printf("Input %d numbers:", n);

    ReadData(a, n);

    MaxMinExchang(a, n);

    printf("Exchange results:");

    PrintData(a, n);

    return 0;

}

/\* 函数功能：输入数组a的n个元素值 \*/

void ReadData(int a[], int n)

{

    int i;

    for (i = 0; i < n; i++)

    {

        scanf("%d", &a[i]);

    }

}

/\* 函数功能：输出数组a的n个元素值 \*/

void PrintData(int a[], int n)

{

    int i;

    for (i = 0; i < n; i++)

    {

        printf("%5d", a[i]);

    }

    printf("\n");

}

/\* 函数功能：将数组a中的最大数与最小数位置互换 \*/

void  MaxMinExchang(int a[], int n)

{

    int  maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;

    int  i;

    for (i = 1; i < n; i++)

    {

        if (a[i] > maxValue)

        {

            maxValue = a[i];

            maxPos = i;

        }

        if (a[i] < minValue)

        {

            minValue = a[i];

            minPos = i;

        }

    }

    Swap(&a[maxPos], &a[minPos]);

}

/\* 函数功能：两整数值互换 \*/

void  Swap(int \*x, int \*y)

{

    int  temp;

    temp = \*x;

    \*x = \*y;

    \*y = temp;

}

9.下面程序的功能是将数组元素倒置

#include "stdio.h"

#define M 5

main()

{   int a[M]={1,2,3,4,5};

    int i,j,t;

    i=0;j=M-1;

    while(i<=j)

    {

        t=\*(a+i);

        \*(a+i)=\*(a+j);

        \*(a+j)=t;

        i++;j--;

    }

    for(i=0;i<M;i++) printf("%d",\*(a+i));

}

10.指针和字符数组的应用,用五种方法输出字符串“China”。

#include <stdio.h>

main()

{

    int i = 0;

    char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[6];

    while (str[i] != '\0')

    {

        putchar(str[i]);

        i++;

    }

    ptr = str1;

    scanf("%s", ptr);

    puts(ptr);

    ptr1 = str;

    puts(ptr1);

    ptr2 = "China";

    puts(ptr2);

    strcpy(str2, "China");

    printf("%s", str2);

}

11.请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);

\*\*输入格式要求：无

\*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身

#include <stdio.h>

#include <string.h>

int  Judger( char  a[ ]);

int main()

{

    char a[80];

    int flag;

    gets(a);

    flag = Judger(a);

    if (flag == 1)

    {

        puts(a);

    }

    else

    {

        printf("No!");

    }

    return 0;

}

int  Judger( char  a[ ])

{

    char \*pStar, \*pEnd;

    int i, j, flag = 1;

    j = strlen(a);

    pStar = &a[0];

    pEnd = &a[j - 1];

    for (i = 0; i < (j / 2); i++)

    {

        if (\*pStar != \*pEnd)

        {

            flag = 0;

        }

        pStar++;

        pEnd--;

    }

    return flag;

}

12.有n个人围成一圈，顺序排号。从第一人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

程序的运行示例如下：

请输入人数：30

30个人中最后留下的是第29号。

#include <stdio.h>

#define     NMAX    50 /\* 定义最多人数为50人 \*/

int main(void)

{

    int i, k, m, n, num[NMAX], \*p;

    printf("请输入人数：");

    scanf("%d", &n);

    p = num;

    for (i = 0; i < n; i++)

    {

        \*(p + i) = i + 1; /\* 以1至n为序给每个人编号 \*/

    }

    i = 0;

    k = 0; /\* k为按1,2,3报数时的计数变量 \*/

    m = 0; /\* m为退出人数 \*/

    while (m < n - 1)

    {

        if (\*(p + i) != 0)

        {

            k++;

        }

        if (k == 3)

        {

            \*(p + i) = 0; /\* 对退出人员的编号置为0 \*/

            k = 0;

            m++;

        }

        i++;

        if (i == n)

        {

            i = 0; /\* 报数到尾后i恢复为0 \*/

        }

    }

    while (\*p == 0)

    {

        p++;

    }

    printf("%d个人中最后留下的是第%d号。", n, \*p);

    return 0;

}

1.定义一个结构体变量（包括年、月、日）。计算该日在本年中是第几天？注意闰年问题。

#include <stdio.h>

struct

{

    int year;

    int month;

    int day;

} date;

int main(void)

{

    int days;

    printf("请输入日期（年，月，日）\n");

    scanf("%d,%d,%d", &date.year, &date.month, &date.day);

    switch (date.month)

    {

    case 1:

        days = date.day;

        break;

    case 2:

        days = date.day + 31;

        break;

    case 3:

        days = date.day + 59;

        break;

    case 4:

        days = date.day + 90;

        break;

    case 5:

        days = date.day + 120;

        break;

    case 6:

        days = date.day + 151;

        break;

    case 7:

        days = date.day + 181;

        break;

    case 8:

        days = date.day + 212;

        break;

    case 9:

        days = date.day + 243;

        break;

    case 10:

        days = date.day + 273;

        break;

    case 11:

        days = date.day + 304;

        break;

    case 12:

        days = date.day + 334;

        break;

    }

    if ((date.year % 4 == 0 && date.year % 100 != 0

            || date.year % 400 == 0) && date.month >= 3)

    {

        days += 1;

    }

    printf("\n%d月%d日是%d年的第%d天。", date.month, date.day, date.year, days);

    return 0;

}

2.编程统计候选人的得票数。设有3个候选人，每次输入一个得票候选人的名字，不考虑弃权情况，要求最后输出各个候选人的得票结果（候选人姓名和参加投票人数由程序运行时输入）。

要求输入、输出在主函数中实现，统计候选人由函数实现。

按如下函数原型编写程序

void Vote(struct Person ld[], int n)

#include <stdio.h>

#define N 3

struct Person

{

    char name[16];

    int count;

};

void Input(struct Person ld[], int n);  //输入n个候选人的基本信息

void Vote(struct Person ld[], int n);   //对n个候选人进行投票

void Output(struct Person ld[], int n); //输出n个候选人的基本信息

int main(int argc, char \*argv[])

{

    struct Person leader[N];

    Input(leader, N);

    Vote(leader, N);

    Output(leader, N);

    return 0;

}

void Input(struct Person ld[], int n)

{

    int i;

    printf("输入%d个候选人的基本信息：姓名\n", n);

    for (i = 0; i < n; i++)

        scanf("%s", ld[i].name);

}

void Vote(struct Person ld[], int n)

{

    int m, i, j; //m表示参加投票的人数

    char ename[16];

    printf("输入参加投票的人数:");

    scanf("%d", &m);

    printf("输入%d个参加投票人的投票情况\n", m);

    for (i = 0; i < m; i++)

    {

        ld[i].count = 0;

    }

    for (i = 0; i < m; i++)

    {

        scanf("%s", ename);

        for (j = 0; j < n; j++)

            if (strcmp(ld[j].name, ename) == 0)

                ld[j].count++;

    }

}

void Output(struct Person ld[], int n)

{

    int i;

    printf("输出%d个候选人的基本信息：姓名，票数\n", n);

    for (i = 0; i < n; i++)

        printf("%s %d\n", ld[i].name, ld[i].count);

}

3.中国有句俗语叫“三天打鱼两天晒网”，某人从90年1月1日起开始“三天打鱼两天晒网”。问这个人在以后的某一天中是在“打渔”，还是在“晒网”.

\*\*输入格式要求："%d%d%d" 提示信息："Enter year/month/day:"

\*\*输出格式要求："He is fishing.\n" "He is sleeping.\n"

#include <stdio.h>

struct date

{

  int year,month,day;

};

main()

{

  struct date today,term;

  int yeardays=0,day;

  printf("Enter year/month/day:");

  scanf("%d%d%d",&today.year,&today.month,&today.day);

  term.month=12;

  term.day=31;

  for(term.year=1990;term.year>today.year;term.year++)

      yeardays+=days(term);

  yeardays+=days(today);

  day=yeardays%5;

  if(day>0&&day<4)

        printf("He is fishing.\n");

    else

        printf("He is sleeping.\n");

}

  days(day)

    struct date day;

    {

      static int day\_tab[2][13]={{0,31,28,31,30,31,30,31,31,30,31,30,31},

                                 {0,31,29,31,30,31,30,31,31,30,31,30,31}};

      int i,lp;

      lp=day.year%4==0&&day.year%100!=0||day.year%400==0;

      for(i=1;i<day.month;i++)

         day.day+=day\_tab[lp][i];

       return (day.day);

    }

4. 写一个函数days，实现主函数将年、月、日（结构体类型）传递给days函数，days函数计算该年该月该日是该年的第几天并传回主函数输出。

#include <stdio.h>

struct dt

{

    int year;

    int month;

    int day;

} date;

int days(struct dt date)

{

    int sum;

    switch (date.month)

    {

    case 1:

        sum = date.day;

        break;

    case 2:

        sum = date.day + 31;

        break;

    case 3:

        sum = date.day + 59;

        break;

    case 4:

        sum = date.day + 90;

        break;

    case 5:

        sum = date.day + 120;

        break;

    case 6:

        sum = date.day + 151;

        break;

    case 7:

        sum = date.day + 181;

        break;

    case 8:

        sum = date.day + 212;

        break;

    case 9:

        sum = date.day + 243;

        break;

    case 10:

        sum = date.day + 273;

        break;

    case 11:

        sum = date.day + 304;

        break;

    case 12:

        sum = date.day + 334;

        break;

    }

    if ((date.year % 4 == 0 && date.year % 100 != 0

            || date.year % 400 == 0) && date.month >= 3)

    {

        sum += 1;

    }

    return sum;

};

int main(void)

{

    printf("请输入日期（年，月，日）\n");

    scanf("%d,%d,%d", &date.year, &date.month, &date.day);

    printf("\n%d月%d日是%d年的第%d天。", date.month, date.day, date.year, days(date));

    return 0;

}

5. 13个人围成一圈，从第1个人开始顺序报号1、2、3，凡报到3的人退出圈子。找出最后留在圈子里的人原来的序号。要求用结构体编程实现。

#include <stdio.h>

#define     N   13

struct person

{

    int number;

    int nextp;

} link[N + 1];

int main(void)

{

    int i, count, h;

    /\* 建立队列 \*/

    for (i = 1; i <= N; i++)

    {

        if (i == N)

        {

            link[i].nextp = 1;

        }

        else

        {

            link[i].nextp = i + 1;

        }

        link[i].number = i;

    }

    printf("\n");

    count = 0;

    h = N;

    printf("出圈成员及顺序：");

    while (count < N - 1)

    {

        i = 0;

        while (i != 3)

        {

            h = link[h].nextp;

            if (link[h].number)

            {

                i++;

            }

        }

        printf("%3d", link[h].number);

        link[h].number = 0;

        count++;

    }

    printf("\n最后的成员是：");

    for (i = 1; i <= N; i++)

    {

        if (link[i].number)

        {

            printf("%3d", link[i].number);

        }

    }

    return 0;

}

6. 编程模拟洗牌和发牌过程。一副扑克有52张牌，分为4种花色（suit）：黑桃（Spades）、红桃（Hearts）、草花（Clubs）、方块（Diamonds）。每种花色又有13张牌面（face）：A，2，3，4，5，6，7，8，9，10，Jack，Queen，King。要求用结构体数组card表示52张牌，每张牌包括花色和牌面两个字符型数组类型的数据成员。

#include <stdio.h>

#include <string.h>

#include <time.h>

#include <stdlib.h>

typedef struct card

{

    char  suit[10];

    char  face[10];

} CARD;

void Deal(CARD \*wCard);

void Shuffle(CARD \*wCard);

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[]);

int main()

{

    char \*suit[] = {"Spades", "Hearts", "Clubs", "Diamonds"};

    char \*face[] = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10",

                    "Jack", "Queen", "King"

                   };

    CARD card[52];

    srand (time(NULL));

    FillCard(card, face, suit);

    Shuffle(card);

    Deal(card);

    return 0;

}

/\*  函数功能：花色按黑桃、红桃、草花、方块的顺序，面值按A~K的顺序，排列52张牌 \*/

void  FillCard(CARD wCard[], char \*wFace[], char \*wSuit[])

{

    int i;

    for (i = 0; i < 52; i++)

    {

        strcpy(wCard[i].suit, wSuit[i / 13]);

        strcpy(wCard[i].face, wFace[i % 13]);

    }

}

/\*  函数功能：将52张牌的顺序打乱以模拟洗牌过程 \*/

void Shuffle(CARD \*wCard)

{

    int i, j;

    CARD temp;

    for (i = 0; i < 52; i++) /\*每次循环产生一个随机数，交换当前牌与随机数指示的牌\*/

    {

        j = rand() % 52;   /\* 每次循环产生一个0~51的随机数 \*/

        temp = wCard[i];

        wCard[i] = wCard[j];

        wCard[j] = temp;

    }

}

/\*  函数功能：输出每张牌的花色和面值以模拟发牌过程 \*/

void Deal(CARD \*wCard)

{

    int i;

    for (i = 0; i < 52; i++)

    {

        printf("%9s%9s%c", wCard[i].suit, wCard[i].face, i % 2 == 0 ? '\n' : '\t');

    }

    printf("\n");

}

7. 模拟人工洗牌。编写一个模拟人工洗牌的程序，讲洗好的牌分别发给四个人。

\*\*输出格式要求："\nNo.%d:" " %c%2d,"

#include <stdio.h>

#include <stdlib.h>

struct card

{

  int pips;

  char suit;

};

struct card deck[]={

                    {1,'C'},{2,'C'},{3,'C'},{4,'C'},{5,'C'},{6,'C'},{7,'C'},

                    {8,'C'},{9,'C'},{10,'C'},{11,'C'},{12,'C'},{13,'C'},

                    {1,'D'},{2,'D'},{3,'D'},{4,'D'},{5,'D'},{6,'D'},{7,'D'},

                    {8,'D'},{9,'D'},{10,'D'},{11,'D'},{12,'D'},{13,'D'},

                    {1,'H'},{2,'H'},{3,'H'},{4,'H'},{5,'H'},{6,'H'},{7,'H'},

                    {8,'H'},{9,'H'},{10,'H'},{11,'H'},{12,'H'},{13,'H'},

                    {1,'S'},{2,'S'},{3,'S'},{4,'S'},{5,'S'},{6,'S'},{7,'S'},

                    {8,'S'},{9,'S'},{10,'S'},{11,'S'},{12,'S'},{13,'S'},

                    };

 shuffle(deck)

   struct card deck[];

 {

   int i,j;

   srand(time(NULL));

   for(i=0;i<52;i++)

   {

     j=rand()%52;

     swapcard(&deck[i],&deck[j]);

   }

 }

 swapcard(p,q)

   struct card \*p,\*q;

 {

   struct card temp;

   temp=\*p;

   \*p=\*q;

   \*q=temp;

 }

 main()

 {

   int i;

   shuffle(deck);

   for(i=0;i<52;i++)

     if(i%13==0)

        printf("\nNo.%d:",i/13+1);

      else

        printf(" %c%2d,",deck[i].suit,deck[i].pips);

 }

8. 完成对输入的字符串中C关键词的查找统计。

#include <stdio.h>

#include <ctype.h>

#include <string.h>

#define MAXWORD 80

#define NKEYS (sizeof(keytab) / sizeof(struct key))

struct key

{

    char \*word;

    int count;

} keytab[] =

{

    "auto", 0,

    "break", 0,

    "case", 0,

    "char", 0,

    "const", 0,

    "continue", 0,

    "default", 0,

    "do", 0,

    "double", 0,

    "else", 0,

    "enum", 0,

    "extern", 0,

    "float", 0,

    "for", 0,

    "goto", 0,

    "if", 0,

    "int", 0,

    "long", 0,

    "register", 0,

    "return", 0,

    "short", 0,

    "singed", 0,

    "sizeof", 0,

    "static", 0,

    "struct", 0,

    "switch", 0,

    "typedef", 0,

    "union", 0,

    "unsigned", 0,

    "void", 0,

    "volatile", 0,

    "while", 0

};

void getword(char \*, int);

int binsearch(char \*, struct key \*, int);

int main(int argc, char \*argv[])

{

    int n;

    char word[MAXWORD];

    printf("本程序将为您统计C语言的关键字的个数，请输入，输入end结束输入：\n");

    do

    {

        getword(word, MAXWORD);

        if (isalpha(word[0]))

        {

            if ((n = binsearch(word, keytab, NKEYS)) >= 0)

            {

                keytab[n].count++; //找到则对应次数+1

            }

        }

    }

    while (strcmp(word, "end") != 0);

    printf("您的输入中C语言关键字出现的次数统计如下：\n");

    for (n = 0; n < NKEYS; n++)

    {

        if (keytab[n].count > 0)

        {

            printf("%-10s: %6d\n", keytab[n].word, keytab[n].count);

        }

    }

    return 0;

}

/\* 折半查找：在tab[0]到tab[n-1]中查找word \*/

int binsearch(char \*word, struct key tab[], int n)

{

    int result;

    int low, high, mid;

    low = 0;

    high = n - 1;

    while (low <= high)

    {

        mid = (low + high) / 2;

        if ((result = strcmp(word, tab[mid].word)) < 0)

        {

            high = mid - 1;

        }

        else if (result > 0)

        {

            low = mid + 1;

        }

        else

        {

            return mid;

        }

    }

    return -1;

}

/\* getword：从输入中获取某个单词 \*/

void getword(char \*word, int lim)

{

    int c;

    void ungetch(int);

    char \*w = word;

    while (isspace(c = getchar()))

    {

    }

    if (c != EOF)

    {

        \*w = c;

        w++;

    }

    if (!isalpha(c))

    {

        \*w = '\0';

    }

    for ( ; --lim > 0; w++)

    {

        if (!isalnum(\*w = getchar()))

        {

            //读入的某个字符不是字母，则将它退还给输入缓冲区

            ungetch(\*w);

            break;

        }

    }

    \*w = '\0';

}

9.以下程序用于打印输出结构体数组的所有元素

struct s

{

    char name[10];

    int age;

};

main()

{

    struct s a[3] = {"John",19,"Paul",17,"Marry",18};

    struct s \*p;

    for(p = a; p < a+3; p++)

    {

        printf("%s,%d\n", p->name, p->age );

    }

}

10.编程统计候选人的得票数。设有3个候选人zhang、li、wang，10个选民，选民每次输入一个得票的候选人的名字，若选民输错候选人姓名，则按废票处理。选民投票结束后程序自动显示各候选人的得票结果和废票信息。要求用结构体数组candidate表示3个候选人的姓名和得票结果。

#include  <stdio.h>

#include  <string.h>

#define NUM\_ELECTORATE 10

#define NUM\_CANDIDATE 3

struct candidate

{

    char  name[20];

    int   count;

} candidate[3] = {"li", 0, "zhang", 0, "wang", 0};

int main()

{

    int  i, j, flag = 1, wrong = 0;

    char  name[20];

    for (i = 1; i <= NUM\_ELECTORATE; i++)

    {

        printf("Input vote %d:", i);

        scanf("%s", name);

        flag = 1;

        for (j = 0; j < NUM\_CANDIDATE; j++)

        {

            if (strcmp(name, candidate[j].name) == 0)

            {

                candidate[j].count++;

                flag = 0;

            }

        }

        if (flag)

        {

            wrong++;  /\* 废票计数 \*/

            flag = 0;

        }

    }

    printf("Election results:\n");

    for (i = 0; i < NUM\_CANDIDATE; i++)

    {

        printf("%8s:%d\n", candidate[i].name, candidate[i].count);

    }

    printf("Wrong election:%d\n", wrong);

    return 0;

}

11. \*约瑟夫问题。这是十七世纪的法国数学家加斯帕在《数目的游戏问题》中讲的一个故事：15个基督教徒和15个异教徒在海上遇险，必须将一半的人投入海中，其余的人才能幸免于难，于是想了个办法：30个人围成一个圆圈，从第一个人开始依次报数，每数到第9个人就将他扔入大海，如此循环进行直到仅余15个人为止。问怎样排法，才能使每次投入大海的都是异教徒。

\*\*输出格式要求："The original circle is (+:papandom, @:christian);\n" "%c "

程序运行示例如下：

The original circle is (+:papandom, @:christian);

@ @ @ @ + + + + + @ @ + @ @ @ + @ + + @ @ + + + @ + + @ @ +

#include <stdio.h>

struct node

{

int nextp;

int no\_out;

}link[31];

main()

{

int i,j,k;

printf("The original circle is (+:papandom, @:christian);\n");

for(i=1;i<=30;i++)

{

link[i].nextp=i+1;

link[i].no\_out=1;

}

link[30].nextp=1;

j=30;

for(i=0;i<15;i++)

{

for(k=0;;)

if(k<9)

{

j=link[j].nextp;

k+=link[j].no\_out;

}

else break;

link[j].no\_out=0;

}

for(i=1;i<=30;i++)

printf("%c ",link[i].no\_out?'@':'+');

printf("\n");

}

12. 以下程序有若干语法错误。请找出并改正之。

#include <stdio.h>

struct date

{

    int  y,m,d;

};

struct stu

{   char n[10];

    struct date b;

    int a;

}s={"Wang",{1980,11,11},30};

main()

{

    printf("%s,%d,%d\n",s.n,s.b.d,s.a);

}

13. 求复数之积。利用结构变量求解两个复数之积。

za={3,4}, zb={5,6}

za={10,20}, zb={30,40}

\*\*输出格式要求："(%d+%di)\*(%d+%di)=" "(%d+%di)\n"

#include "stdio.h"

struct complx

{

  int real;

  int im;

};

main()

{

  static struct complx za={3,4};

  static struct complx zb={5,6};

  struct complx x,y,z;

  struct complx cmult();

  void cpr();

  z=cmult(za,zb);

  cpr(za,zb,z);

  x.real = 10; x.im = 20;

  y.real = 30; y.im = 40;

  z=cmult(x,y);

  cpr(x,y,z);

}

struct complx cmult (za,zb)

  struct complx za,zb;

{

      struct complx w;

      w.real = za.real\*zb.real - za.im\*zb.im;

      w.im   = za.real\*zb.im+za.im\*zb.real;

      return (w);

}

void cpr(za,zb,z)

   struct complx za,zb,z;

{

    printf("(%d+%di)\*(%d+%di)=",za.real,za.im,zb.real,zb.im);

    printf("(%d+%di)\n",z.real,z.im);

}

14. 对候选人得票的统计程度。设由3个候选人，每次输入一个得票的候选人的名字，要求最后输入各人得票结果。（总票数为n）请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

struct person

{

    char name[20];

    int count;

};

struct person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

    int i, j, n;

    char leader\_name[20];

    scanf("%d\n", &n);

    for (i = 0; i < n; i++)

    {

        scanf("%s", leader\_name);

        for (j = 0; j < 3; j++)

            if (!strcmp(leader\_name, leader[j].name))

                leader[j].count++;

    }

    printf("\n");

    for (i = 0; i < 3; i++)

        printf("%5s:%d\n", leader[i].name, leader[i].count);

}

15.

Q501.(25分)用结构体实现输入4个学生的学号（无符号整数unsigned int）、姓名（字符串）和考试成绩（整数），然后按从高分到低分的顺序排列后输出。

程序的运行示例如下：

输入第1个学生的信息：学号、姓名、成绩

2011 zhangsan 89

输入第2个学生的信息：学号、姓名、成绩

2012 lisi 96

输入第3个学生的信息：学号、姓名、成绩

2013 wangwu 67

输入第4个学生的信息：学号、姓名、成绩

2014 liuliu 59

排序后学生的信息为:

2012 lisi 96

2011 zhangsan 89

2013 wangwu 67

2014 liuliu 59

答案：

#include <stdio.h>

#define N 4

struct Student1

{

    unsigned int sno;

    char sname[16];

    int score;

};

int main(int argc, char \*argv[])

{

    struct Student1 stu[N];

    int i, j;

    struct Student1 temp;

    for (i = 0; i < N; i++)

    {

        printf("输入第%d个学生的信息：学号、姓名、成绩\n", i + 1);

        scanf("%d %s %d", &stu[i].sno, stu[i].sname, &stu[i].score);

    }

    for (i = 0; i < N - 1; i++)

    {

        for (j = 0; j < N - i - 1; j++)

        {

            if (stu[j].score < stu[j + 1].score)

            {

                temp = stu[j];

                stu[j] = stu[j + 1];

                stu[j + 1] = temp;

            }

        }

    }

    printf("排序后学生的信息为:\n");

    for (i = 0; i < N; i++)

    {

        printf("%d %s %d\n", stu[i].sno, stu[i].sname, stu[i].score);

    }

    return 0;

}

Q493.(10分)输入6个字符串，输出最小串及最大串。

\*\*输入格式要求："%s" 提示信息："请输入6行字符串：\n"

\*\*输出格式要求："The max string is: %s\n" "The min string is: %s\n"

程序示例运行如下：

请输入6行字符串：

hello,world

vb

vc

Java

c++

c#

The max string is: vc

The min string is: Java

答案：

#include <stdio.h>

#include <string.h>

#define N 6

#define N1 20

int main()

{

    char str[N1], min[N1], max[N1];

    int i;

    printf("请输入6行字符串：\n");

    gets(min);  //假设第1个串最小

    strcpy(max, min);//假设最大串的值也为min

    //循环输入其它串，并与最小的串及最大的串比较

    for (i = 2; i <= N; i++)

    {

        gets(str);

        if (strcmp(str, min) < 0)

            strcpy(min, str);

        if (strcmp(str, max) > 0)

            strcpy(max, str);

    }

    printf("The max string is: %s\n", max);

    printf("The min string is: %s\n", min);

    return 0;

}

Q382.(10分)\*约瑟夫问题。这是十七世纪的法国数学家加斯帕在《数目的游戏问题》中讲的一个故事：15个基督教徒和15个异教徒在海上遇险，必须将一半的人投入海中，其余的人才能幸免于难，于是想了个办法：30个人围成一个圆圈，从第一个人开始依次报数，每数到第9个人就将他扔入大海，如此循环进行直到仅余15个人为止。问怎样排法，才能使每次投入大海的都是异教徒。

\*\*输出格式要求："The original circle is (+:papandom, @:christian);\n" "%c "

程序运行示例如下：

The original circle is (+:papandom, @:christian);

@ @ @ @ + + + + + @ @ + @ @ @ + @ + + @ @ + + + @ + + @ @ +

答案：

#include <stdio.h>

struct node

{

  int nextp;

  int no\_out;

}link[31];

main()

{

  int i,j,k;

  printf("The original circle is (+:papandom,  @:christian);\n");

  for(i=1;i<=30;i++)

  {

    link[i].nextp=i+1;

    link[i].no\_out=1;

  }

  link[30].nextp=1;

  j=30;

  for(i=0;i<15;i++)

  {

     for(k=0;;)

     if(k<9)

     {

       j=link[j].nextp;

       k+=link[j].no\_out;

     }

     else break;

     link[j].no\_out=0;

  }

  for(i=1;i<=30;i++)

        printf("%c ",link[i].no\_out?'@':'+');

    printf("\n");

}

Q1327.(10分)有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。

程序的运行示例如下：

请输入一个字符串：1234567890

从第几个字符开始复制？3

复制结果为：34567890

答案：

#include <stdio.h>

copystr(char \*p1, char \*p2, int m)

{

    int n;

    n = 0;

    while (n < m - 1)

    {

        n++;

        p1++;

    }

    while (\*p1 != '\0')

    {

        \*p2 = \*p1;

        p1++;

        p2++;

    }

    \*p2 = '\0';

}

int main(void)

{

    int m;

    char str1[20], str2[20];

    printf("请输入一个字符串：\n");

    scanf("%s", str1);

    printf("从第几个字符开始复制？");

    scanf("%d", &m);

    if (strlen(str1) < m)

    {

        printf("输入错误。");

    }

    else

    {

        copystr(str1, str2, m);

        printf("复制结果为：%s", str2);

    }

    return 0;

}

Q1636.(10分)按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char dstStr[], char srcStr[]);

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

答案：

#include <stdio.h>

#include <string.h>

#define N 80

void MyStrcat(char dstStr[], char srcStr[]);

int main()

{

    char  s[N], t[N];

    printf("Input a string:");

    gets(s);

    printf("Input another string:");

    gets(t);

    MyStrcat(s, t);

    printf("Concatenate results:%s\n", s);

    return 0;

}

/\*  函数功能：将字符串srcStr连接到字符串dstStr之后 \*/

void MyStrcat(char dstStr[], char srcStr[])

{

    unsigned int  i, j;

    i = strlen(dstStr);               /\* 将下标移动到字符串dstStr的末尾 \*/

    for (j = 0; j <= strlen(srcStr); j++, i++)

    {

        dstStr[i] = srcStr[j];

    }

}

写一函数，输入一个十六进制数，输出相应的十进制数。

程序的运行示例如下：

输入十六进制数：

3a

十进制数58

继续吗？

y

输入十六进制数：

2f

十进制数47

继续吗？

N

#include <stdio.h>

#define     MAX     1000

int main()

{

    int c, i, flag, flag1;

    char t[MAX];

    i = 0;

    flag = 0;

    flag1 = 1;

    printf("\n输入十六进制数：");

    while ((c = getchar()) != '\0' && i < MAX && flag1)

    {

        if (c >= '0' && c <= '9' || c >= 'a' && c <= 'f' || c >= 'A' && c <= 'F')

        {

            flag = 1;

            t[i++] = c;

        }

        else if (flag)

        {

            t[i] = '\0';

            printf("\n十进制数%d\n", htoi(t));

            printf("继续吗？");

            c = getchar();

            if (c == 'N' || c == 'n')

            {

                flag1 = 0;

            }

            else

            {

                flag = 0;

                i = 0;

                printf("\n输入十六进制数：");

            }

        }

    }

    return 0;

}

htoi(char s[])

{

    int i, n;

    n = 0;

    for (i = 0; s[i] != '\0'; i++)

    {

        if (s[i] >= '0' && s[i] <= '9')

        {

            n = n \* 16 + s[i] - '0';

        }

        if (s[i] >= 'a' && s[i] <= 'f')

        {

            n = n \* 16 + s[i] - 'a' + 10;

        }

        if (s[i] >= 'A' && s[i] <= 'F')

        {

            n = n \* 16 + s[i] - 'A' + 10;

        }

    }

    return n;

}

Q301.(5分)用指针做函数参数自己编程实现字符串连接函数strcat()的功能。

下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

经教师手工核对后，如果未用指针做函数参数编程，那么即使做对也不给分。

改错时不能改变程序原有的意图，不能改变函数原型。

|  |
| --- |
| #include <stdio.h>  #define N = 80;  void MyStrcat(char \*dstStr, char \*srcStr);  main()  {      char s[N], t[N];      printf("Input a string:\n");      gets(s);      printf("Input another string:\n");      gets(t);      MyStrcat(s, t);      printf("Concatenate results:%s\n", s);  }  void MyStrcat(char \*dstStr, char \*srcStr)  {      while (\*srcStr != '\0');      {          srcStr++;      }      while (\*srcStr != '\0');      {          \*dstStr = \*srcStr;          srcStr++;          dstStr++;      }  } |

Q295.(15分)编程实现找出字符串中最大字符元素并输出该元素及其对应的ASCII值.

\*\*\*\*要求输入提示信息为：

"Input a string:\n"

\*\*\*\*输出格式要求为：

"The largest character of \"%s\" is \'%c\' ,The ASCII is %d."

答案：

#include <stdio.h>

#include <stdlib.h>

main()

{

    char string[81];

    int i=0,max;

    printf("Input a string:");  //1

    printf("\n");

    gets(string);

    max=string[i];      //1

    while(string[i]!='\0')      //2

    {

        i=i+1;

        if(string[i]>max)        //2

           max=string[i];       //1

    }

    printf("The largest character of \"%s\" is \'%c\' ,The ASCII is %d.",string,max,max);   //2

}

一．

身高预测。每个做父母的都关心自己孩子成人后的身高，据有关生理卫生知识与数理统计分析表明，影响小孩成人后的身高的因素包括遗传、饮食习惯与体育锻炼等。小孩成人后的身高与其父母的身高和自身的性别密切相关。

设faHeight为其父身高，moHeight为其母身高，身高预测公式为

男性成人时身高 = (faHeight + moHeight) × 0.54 cm

女性成人时身高 = (faHeight × 0.923 + moHeight) / 2 cm

此外，如果喜爱体育锻炼，那么可增加身高2%；如果有良好的卫生饮食习惯，那么可增加身高1.5%。

请编程从键盘输入用户的性别（用字符型变量sex存储，输入字符F表示女性，输入字符M表示男性）、父母身高（用实型变量存储，faHeight为其父身高，moHeight为其母身高）、是否喜爱体育锻炼（用字符型变量sports存储，输入字符Y表示喜爱，输入字符N表示不喜爱）、是否有良好的饮食习惯等条件（用字符型变量diet存储，输入字符Y表示良好，输入字符N表示不好），利用给定公式和身高预测方法对身高进行预测。

运行示例：

Are you a boy(M) or a girl(F)?F↙

Please input your father's height(cm):182↙

Please input your mother's height(cm):162↙

Do you like sports(Y/N)?N↙

Do you have a good habit of diet(Y/N)?Y↙

Your future height will be 167(cm)

参考答案：

#include  <stdio.h>

main()

{

    char sex;                    /\* 孩子的性别 \*/

    char sports;            /\* 是否喜欢体育运动 \*/

    char diet;          /\* 是否有良好的饮食习惯 \*/

    float myHeight;     /\* 孩子身高 \*/

    float faHeight;     /\* 父亲身高 \*/

    float moHeight;     /\* 母亲身高 \*/

    printf("Are you a boy(M) or a girl(F)?");

    scanf(" %c", &sex);   /\* 在%c前加一个空格，将存于缓冲区中的回车符读走 \*/

    printf("Please input your father's height(cm):");

    scanf("%f", &faHeight);

    printf("Please input your mother's height(cm):");

    scanf("%f", &moHeight);

    printf("Do you like sports(Y/N)?");

    scanf(" %c", &sports);/\* %c前加一空格，读走缓冲区中的回车符 \*/

    printf("Do you have a good habit of diet(Y/N)?");

    scanf(" %c", &diet); /\* %c前加一空格，读走缓冲区中的回车符 \*/

    if (sex == 'M' || sex == 'm')

        myHeight = (faHeight + moHeight) \* 0.54;

    else

        myHeight = (faHeight \* 0.923 + moHeight) / 2.0;

    if (sports == 'Y' || sports == 'y')

        myHeight = myHeight \* (1 + 0.02);

    if (diet == 'Y' || diet == 'y')

        myHeight = myHeight \* (1 + 0.015);

    printf("Your future height will be %.0f(cm)\n", myHeight);

}

一．从键盘任意输入一个整数m，若m不是素数，则对m进行质因数分解，并将m表示为质因 数从小到大顺序排列的乘积形式输出，否则输出"It is a prime number"。例如，用户输入90时，程序输出90 = 2 \* 3 \* 3 \* 5；用户输入17时，程序输出"It is a prime number"。

输入提示信息："Input m:"

输入格式："%d"

输出格式：

是素数时输出"It is a prime number\n"

否则输出用"%d = "，"%d \* "

运行示例1：

Input m:90↙

90 = 2 \* 3 \* 3 \* 5

运行示例2：

Input m:13↙

It is a prime number

一．

#include <stdio.h>

#include <math.h>

int IsPrime(int x);

void OutputPrimeFactor(int x);

int main()

{

int m;

printf("Input m:");

scanf("%d", &m);

if (IsPrime(m)) /\* 素数判定 \*/

{

printf("It is a prime number\n");

}

else

{

printf("%d = ", m);

OutputPrimeFactor(m); /\* 输出x的质因数连乘 \*/

}

return 0;

}

/\* 函数功能：判断x是否是素数，若函数返回0，则表示不是素数，若返回1，则代表是素数 \*/

int IsPrime(int x)

{

int i, flag = 1;

int squareRoot = (int)sqrt(x);

if (x <= 1) flag = 0; /\* 负数、0和1都不是素数 \*/

for (i = 2; i <= squareRoot && flag; i++)

{

if (x % i == 0) flag = 0; /\* 若能被整除，则不是素数 \*/

}

return flag;

}

/\* 函数功能：输出x的质因数连乘 \*/

void OutputPrimeFactor(int x)

{

int i;

for (i = 2; i < x; i++)

{

if (x % i == 0)

{

printf("%d \* ", i);

OutputPrimeFactor(x / i); /\* 递归调用该函数 \*/

return; /\* 不可以使用break \*/

}

}

printf("%d", x); /\* 输出最后一个因子（质因数，不能再分解）\*/

}

二、编程模拟洗牌和发牌过程。一副扑克有52张牌，分为4种花色（suit）：黑桃（Spades）、红桃（Hearts）、草花（Clubs）、方块（Diamonds）。每种花色又有13张牌面（face）：A，2，3，4，5，6，7，8，9，10，Jack，Queen，King。要求用结构体数组card表示52张牌，每张牌包括花色和牌面两个字符型数组类型的数据成员。

二、

#include <stdio.h>

#include <string.h>

#include <time.h>

#include <stdlib.h>

typedef struct card

{

char suit[10];

char face[10];

} CARD;

void Deal(CARD \*wCard);

void Shuffle(CARD \*wCard);

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[]);

int main()

{

char \*suit[] = {"Spades", "Hearts", "Clubs", "Diamonds"};

char \*face[] = {"A", "2", "3", "4", "5", "6", "7", "8", "9", "10",

"Jack", "Queen", "King"

};

CARD card[52];

srand (time(NULL));

FillCard(card, face, suit);

Shuffle(card);

Deal(card);

return 0;

}

/\* 函数功能：花色按黑桃、红桃、草花、方块的顺序，面值按A~K的顺序，排列52张牌 \*/

void FillCard(CARD wCard[], char \*wFace[], char \*wSuit[])

{

int i;

for (i = 0; i < 52; i++)

{

strcpy(wCard[i].suit, wSuit[i / 13]);

strcpy(wCard[i].face, wFace[i % 13]);

}

}

/\* 函数功能：将52张牌的顺序打乱以模拟洗牌过程 \*/

void Shuffle(CARD \*wCard)

{

int i, j;

CARD temp;

for (i = 0; i < 52; i++) /\*每次循环产生一个随机数，交换当前牌与随机数指示的牌\*/

{

j = rand() % 52; /\* 每次循环产生一个0~51的随机数 \*/

temp = wCard[i];

wCard[i] = wCard[j];

wCard[j] = temp;

}

}

/\* 函数功能：输出每张牌的花色和面值以模拟发牌过程 \*/

void Deal(CARD \*wCard)

{

int i;

for (i = 0; i < 52; i++)

{

printf("%9s%9s%c", wCard[i].suit, wCard[i].face, i % 2 == 0 ? '\n' : '\t');

}

printf("\n");

}

三、编程统计候选人的得票数。设有3个候选人zhang、li、wang，10个选民，选民每次输入一个得票的候选人的名字，若选民输错候选人姓名，则按废票处理。选民投票结束后程序自动显示各候选人的得票结果和废票信息。要求用结构体数组candidate表示3个候选人的姓名和得票结果。

三、

#include <stdio.h>

#include <string.h>

#define NUM\_ELECTORATE 10

#define NUM\_CANDIDATE 3

struct candidate

{

char name[20];

int count;

} candidate[3] = {"li", 0, "zhang", 0, "wang", 0};

int main()

{

int i, j, flag = 1, wrong = 0;

char name[20];

for (i = 1; i <= NUM\_ELECTORATE; i++)

{

printf("Input vote %d:", i);

scanf("%s", name);

flag = 1;

for (j = 0; j < NUM\_CANDIDATE; j++)

{

if (strcmp(name, candidate[j].name) == 0)

{

candidate[j].count++;

flag = 0;

}

}

if (flag)

{

wrong++; /\* 废票计数 \*/

flag = 0;

}

}

printf("Election results:\n");

for (i = 0; i < NUM\_CANDIDATE; i++)

{

printf("%8s:%d\n", candidate[i].name, candidate[i].count);

}

printf("Wrong election:%d\n", wrong);

return 0;

}

四、)\*约瑟夫问题。这是十七世纪的法国数学家加斯帕在《数目的游戏问题》中讲的一个故事：15个基督教徒和15个异教徒在海上遇险，必须将一半的人投入海中，其余的人才能幸免于难，于是想了个办法：30个人围成一个圆圈，从第一个人开始依次报数，每数到第9个人就将他扔入大海，如此循环进行直到仅余15个人为止。问怎样排法，才能使每次投入大海的都是异教徒。

\*\*输出格式要求："The original circle is (+:papandom, @:christian);\n" "%c "

程序运行示例如下：

The original circle is (+:papandom, @:christian);

@ @ @ @ + + + + + @ @ + @ @ @ + @ + + @ @ + + + @ + + @ @ +

四、

#include <stdio.h>

struct node

{

int nextp;

int no\_out;

}link[31];

main()

{

int i,j,k;

printf("The original circle is (+:papandom, @:christian);\n");

for(i=1;i<=30;i++)

{

link[i].nextp=i+1;

link[i].no\_out=1;

}

link[30].nextp=1;

j=30;

for(i=0;i<15;i++)

{

for(k=0;;)

if(k<9)

{

j=link[j].nextp;

k+=link[j].no\_out;

}

else break;

link[j].no\_out=0;

}

for(i=1;i<=30;i++)

printf("%c ",link[i].no\_out?'@':'+');

printf("\n");

}

五、完成对输入的字符串中C关键词的查找统计。

程序运行示例如下：

本程序将为您统计C语言的关键字的个数，请输入，输入end结束输入：

if do while while do break goto helloworld end

您的输入中C语言关键字出现的次数统计如下：

break : 1

do : 2

goto : 1

if : 1

while : 2

五、

#include <stdio.h>

#include <ctype.h>

#include <string.h>

#define MAXWORD 80

#define NKEYS (sizeof(keytab) / sizeof(struct key))

struct key

{

char \*word;

int count;

} keytab[] =

{

"auto", 0,

"break", 0,

"case", 0,

"char", 0,

"const", 0,

"continue", 0,

"default", 0,

"do", 0,

"double", 0,

"else", 0,

"enum", 0,

"extern", 0,

"float", 0,

"for", 0,

"goto", 0,

"if", 0,

"int", 0,

"long", 0,

"register", 0,

"return", 0,

"short", 0,

"singed", 0,

"sizeof", 0,

"static", 0,

"struct", 0,

"switch", 0,

"typedef", 0,

"union", 0,

"unsigned", 0,

"void", 0,

"volatile", 0,

"while", 0

};

void getword(char \*, int);

int binsearch(char \*, struct key \*, int);

int main(int argc, char \*argv[])

{

int n;

char word[MAXWORD];

printf("本程序将为您统计C语言的关键字的个数，请输入，输入end结束输入：\n");

do

{

getword(word, MAXWORD);

if (isalpha(word[0]))

{

if ((n = binsearch(word, keytab, NKEYS)) >= 0)

{

keytab[n].count++; //找到则对应次数+1

}

}

}

while (strcmp(word, "end") != 0);

printf("您的输入中C语言关键字出现的次数统计如下：\n");

for (n = 0; n < NKEYS; n++)

{

if (keytab[n].count > 0)

{

printf("%-10s: %6d\n", keytab[n].word, keytab[n].count);

}

}

return 0;

}

/\* 折半查找：在tab[0]到tab[n-1]中查找word \*/

int binsearch(char \*word, struct key tab[], int n)

{

int result;

int low, high, mid;

low = 0;

high = n - 1;

while (low <= high)

{

mid = (low + high) / 2;

if ((result = strcmp(word, tab[mid].word)) < 0)

{

high = mid - 1;

}

else if (result > 0)

{

low = mid + 1;

}

else

{

return mid;

}

}

return -1;

}

/\* getword：从输入中获取某个单词 \*/

void getword(char \*word, int lim)

{

int c;

void ungetch(int);

char \*w = word;

while (isspace(c = getchar()))

{

}

if (c != EOF)

{

\*w = c;

w++;

}

if (!isalpha(c))

{

\*w = '\0';

}

for ( ; --lim > 0; w++)

{

if (!isalnum(\*w = getchar()))

{

//读入的某个字符不是字母，则将它退还给输入缓冲区

ungetch(\*w);

break;

}

}

\*w = '\0';

}

六、模拟人工洗牌。编写一个模拟人工洗牌的程序，讲洗好的牌分别发给四个人。

\*\*输出格式要求："\nNo.%d:" " %c%2d,"

程序运行示例如下：

No.1: H 5, C 8, D 2, H11, H13, D10, C11, D 7, C 5, C 9, S 2, H10,

No.2: S 7, D 9, C 7, S10, D 3, D 4, D 6, H 1, S 1, H 7, D12, S 9,

No.3: D11, H 3, H 6, C 3, D13, S 4, C 1, S 5, D 1, C13, C 4, S 8,

No.4: S13, C 2, C 6, C12, H 8, S11, D 5, H12, S 3, S12, C10, H 9,

六、

#include <stdio.h>

#include <stdlib.h>

struct card

{

int pips;

char suit;

};

struct card deck[]={

{1,'C'},{2,'C'},{3,'C'},{4,'C'},{5,'C'},{6,'C'},{7,'C'},

{8,'C'},{9,'C'},{10,'C'},{11,'C'},{12,'C'},{13,'C'},

{1,'D'},{2,'D'},{3,'D'},{4,'D'},{5,'D'},{6,'D'},{7,'D'},

{8,'D'},{9,'D'},{10,'D'},{11,'D'},{12,'D'},{13,'D'},

{1,'H'},{2,'H'},{3,'H'},{4,'H'},{5,'H'},{6,'H'},{7,'H'},

{8,'H'},{9,'H'},{10,'H'},{11,'H'},{12,'H'},{13,'H'},

{1,'S'},{2,'S'},{3,'S'},{4,'S'},{5,'S'},{6,'S'},{7,'S'},

{8,'S'},{9,'S'},{10,'S'},{11,'S'},{12,'S'},{13,'S'},

};

shuffle(deck)

struct card deck[];

{

int i,j;

srand(time(NULL));

for(i=0;i<52;i++)

{

j=rand()%52;

swapcard(&deck[i],&deck[j]);

}

}

swapcard(p,q)

struct card \*p,\*q;

{

struct card temp;

temp=\*p;

\*p=\*q;

\*q=temp;

}

main()

{

int i;

shuffle(deck);

for(i=0;i<52;i++)

if(i%13==0)

printf("\nNo.%d:",i/13+1);

else

printf(" %c%2d,",deck[i].suit,deck[i].pips);

}

七、)13个人围成一圈，从第1个人开始顺序报号1、2、3，凡报到3的人退出圈子。找出最后留在圈子里的人原来的序号。要求用结构体编程实现。

程序的运行示例如下：

出圈成员及顺序： 3 6 9 12 2 7 11 4 10 5 1 8

最后的成员是： 13

七、

#include <stdio.h>

#define N 13

struct person

{

int number;

int nextp;

} link[N + 1];

int main(void)

{

int i, count, h;

/\* 建立队列 \*/

for (i = 1; i <= N; i++)

{

if (i == N)

{

link[i].nextp = 1;

}

else

{

link[i].nextp = i + 1;

}

link[i].number = i;

}

printf("\n");

count = 0;

h = N;

printf("出圈成员及顺序：");

while (count < N - 1)

{

i = 0;

while (i != 3)

{

h = link[h].nextp;

if (link[h].number)

{

i++;

}

}

printf("%3d", link[h].number);

link[h].number = 0;

count++;

}

printf("\n最后的成员是：");

for (i = 1; i <= N; i++)

{

if (link[i].number)

{

printf("%3d", link[i].number);

}

}

return 0;

}

八、求复数之积。利用结构变量求解两个复数之积。

za={3,4}, zb={5,6}

za={10,20}, zb={30,40}

\*\*输出格式要求："(%d+%di)\*(%d+%di)=" "(%d+%di)\n"

程序运行示例如下：

(3+4i)\*(5+6i)=(-9+38i)

(10+20i)\*(30+40i)=(-500+1000i)

八、#include "stdio.h"

struct complx

{

int real;

int im;

};

main()

{

static struct complx za={3,4};

static struct complx zb={5,6};

struct complx x,y,z;

struct complx cmult();

void cpr();

z=cmult(za,zb);

cpr(za,zb,z);

x.real = 10; x.im = 20;

y.real = 30; y.im = 40;

z=cmult(x,y);

cpr(x,y,z);

}

struct complx cmult (za,zb)

struct complx za,zb;

{

struct complx w;

w.real = za.real\*zb.real - za.im\*zb.im;

w.im = za.real\*zb.im+za.im\*zb.real;

return (w);

}

void cpr(za,zb,z)

struct complx za,zb,z;

{

printf("(%d+%di)\*(%d+%di)=",za.real,za.im,zb.real,zb.im);

printf("(%d+%di)\n",z.real,z.im);

}

九、对候选人得票的统计程度。设由3个候选人，每次输入一个得票的候选人的名字，要求最后输入各人得票结果。（总票数为n）请改正程序中的错误，使它能得出正确的结果。

九、#include <string.h>

struct person

{

char name[20];

int count;

};

struct person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

int i, j, n;

char leader\_name[20];

scanf("%d\n", &n);

for (i = 0; i < n; i++)

{

scanf("%s", leader\_name);

for (j = 0; j < 3; j++)

if (!strcmp(leader\_name, leader[j].name))

leader[j].count++;

}

printf("\n");

for (i = 0; i < 3; i++)

printf("%5s:%d\n", leader[i].name, leader[i].count);

}

十、定义一个结构体变量（包括年、月、日）。计算该日在本年中是第几天？注意闰年问题。

程序的运行示例如下：

请输入日期（年，月，日）

1990,2,14

2月14日是1990年的第45天。

十、#include <stdio.h>

struct

{

int year;

int month;

int day;

} date;

int main(void)

{

int days;

printf("请输入日期（年，月，日）\n");

scanf("%d,%d,%d", &date.year, &date.month, &date.day);

switch (date.month)

{

case 1:

days = date.day;

break;

case 2:

days = date.day + 31;

break;

case 3:

days = date.day + 59;

break;

case 4:

days = date.day + 90;

break;

case 5:

days = date.day + 120;

break;

case 6:

days = date.day + 151;

break;

case 7:

days = date.day + 181;

break;

case 8:

days = date.day + 212;

break;

case 9:

days = date.day + 243;

break;

case 10:

days = date.day + 273;

break;

case 11:

days = date.day + 304;

break;

case 12:

days = date.day + 334;

break;

}

if ((date.year % 4 == 0 && date.year % 100 != 0

|| date.year % 400 == 0) && date.month >= 3)

{

days += 1;

}

printf("\n%d月%d日是%d年的第%d天。", date.month, date.day, date.year, days);

return 0;

}

十一、以下程序有若干语法错误。请找出并改正之。

#include <stdio.h>

struct date

{

int y;m;d;

}

struct stu

{ char n[10];

struct date b;

int a;

}s={“Wang”,{1980,11,11},30};

main()

{

printf(“%c,%d,%d\n”,s.n,s.d,s.a);

}

十一、

#include <stdio.h>

struct date

{

int y,m,d;

};

struct stu

{ char n[10];

struct date b;

int a;

}s={"Wang",{1980,11,11},30};

main()

{

printf("%s,%d,%d\n",s.n,s.b.d,s.a);

}

十二、编程统计候选人的得票数。设有3个候选人，每次输入一个得票候选人的名字，不考虑弃权情况，要求最后输出各个候选人的得票结果（候选人姓名和参加投票人数由程序运行时输入）。

要求输入、输出在主函数中实现，统计候选人由函数实现。

按如下函数原型编写程序

void Vote(struct Person ld[], int n)

程序的运行示例如下：

输入3个候选人的基本信息：姓名

wang

song

zhao

输入参加投票的人数:5

输入5个参加投票人的投票情况

wang

wang

song

song

zhao

输出3个候选人的基本信息：姓名，票数

wang 2

song 2

zhao 1

十二、#include <stdio.h>

#define N 3

struct Person

{

char name[16];

int count;

};

void Input(struct Person ld[], int n); //输入n个候选人的基本信息

void Vote(struct Person ld[], int n); //对n个候选人进行投票

void Output(struct Person ld[], int n); //输出n个候选人的基本信息

int main(int argc, char \*argv[])

{

struct Person leader[N];

Input(leader, N);

Vote(leader, N);

Output(leader, N);

return 0;

}

void Input(struct Person ld[], int n)

{

int i;

printf("输入%d个候选人的基本信息：姓名\n", n);

for (i = 0; i < n; i++)

scanf("%s", ld[i].name);

}

void Vote(struct Person ld[], int n)

{

int m, i, j; //m表示参加投票的人数

char ename[16];

printf("输入参加投票的人数:");

scanf("%d", &m);

printf("输入%d个参加投票人的投票情况\n", m);

for (i = 0; i < m; i++)

{

ld[i].count = 0;

}

for (i = 0; i < m; i++)

{

scanf("%s", ename);

for (j = 0; j < n; j++)

if (strcmp(ld[j].name, ename) == 0)

ld[j].count++;

}

}

void Output(struct Person ld[], int n)

{

int i;

printf("输出%d个候选人的基本信息：姓名，票数\n", n);

for (i = 0; i < n; i++)

printf("%s %d\n", ld[i].name, ld[i].count);

}

十三、中国有句俗语叫“三天打鱼两天晒网”，某人从90年1月1日起开始“三天打鱼两天晒网”。问这个人在以后的某一天中是在“打渔”，还是在“晒网”.

\*\*输入格式要求："%d%d%d" 提示信息："Enter year/month/day:"

\*\*输出格式要求："He is fishing.\n" "He is sleeping.\n"

程序运行示例如下：

Enter year/month/day:1990 1 5

He is sleeping.

十三、#include <stdio.h>

struct date

{

int year,month,day;

};

main()

{

struct date today,term;

int yeardays=0,day;

printf("Enter year/month/day:");

scanf("%d%d%d",&today.year,&today.month,&today.day);

term.month=12;

term.day=31;

for(term.year=1990;term.year>today.year;term.year++)

yeardays+=days(term);

yeardays+=days(today);

day=yeardays%5;

if(day>0&&day<4)

printf("He is fishing.\n");

else

printf("He is sleeping.\n");

}

days(day)

struct date day;

{

static int day\_tab[2][13]={{0,31,28,31,30,31,30,31,31,30,31,30,31},

{0,31,29,31,30,31,30,31,31,30,31,30,31}};

int i,lp;

lp=day.year%4==0&&day.year%100!=0||day.year%400==0;

for(i=1;i<day.month;i++)

day.day+=day\_tab[lp][i];

return (day.day);

}

十四、程序改错。以下程序用于打印输出结构体数组的所有元素。

struct s

{

char name[10];

int age;

}

main()

{

struct s a[3] = {”John”,19,”Paul”,17,”Marry”,18};

int \*p;

for(p = a; p < 3; p++)

{

printf("%s,%d\n", p->name, p->age );

}

}

十四、struct s

{

char name[10];

int age;

};

main()

{

struct s a[3] = {"John",19,"Paul",17,"Marry",18};

struct s \*p;

for(p = a; p < a+3; p++)

{

printf("%s,%d\n", p->name, p->age );

}

}

十五、写一个函数days，实现主函数将年、月、日（结构体类型）传递给days函数，days函数计算该年该月该日是该年的第几天并传回主函数输出。

程序的运行示例如下：

请输入日期（年，月，日）

1990,2,14

2月14日是1990年的第45天。

十五、#include <stdio.h>

struct dt

{

int year;

int month;

int day;

} date;

int days(struct dt date)

{

int sum;

switch (date.month)

{

case 1:

sum = date.day;

break;

case 2:

sum = date.day + 31;

break;

case 3:

sum = date.day + 59;

break;

case 4:

sum = date.day + 90;

break;

case 5:

sum = date.day + 120;

break;

case 6:

sum = date.day + 151;

break;

case 7:

sum = date.day + 181;

break;

case 8:

sum = date.day + 212;

break;

case 9:

sum = date.day + 243;

break;

case 10:

sum = date.day + 273;

break;

case 11:

sum = date.day + 304;

break;

case 12:

sum = date.day + 334;

break;

}

if ((date.year % 4 == 0 && date.year % 100 != 0

|| date.year % 400 == 0) && date.month >= 3)

{

sum += 1;

}

return sum;

};

int main(void)

{

printf("请输入日期（年，月，日）\n");

scanf("%d,%d,%d", &date.year, &date.month, &date.day);

printf("\n%d月%d日是%d年的第%d天。", date.month, date.day, date.year, days(date));

return 0;

}

十六、用结构体实现输入4个学生的学号（无符号整数unsigned int）、姓名（字符串）和考试成绩（整数），然后按从高分到低分的顺序排列后输出。

程序的运行示例如下：

输入第1个学生的信息：学号、姓名、成绩

2011 zhangsan 89

输入第2个学生的信息：学号、姓名、成绩

2012 lisi 96

输入第3个学生的信息：学号、姓名、成绩

2013 wangwu 67

输入第4个学生的信息：学号、姓名、成绩

2014 liuliu 59

排序后学生的信息为:

2012 lisi 96

2011 zhangsan 89

2013 wangwu 67

2014 liuliu 59

十六、#include <stdio.h>

#define N 4

struct Student1

{

unsigned int sno;

char sname[16];

int score;

};

int main(int argc, char \*argv[])

{

struct Student1 stu[N];

int i, j;

struct Student1 temp;

for (i = 0; i < N; i++)

{

printf("输入第%d个学生的信息：学号、姓名、成绩\n", i + 1);

scanf("%d %s %d", &stu[i].sno, stu[i].sname, &stu[i].score);

}

for (i = 0; i < N - 1; i++)

{

for (j = 0; j < N - i - 1; j++)

{

if (stu[j].score < stu[j + 1].score)

{

temp = stu[j];

stu[j] = stu[j + 1];

stu[j + 1] = temp;

}

}

}

printf("排序后学生的信息为:\n");

for (i = 0; i < N; i++)

{

printf("%d %s %d\n", stu[i].sno, stu[i].sname, stu[i].score);

}

return 0;

}

十七、编程用链表方法实现堆栈数据结构。堆栈（Strack）是指这样一段内存，它可以理解为一个筒结构，先放进筒中的数据被后放进筒中的数据“压住”，只有后放进筒中的数据都取出后，先放进去的数据才能被取出，称为“后进先出”。堆栈的长度可随意增加。堆栈结构可用链表实现。设计一个链表结构需包含两个成员：一个存放数据，一个为指向下一个节点的指针。当每次有一个新数据要放入堆栈时，称为“压入堆栈”，这时动态建立一个链表的节点，并连接到链表的结尾；当每次从堆栈中取出一个数据时，称为“弹出堆栈”，这意味着从链表的最后一个节点中取出该节点的数据成员，同时删除该节点，释放该节点所占的内存。

十七、#include <stdio.h>

#include <stdlib.h>

typedef struct stack

{

int data;

struct stack \*next;

} STACK;

STACK \*head, \*pr;

int nodeNum = 0; /\* 堆栈节点数寄存器 \*/

STACK \*CreateNode(int num);

STACK \*PushStack(int num);

int PopStack(void);

int main()

{

int pushNum[5] = {111, 222, 333, 444, 555}, popNum[5], i;

for (i = 0; i < 5; i++)

{

PushStack(pushNum[i]);

printf("Push %dth Data:%d\n", i + 1, pushNum[i]);

}

for (i = 0; i < 5; i++)

{

popNum[i] = PopStack();

printf("Pop %dth Data:%d\n", 5 - i, popNum[i]);

}

return 0;

}

/\* 函数功能：生成一个新的节点，并为该节点赋初值，返回指向新的节点的指针 \*/

STACK \*CreateNode(int num)

{

STACK \*p;

p = (STACK \*)malloc(sizeof(STACK));

if (p == NULL)

{

printf("No enough memory!\n");

exit(0);

}

p->next = NULL; /\* 为新建的节点指针域赋空指针 \*/

p->data = num; /\* 为新建的节点数据区赋值 \*/

return p;

}

/\* 函数功能：将整型变量num的值压入堆栈，返回指向链表新节点的指针 \*/

STACK \*PushStack(int num)

{

if (nodeNum == 0) /\* 若为首节点，则保留该节点地址在head中\*/

{

head = CreateNode(num);

pr = head;

nodeNum++; /\* 堆栈节点数寄存器+1 \*/

}

else /\* 若不是首节点，则将新建节点连到链表的结尾处 \*/

{

pr->next = CreateNode(num);

pr = pr->next;

nodeNum++; /\* 堆栈节点数寄存器+1 \*/

}

return pr;

}

/\* 函数功能：将当前栈顶的数据弹出堆栈，返回从堆栈中弹出的数据 \*/

int PopStack(void)

{

STACK \*p = head;

int result;

for (;;)

{

if (p->next == NULL) /\* 查找最后一个节点 \*/

{

break;

}

else

{

pr = p; /\* 记录最后一个节点的前一个节点的地址 \*/

p = p->next;

nodeNum--; /\* 堆栈节点数寄存器-1 \*/

}

}

pr->next = NULL; /\* 将末节点的前一个节点置成末节点 \*/

result = p->data;

free(p);

return result;

}

十八、将一个链表按逆序排列，即将链头当链尾，链尾当链头。

程序的运行示例如下：

请输入链表（非数表示结束）

结点值：3

结点值：4

结点值：5

结点值：6

结点值：7

结点值：end

原来表：

3 4 5 6 7

反转表：

7 6 5 4 3

十八、#include <stdio.h>

struct line

{

int num;

struct line \*next;

} \*p1, \*p2;

struct line \*creat()

{

int temp;

struct line \*head = NULL;

printf("\n请输入链表（非数表示结束）\n结点值：");

while (scanf("%d", &temp))

{

p1 = (struct line \*)malloc(sizeof(struct line));

(head == NULL) ? (head = p1) : (p2->next = p1);

p1->num = temp;

printf("结点值：");

p2 = p1;

}

p2->next = NULL;

return head;

}

output(struct line \*outhead)

{

for (p1 = outhead; p1 != NULL; printf("%4d", p1->num), p1 = p1->next);

{

}

}

struct line \*turnback(struct line \*head)

{

struct line \*new, \*newhead = NULL;

do

{

p2 = NULL;

p1 = head;

while (p1->next != NULL)

{

p2 = p1;

p1 = p1->next;

}

if (newhead == NULL)

{

newhead = p1;

new = newhead->next = p2;

}

new = new->next = p2;

p2->next = NULL;

}

while (head->next != NULL);

return newhead;

}

int main(void)

{

struct line \*head;

head = creat();

printf("\n原来表：\n");

output(head);

head = turnback(head);

printf("\n\n反转表：\n");

output(head);

return 0;

}

十九、)按如下函数原型编程计算并输出n×n阶矩阵的转置矩阵。其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int \*a, int n);

void Swap(int \*x, int \*y);

void InputMatrix(int \*a, int n);

void PrintMatrix(int \*a, int n);

输入提示信息："Input n:"

输入格式："%d"

输出提示信息和格式："The transposed matrix is:\n"

十九、#include <stdio.h>

#define N 10

void Swap(int \*x, int \*y);

void Transpose(int \*a, int n);

void InputMatrix(int \*a, int n);

void PrintMatrix(int \*a, int n);

int main()

{

int s[N][N], n;

printf("Input n:");

scanf("%d", &n);

InputMatrix(\*s, n);

Transpose(\*s, n);

printf("The transposed matrix is:\n");

PrintMatrix(\*s, n);

return 0;

}

/\* 函数功能：交换两个整型数的值 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* 函数功能：计算n\*n矩阵的转置矩阵 \*/

void Transpose(int \*a, int n)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = i; j < n; j++)

{

Swap(&a[i \* n + j], &a[j \* n + i]);

}

}

}

/\* 函数功能：输入n\*n矩阵的值 \*/

void InputMatrix(int \*a, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", n, n);

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i \* n + j]);

}

}

}

/\* 函数功能：输出n\*n矩阵的值 \*/

void PrintMatrix(int \*a, int n)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

printf("%d\t", a[i \* n + j]);

}

printf("\n");

}

}

二十、按如下函数原形编程计算并输出求m\*m阶矩阵的转置矩阵。期中m的值由用户从键盘输入。已知m的值都不超过10。

求转置矩阵函数原形：void Transpose(int a[][M], int at[][M], int m);

\*\*\*\*输入提示信息和格式要求为：

"Input m"

"%d"

"Input %d\*%d matrix:\n"

"%d"

\*\*\*\*输出提示信息和格式要求为：

"The transposed matrix is:\n"

"%8d"

二十、#include <stdio.h>

#define M 10

void Transpose(int a[][M], int at[][M], int m);

int main()

{

int a[M][M],at[M][M];

int m;

int i,j;

printf("Input m:"); //1

scanf("%d",&m);

printf("Input %d\*%d matrix:\n",m,m); //1

for (i=0;i<m;i++)

{

for (j=0;j<m;j++)

{

scanf("%d",&a[i][j]);//1

}

}

Transpose(a,at,m); //2

for (i=0;i<m;i++)

{

for (j=0;j<m;j++)

{

printf("%8d",at[i][j]);//1

}

printf( "\n" );

}

return 0;

}

void Transpose(int a[][M], int at[][M], int m)

{

int i,j;

for (i=0;i<m;i++) //1

{

for (j=0;j<m;j++) //1

{

at[i][j]=a[j][i]; //1

}

}

}

二十一、给定某年某月某日，将其转换成这一年的第几天并输出。

\*\*输入格式要求："%d%d%d"

\*\*输出格式要求："%d"

程序运行示例如下：

2012 11 8

313

二十一、#include <stdio.h>

main()

{

static int day\_tab[2][13]={

{0,31,28,31,30,31,30,31,31,30,31,30,31},

{0,31,29,31,30,31,30,31,31,30,31,30,31}};

int y,m,d;

scanf("%d%d%d",&y,&m,&d);

printf("%d\n",day\_of\_year(day\_tab,y,m,d));

}

day\_of\_year(day\_tab,year,month,day)

int day\_tab[][13];

int year,month,day;

{

int i,j;

i=(year%4==0&&year%100!=0)||year%400==0;

for(j=1;j<month;j++)

day+=day\_tab[i][j];

return day;

}

二十二、口袋中有若干红、黄、蓝、白、黑5种颜色的球，每次从口袋中取出3个球，定义如下指针数组，编程打印出得到3种不同颜色的球的所有可能取法。

char \*bColor[] = {"RED", "YELLOW", "BLUE", "WHITE", "BLACK"};

输入提示信息和数据格式：无

输出提示信息和数据格式："%d:%s,%s,%s\n"

二十二、#include <stdio.h>

int main(void)

{

char \*bColor[] = {"RED", "YELLOW", "BLUE", "WHITE", "BLACK"};

int i, j, k, m = 0;

for (i = 0; i < 5; i++)

{

for (j = i + 1; j < 5; j++)

{

for (k = j + 1; k < 5; k++)

{

m++;

printf("%d:%s,%s,%s\n", m, bColor[i], bColor[j], bColor[k]);

}

}

}

return 0;

}

二十三、用动态数组编程输入任意m个班学生（每班n个学生）的某门课的成绩，计算最高分，并指出具有该最高分成绩的学生是第几个班的第几个学生。其中，m和n的值由用户从键盘任意输入（不限定m和n的上限值）。

输入提示信息：

"Input array size m,n:"

"Input %d\*%d array:\n"

输入格式："%d,%d"

输出提示信息和格式："maxScore = %d, class = %d, number = %d\n"

二十三、#include <stdio.h>

#include <stdlib.h>

void InputScore(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);

int main()

{

int \*pScore, m, n, maxScore, row, col;

printf("Input array size m,n:");

scanf("%d,%d", &m, &n);

pScore = (int \*)calloc(m \* n, sizeof (int)); /\* 申请动态内存 \*/

if (pScore == NULL)

{

printf("No enough memory!\n");

exit(0);

}

InputScore(pScore, m, n);

maxScore = FindMax(pScore, m, n, &row, &col);

printf("maxScore = %d, class = %d, number = %d\n", maxScore, row + 1, col + 1);

free(pScore); /\* 释放动态内存 \*/

return 0;

}

/\* 函数功能：输入m行n列二维数组的值 \*/

void InputScore(int \*p, int m, int n)

{

int i, j;

printf("Input %d\*%d array:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &p[i \* n + j]);

}

}

}

/\* 函数功能：计算任意m行n列二维数组中元素的最大值，并指出其所在行列下标值 \*/

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol)

{

int i, j, max = p[0];

\*pRow = 0;

\*pCol = 0;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

if (p[i \* n + j] > max)

{

max = p[i \* n + j];

\*pRow = i; /\*记录行下标\*/

\*pCol = j; /\*记录列下标\*/

}

}

}

return max;

}

二十四、用二维数组的列指针作为函数实参，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int \*a, int \*at, int m, int n);

void InputMatrix(int \*a, int m, int n);

void PrintMatrix(int \*at, int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输出提示信息和格式："The transposed matrix is:\n"

二十四、#include <stdio.h>

#define M 10

#define N 10

void Transpose(int (\*a)[N], int (\*at)[M], int m, int n);

void InputMatrix(int (\*a)[N], int m, int n);

void PrintMatrix(int (\*at)[M], int n, int m);

int main()

{

int s[M][N], st[N][M], m, n;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(s, m, n);

Transpose(s, st, m, n);

printf("The transposed matrix is:\n");

PrintMatrix(st, n, m);

return 0;

}

/\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/

void Transpose(int (\*a)[N], int (\*at)[M], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

\*(\*(at + j) + i) = \*(\*(a + i) + j);

}

}

}

/\* 函数功能：输入m\*n矩阵a的值 \*/

void InputMatrix(int (\*a)[N], int m, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", \*(a + i) + j);

}

}

}

/\* 函数功能：输出n\*m矩阵at的值 \*/

void PrintMatrix(int (\*at)[M], int n, int m)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

printf("%d\t", \*(\*(at + i) + j));

}

printf("\n");

}

}

二十五、乘式还原。有乘法算式如下：

× o o o o o o o o o o o o o o o o o o

18个O的位置上全部是素数（2、3、5或7），请还原此算式。

程序运行示例如下：

775

\* 33

-----

2325

2325

-----

25575

二十五、#include <stdio.h>

#define NUM 5

#define C\_NUM 4

int a[NUM+1];

int b[]={0,2,3,5,7};

main()

{

int i,not\_finish=1;

i=2;

a[1]=1;

while(not\_finish)

{

while(not\_finish&&i<=NUM)

if(a[i]>=C\_NUM)

if(i==1&&a[i]==C\_NUM)

not\_finish=0;

else a[i--]=0;

else a[i++]++;

if(not\_finish)

{

long int sum1,sum2,sum3,sum4;

sum1=b[a[1]]\*100+b[a[2]]\*10+b[a[3]];

sum2=sum1\*b[a[5]];

sum3=sum1\*b[a[4]];

if(sum2>=2222&&sum2<=7777&&f(sum2)&&sum3>=2222&&sum3<=7777&&f(sum3))

if((sum4=sum2+sum3\*10)>=22222&&sum4<=77777&&f(sum4))

{

printf(" %d\n",sum1);

printf("\* %d%d\n",b[a[4]],b[a[5]]);

printf("-----\n");

printf(" %d\n",sum2);

printf("%d\n",sum3);

printf("-----\n");

printf("%d\n",sum4);

}

i=NUM;

}

}

}

f(sum)

long sum;

{ int i,k,flag;

while(sum>0)

{

i=sum%10;

for(flag=0,k=1;!flag&&k<=C\_NUM;k++)

if(b[k]==i)

{

flag=1;

break;

}

if(!flag) return(0);

else sum=sum/10;

}

return (1);

}

1. 编写一个函数，输入n为偶数时，调用函数求1/2+1/4+...+1/n,当输入n为奇数时，调用函数 求1/1+1/3+...+1/n(要求利用函数指针编程)

程序运行示例1如下：

10

Even=2.141666

程序运行示例2如下：

9

Odd=1.787302

一

#include "stdio.h"

float peven(int n)

{

float s;

int i;

s = 1;

for (i = 2; i <= n; i += 2)

s += 1 / (float)i;

return(s);

}

float podd(int n)

{

float s;

int i;

s = 0;

for (i = 1; i <= n; i += 2)

s += 1 / (float)i;

return(s);

}

float dcall(float (\*fp)(), int n)

{

float s;

s = (\*fp)(n);

return(s);

}

main()

{

float sum;

int n;

while (1)

{

scanf("%d", &n);

if (n > 1)

break;

}

if (n % 2 == 0)

{

printf("Even=");

sum = dcall(peven, n);

}

else

{

printf("Odd=");

sum = dcall(podd, n);

}

printf("%f", sum);

}

2.按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

/\* 函数功能： 对给定的某年某月某日，计算它是这一年的第几天

函数参数： 整型变量year、month、day，分别代表年、月、日

函数返回值：这一年的第几天 \*/

int DayofYear(int year, int month, int day);

/\* 函数功能： 对给定的某一年的第几天，计算它是这一年的第几月第几日

函数入口参数：整型变量year，存储年

整型变量yearDay，存储这一年的第几天

函数出口参数：整型指针pMonth，指向存储这一年第几月的整型变量

整型指针pDay，指向存储第几日的整型变量

函数返回值： 无 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

输出如下菜单，用switch语句实现根据用户输入的选择执行相应的操作。

1. year/month/day -> yearDay

2. yearDay -> year/month/day

3. Exit

Please enter your choice:

输入提示信息：

"Please enter year, month, day:"

"Please enter year, yearDay:"

输入格式：

"%d,%d,%d"

"%d,%d"

输出提示信息和格式：

"yearDay = %d\n"

"month = %d, day = %d\n"

二

#include <stdio.h>

#include <stdlib.h>

int DayofYear(int year, int month, int day);

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay);

void Menu(void);

int dayTab[2][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}, {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

int main()

{

int year, month, day, yearDay;

char c;

Menu(); /\* 调用Menu函数显示一个固定式菜单 \*/

c = getchar(); /\* 输入选择 \*/

switch (c) /\* 判断选择的是何种操作 \*/

{

case '1':

printf("Please enter year, month, day:");

scanf("%d,%d,%d", &year, &month, &day);

yearDay = DayofYear(year, month, day);

printf("yearDay = %d\n", yearDay);

break;

case '2':

printf("Please enter year, yearDay:");

scanf("%d,%d", &year, &yearDay);

MonthDay(year, yearDay, &month, &day);

printf("month = %d,day = %d\n", month, day);

break;

case '3':

exit(0); /\* 退出程序的运行 \*/

default:

printf("Input error!");

}

return 0;

}

/\* 函数功能：对给定的某年某月某日，计算并返回它是这一年的第几天 \*/

int DayofYear(int year, int month, int day)

{

int i, leap;

/\* 若year为闰年，即leap值为1，则用第1行元素dayTab[1][i]计算；

否则leap值为0，用第0行dayTab[0][i]计算 \*/

leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

for (i = 1; i < month; i++)

{

day = day + dayTab[leap][i];

}

return day; /\* 返回计算出的day的值 \*/

}

/\* 函数功能：对给定的某一年的第几天，计算它是这一年的第几月第几日 \*/

void MonthDay(int year, int yearDay, int \*pMonth, int \*pDay)

{

int i, leap;

leap = ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

for (i = 1; yearDay > dayTab[leap][i]; i++)

{

yearDay = yearDay - dayTab[leap][i];

}

\*pMonth = i; /\* 将计算出的月份值赋值给pMonth所指向的变量 \*/

\*pDay = yearDay; /\* 将计算出的日号赋值给pDay所指向的变量 \*/

}

/\* 函数功能：显示菜单 \*/

void Menu(void)

{

printf("1. year/month/day -> yearDay\n");

printf("2. yearDay -> year/month/day\n");

printf("3. Exit\n");

printf("Please enter your choice:");

}

3. 按如下函数原型，采用梯形法编程实现，在积分区间[a,b]内计算函数y 1 =∫ 1 0 (1+x 2 )dx 和y 2 =∫ 3 0 x 1+x 2 dx 的定积分。其中，指向函数的指针变量f用于接收被积函数的入口地址。 Integral(float (\*f)(float), float a, float b);

\*\*输出格式要求："y1=%f\ny2=%f\n"

三

#include <stdio.h>

float Fun1(float x);

float Fun2(float x);

float Integral(float (\*f)(float), float a, float b);

int main()

{

float y1, y2;

y1 = Integral(Fun2, 0.0, 3.0);

y2 = Integral(Fun1, 0.0, 1.0);

printf("y1=%f\ny2=%f\n", y1, y2);

return 0;

}

/\* 函数功能：计算函数1+x\*x的函数值 \*/

float Fun1(float x)

{

return 1 + x \* x;

}

/\* 函数功能：计算函数x/(1+x\*x)的函数值 \*/

float Fun2(float x)

{

return x / (1 + x \* x);

}

/\* 函数功能：用梯形法计算函数的定积分 \*/

float Integral(float (\*f)(float), float a, float b)

{

float s, h;

int n = 100, i;

s = ((\*f)(a) + (\*f)(b)) / 2;

h = (b - a) / n;

for (i = 1; i < n; i++)

{

s += (\*f)(a + i \* h);

}

return s \* h;

}

4. 从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并返回它们所在数组中的位置。

\*\*输入格式要求："%d" 提示信息："Input n(n<=10):" "Input %d numbers:"

\*\*输出格式要求："%5d" "Exchange results:"

四

|  |
| --- |
| #include <stdio.h>  void ReadData(int a[], int n);  void PrintData(int a[], int n);  void MaxMinExchang(int a[], int n);  void Swap(int \*x, int \*y);  int main()  {  int a[10], n;  printf("Input n(n<=10):");  scanf("%d", &n);  printf("Input %d numbers:", n);  ReadData(a, n);  MaxMinExchang(a, n);  printf("Exchange results:");  PrintData(a, n);  return 0;  }  /\* 函数功能：输入数组a的n个元素值 \*/  void ReadData(int a[], int n)  {  int i;  for (i = 0; i < n; i++)  {  scanf("%d", &a[i]);  }  }  /\* 函数功能：输出数组a的n个元素值 \*/  void PrintData(int a[], int n)  {  int i;  for (i = 0; i < n; i++)  {  printf("%5d", a[i]);  }  printf("\n");  }  /\* 函数功能：将数组a中的最大数与最小数位置互换 \*/  void MaxMinExchang(int a[], int n)  {  int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;  int i;  for (i = 1; i < n; i++)  {  if (a[i] > maxValue)  {  maxValue = a[i];  maxPos = i;  }  if (a[i] < minValue)  {  minValue = a[i];  minPos = i;  }  }  Swap(&a[maxPos], &a[minPos]);  }  /\* 函数功能：两整数值互换 \*/  void Swap(int \*x, int \*y)  {  int temp;  temp = \*x;  \*x = \*y;  \*y = temp;  } |

#include <stdio.h>

main()

{

int i = 0;

char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[6];

while (str[i] != '\0')

{

putchar(str[i]);

i++;

}

ptr = str1;

scanf("%s", ptr);

puts(ptr);

ptr1 = str;

puts(ptr1);

ptr2 = "China";

puts(ptr2);

strcpy(str2, "China");

printf("%s", str2);

}

5. 有n个整数，使其前面各数顺序向后移m个位置，最后m个数变成最前面的m个数。

\*\*输入格式要求："%d" 提示信息："the total numbers is:" "back m:"

\*\*输出格式要求："%d," "%d"

程序运行示例1如下：

the total numbers is:5

back m:2

1 2 3 4 5

4,5,1,2,3

程序运行示例2如下：

the total numbers is:6

back m:3

1 2 3 4 5 6

4,5,6,1,2,3

6. 有n个人围成一圈，顺序排号。从第一个人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

\*\*输入格式要求："%d" 提示信息："please input the total of numbers:"

\*\*输出格式要求："%d is left\n"

程序运行示例如下：

please input the total of numbers:30

2 is left

六．

#include <stdio.h>

#define nmax 50

int main()

{

int i, k, m, n, num[nmax], \*p;

printf("please input the total of numbers:");

scanf("%d", &n);

p = num;

for (i = 0; i < n; i++ )

\*(p + i) = i + 1;

i = 0;

k = 0;

m = 0;

while (m < n)

{

if (\*(p + i) != 0)

k++;

if (k % 3 == 0)

{

\*(p + i) = 0;

m++;

}

i++;

if (i == n)

i = 0;

}

while (\*p == 0)

p++;

printf("%d is left\n", \*p);

}

7. 按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char dstStr[], char srcStr[]);

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

七．

#include <stdio.h>

#include <string.h>

#define N 80

void MyStrcat(char dstStr[], char srcStr[]);

int main()

{

char s[N], t[N];

printf("Input a string:");

gets(s);

printf("Input another string:");

gets(t);

MyStrcat(s, t);

printf("Concatenate results:%s\n", s);

return 0;

}

/\* 函数功能：将字符串srcStr连接到字符串dstStr之后 \*/

void MyStrcat(char dstStr[], char srcStr[])

{

unsigned int i, j;

i = strlen(dstStr); /\* 将下标移动到字符串dstStr的末尾 \*/

for (j = 0; j <= strlen(srcStr); j++, i++)

{

dstStr[i] = srcStr[j];

}

}

8. 编写程序实现将字符串str2拷贝到字符串str1。

\*\*输入格式要求："%s" 提示信息："Enter string 2:"

\*\*输出格式要求："Output string 1:%s\n"

程序运行的输入输出样例：

输入为

Enter string 2:good morning

输出为

Output string 1:good morning

八．#include <stdio.h>

main()

{

char str1[100],str2[100];

int i;

printf("Enter string 2:");

gets(str2);

for(i=0;(str1[i]=str2[i])!='\0';i++)

;

printf("Output string 1:%s\n",str1);

}

9. 按如下函数原型编写程序，用字符数组作函数参数编程实现字符串逆序存放功能。

void Inverse(char str[]);

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

九．

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

int main()

{

char a[N];

printf("Input a string:");

gets(a);

Inverse(a);

printf("Inversed results:%s\n", a);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char str[])

{

int len, i, j;

char temp;

len = strlen(str);

for (i = 0, j = len - 1; i < j; i++, j--)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

}

}

10. 用字符指针作函数参数编程实现字符串逆序存放功能。

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

十，

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char \*pStr);

int main()

{

char a[N];

printf("Input a string:");

gets(a);

Inverse(a);

printf("Inversed results:%s\n", a);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char \*pStr)

{

int len;

char temp;

char \*pStart; /\* 指针变量pStart指向字符串的第一个字符 \*/

char \*pEnd; /\* 指针变量pEnd指向字符串的最后一个字符 \*/

len = strlen(pStr); /\* 求出字符串长度 \*/

for (pStart = pStr, pEnd = pStr + len - 1; pStart < pEnd; pStart++, pEnd--)

{

temp = \*pStart;

\*pStart = \*pEnd;

\*pEnd = temp;

}

}

11. 请编程实现按奥运会参赛国国名在字典中的顺序对其入场次序进行排序。假设参赛国不超过150个。要求通过二维字符数组传参调用函数实现排序,输入和输出在主函数中实现。

\*\*输入格式要求："%d" 提示信息："How many countries?" "Input their names:\n"

\*\*输出格式要求："Sorted results:\n"

程序的运行示例如下：

How many countries? 5

Input their names:

America

England

Australia

Sweden

Finland

Sorted results:

America

Australia

England

Finland

Sweden

十一,

#include <stdio.h>

#include <string.h>

#define MAX\_LEN 10 /\* 字符串最大长度 \*/

#define N 150 /\* 字符串个数 \*/

void SortString(char str[][MAX\_LEN], int n);

int main()

{

int i, n;

char name[N][MAX\_LEN]; /\* 定义二维字符数组 \*/

printf("How many countries?");

scanf("%d", &n);

getchar(); /\* 读走输入缓冲区中的回车符 \*/

printf("Input their names:\n");

for (i = 0; i < n; i++)

{

gets(name[i]); /\* 输入n个字符串 \*/

}

SortString(name, n); /\* 字符串按字典顺序排序 \*/

printf("Sorted results:\n");

for (i = 0; i < n; i++)

{

puts(name[i]); /\* 输出排序后的n个字符串 \*/

}

return 0;

}

/\* 函数功能：交换法实现字符串按字典顺序排序 \*/

void SortString(char str[][MAX\_LEN], int n)

{

int i, j;

char temp[MAX\_LEN];

for (i = 0; i < n - 1; i++)

{

for (j = i + 1; j < n; j++)

{

if (strcmp(str[j], str[i]) < 0)

{

strcpy(temp, str[i]);

strcpy(str[i], str[j]);

strcpy(str[j], temp);

}

}

}

}

12.编写函数，通过指针将一个字符串反向。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："a=%s\n"

程序运行示例如下：

Enter string:love

a=evol

十二

#include <stdio.h>

revstr(char \*s)

{

char \*p=s,c;

while(\*p) p++;

p--;

while(s<p)

{

c=\*s;

\*s++=\*p;

\*p--=c;

}

}

main()

{

char a[50];

printf("Enter string:");

scanf("%s",a);

revstr(a);

printf("a=%s\n",a);

}

13. 利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

十三

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

14. 请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);

\*\*输入格式要求：无

\*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身

程序运行的输入输出样例1：

输入：

abc

输出：

No!

程序运行的输入输出样例2：

输入：

12321

输出：

12321

十四

#include <stdio.h>

#include <string.h>

int Judger( char a[ ]);

int main()

{

char a[80];

int flag;

gets(a);

flag = Judger(a);

if (flag == 1)

{

puts(a);

}

else

{

printf("No!");

}

return 0;

}

int Judger( char a[ ])

{

char \*pStar, \*pEnd;

int i, j, flag = 1;

j = strlen(a);

pStar = &a[0];

pEnd = &a[j - 1];

for (i = 0; i < (j / 2); i++)

{

if (\*pStar != \*pEnd)

{

flag = 0;

}

pStar++;

pEnd--;

}

return flag;

}

15. 用字符指针作函数参数编程实现在字符串每个字符间插入一个空格的功能。

\*\*输入提示信息："Input a string:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Insert results:%s\n"

程序运行示例1如下：

Input a string:Howareyou

Insert results:H o w a r e y o u

十五

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#define N 100

void Insert(char \*s);

int main()

{

char str[N];

printf("Input a string:");

gets(str);

Insert(str);

printf("Insert results:%s\n", str);

return 0;

}

void Insert(char \*s)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; s++, t++)

{

\*s = \*t;

s++;

\*s = ' ';

}

\*s = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

16. 用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*输入格式要求："%s"

输入提示信息：

"Input a string:"

"Input a character:"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

十六

#include <stdio.h>

#include <string.h>

#define N 100

void Squeeze(char \*s, char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char \*s, char c)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; t++)

{

if (\*t != c)

{

\*s = \*t;

s++;

}

}

\*s = '\0'; /\* 在字符串t2的末尾添加字符串结束标志 \*/

}

17. 编写函数，通过指针连接两个字符串。

\*\*输入格式要求："%s" 提示信息："Enter string 1:" "Enter string 2:"

\*\*输出格式要求："a+b=%s\n"

程序运行示例如下：

Enter string 1:happy

Enter string 2:hollween

a+b=happyhollween

十七

#include <stdio.h>

char \*strcat(char \*str1,char \*str2)

{

char \*p=str1;

while(\*p!='\0') p++;

while(\*p++=\*str2++);

return str1;

}

main()

{

char a[50],b[30];

printf("Enter string 1:");

scanf("%s",a);

printf("Enter string 2:");

scanf("%s",b);

printf("a+b=%s\n",strcat(a,b));

}

18. 在一个字符串中找出最大的字符并放在第一个位置上，并将该字符前的原字符往后顺序移动，如：cbyab变成ycbab。

\*\*输入格式要求：用gets读入

\*\*输出格式要求：用puts输出

程序运行的输入输出样例：

输入为：

cbyab

输出为：

Ycbab

十八

#include <stdio.h>

main()

{

char str[80], \*p = str, \*q, max;

gets(p);

max = \*(p++);

while (\*p != 0)

{

if (max < \*p)

{

max = \*p;

q = p;

}

p++;

}

p = q;

while (p > str)

{

\*p = \*(p - 1);

p--;

}

\*p = max;

puts(p);

}

19. 有n个人围成一圈，顺序排号。从第一人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

程序的运行示例如下：

请输入人数：30

30个人中最后留下的是第29号。

十九

/\* n个人循环报数逢3退出求最后一人是第几号 \*/

#include <stdio.h>

#define NMAX 50 /\* 定义最多人数为50人 \*/

int main(void)

{

int i, k, m, n, num[NMAX], \*p;

printf("请输入人数：");

scanf("%d", &n);

p = num;

for (i = 0; i < n; i++)

{

\*(p + i) = i + 1; /\* 以1至n为序给每个人编号 \*/

}

i = 0;

k = 0; /\* k为按1,2,3报数时的计数变量 \*/

m = 0; /\* m为退出人数 \*/

while (m < n - 1)

{

if (\*(p + i) != 0)

{

k++;

}

if (k == 3)

{

\*(p + i) = 0; /\* 对退出人员的编号置为0 \*/

k = 0;

m++;

}

i++;

if (i == n)

{

i = 0; /\* 报数到尾后i恢复为0 \*/

}

}

while (\*p == 0)

{

p++;

}

printf("%d个人中最后留下的是第%d号。", n, \*p);

return 0;

}

二十

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

int transform(char ch[])

{

int n=0,i=0;

while (ch[i] != '\0')

{

n = n \* 2 + ch[i] - '0';

i++;

}

return n;

}

void main()

{

char s[10];

int i;

int n;

printf("输入一个二进制字符串（10位）:");

gets(s);

if (strlen(s) > 10)

{

printf("错误：字符串太长\n");

exit(0);

}

for (i=0;s[i]!='\0';i++)

{

if (s[i] != '0' && s[i] != '1')

{

printf("错误:%c 不是一个二进制字符\n", s[i]);

exit(0);

}

}

printf("输入的二进制字符串为");

puts(s);

n = transform(s);

printf("\n%s 转换成十进制整数为：%d\n", s, n);

}

21. 从键盘任意输入一个字符串，计算其实际字符个数并打印输出，即不使用字符串处理函数strlen(),编程实现strlen()的功能。

\*\*输入格式要求："%s" 提示信息："Please enter a string:"

\*\*输出格式要求："The length of the string is: %u\n"

程序的运行示例如下：

Please enter a string:Hello China

The length of the string is: 11

二十一

#include <stdio.h>

unsigned int MyStrlen(char str[]);

/\* 函数功能：用字符型数组作函数参数，计算字符串的长度 \*/

unsigned int MyStrlen(char str[])

{

int i ;

unsigned int len = 0; /\* 计数器置0 \*/

for (i = 0; str[i] != '\0'; i++)

{

len++; /\* 利用循环统计不包括’\0’在内的字符个数 \*/

}

return (len); /\* 返回实际字符个数 \*/

}

int main()

{

char a[80];

unsigned int len;

printf("Please enter a string:");

gets(a);

len = MyStrlen(a); /\* 计算字符串实际字符个数 \*/

printf("The length of the string is: %u\n", len);

return 0;

}

22. 某班期末考试科目为数学（MT）、英语（EN）和物理（PH），有最多不超过40人参加考试。请编程计算：（1）每个学生的总分和平均分；（2）每门课程的总分和平均分。

\*\*输入格式要求："%d" "%ld"(学号) 提示信息："Input the total number of the students(n<40):" "Input student’s ID and score as: MT EN PH:\n"

\*\*输出格式要求："Counting Result:\n" "Student’s ID\t MT \t EN \t PH \t SUM \t AVER\n" "%12ld\t"(打印学号) "%4d\t"（打印每门课成绩）"%4d\t%5.1f\n"（打印总分和平均分） "SumofCourse \t" "%4d\t"(打印每门课的总分) "\nAverofCourse\t" "%4.1f\t"(每门课的平均分)

程序的运行示例如下：

Input the total number of the students(n<=40):4

Input student’s ID and score as: MT EN PH:

070310122 97 87 92

070310123 92 91 90

070310124 90 81 82

070310125 73 65 80

Counting Result:

Student’s ID MT EN PH SUM AVER

070310122 97 87 92 276 92.0

070310123 92 91 90 273 91.0

070310124 90 81 82 253 84.3

070310125 73 65 80 218 72.7

SumofCourse 352 324 344

AverofCourse 88.0 81.0 86.0

二十二

#include <stdio.h>

#define STUD\_N 40 /\* 最多学生人数 \*/

#define COURSE\_N 3 /\* 考试科目数 \*/

int ReadScore(int score[][COURSE\_N], long num[]);

void AverforStud(int score[][COURSE\_N], int sum[], float aver[], int n);

void AverforCourse(int score[][COURSE\_N], int sum[], float aver[], int n);

void Print(int score[][COURSE\_N], long num[], int sumS[], float averS[],

int sumC[], float averC[], int n);

int main()

{

int score[STUD\_N][COURSE\_N], sumS[STUD\_N], sumC[STUD\_N], n;

long num[STUD\_N];

float averS[STUD\_N], averC[STUD\_N];

n = ReadScore(score, num); /\* 读入学生成绩 \*/

AverforStud(score, sumS, averS, n); /\* 计算每个学生的总分平均分 \*/

AverforCourse(score, sumC, averC, n); /\* 计算每门课程的总分平均分 \*/

Print(score, num, sumS, averS, sumC, averC, n); /\* 输出学生成绩 \*/

return 0;

}

/\* 函数功能：输入学生的学号及其三门课的成绩，当输入负值时，结束输入，返回学生人数 \*/

int ReadScore(int score[][COURSE\_N], long num[])

{

int i, j, n;

printf("Input the total number of the students(n<40):");

scanf("%d", &n); /\* 输入参加考试的学生人数 \*/

printf("Input student’s ID and score as: MT EN PH:\n");

for (i = 0; i < n; i++) /\* 对所有学生进行循环 \*/

{

scanf("%ld", &num[i]); /\* 以长整型格式输入每个学生的学号 \*/

for (j = 0; j < COURSE\_N; j++) /\* 对所有课程进行循环 \*/

{

scanf("%d", &score[i][j]); /\* 输入每个学生的各门课成绩 \*/

}

}

return i; /\* 返回学生人数 \*/

}

/\* 函数功能： 计算每个学生的总分和平均分 \*/

void AverforStud(int score[][COURSE\_N], int sum[], float aver[], int n)

{

int i, j;

for (i = 0; i < n; i++)

{

sum[i] = 0;

for (j = 0; j < COURSE\_N; j++) /\* 对所有课程进行循环 \*/

{

sum[i] = sum[i] + score[i][j]; /\* 计算第i个学生的总分 \*/

}

aver[i] = (float)sum[i] / COURSE\_N;/\* 计算第i个学生的平均分 \*/

}

}

/\* 函数功能：计算每门课程的总分和平均分 \*/

void AverforCourse(int score[][COURSE\_N], int sum[], float aver[], int n)

{

int i, j;

for (j = 0; j < COURSE\_N; j++)

{

sum[j] = 0;

for (i = 0; i < n; i++) /\* 对所有学生进行循环 \*/

{

sum[j] = sum[j] + score[i][j]; /\* 计算第j门课程的总分 \*/

}

aver[j] = (float)sum[j] / n; /\* 计算第j门课程的平均分 \*/

}

}

/\*函数功能：打印每个学生的学号、各门课成绩、总分和平均分，以及每门课的总分和平均分\*/

void Print(int score[][COURSE\_N], long num[], int sumS[], float averS[],

int sumC[], float averC[], int n)

{

int i, j;

printf("Counting Result:\n");

printf("Student’s ID\t MT \t EN \t PH \t SUM \t AVER\n");

for (i = 0; i < n; i++)

{

printf("%12ld\t", num[i]); /\* 以长整型格式打印学生的学号 \*/

for (j = 0; j < COURSE\_N; j++)

{

printf("%4d\t", score[i][j]); /\* 打印学生的每门课成绩 \*/

}

printf("%4d\t%5.1f\n", sumS[i], averS[i]);/\*打印学生的总分平均分\*/

}

printf("SumofCourse \t");

for (j = 0; j < COURSE\_N; j++) /\* 打印每门课的总分 \*/

{

printf("%4d\t", sumC[j]);

}

printf("\nAverofCourse\t");

for (j = 0; j < COURSE\_N; j++) /\* 打印每门课的平均分 \*/

{

printf("%4.1f\t", averC[j]);

}

printf("\n");

}

23. 按如下函数原型编写程序，用函数编程实现两个数组中对应元素值的交换。

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void Swap(int \*x, int \*y);

输入提示信息：

"Input array size(n<=10):"

"Input array a:"

"Input array b:"

输入格式："%d"

输出提示信息：

"Output array a:"

"Output array b:"

输出格式："%5d"

二十三

#include <stdio.h>

#define N 10

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void Swap(int \*x, int \*y);

int main()

{

int a[N], b[N], i, n;

printf("Input array size(n<=10):");

scanf("%d", &n);

printf("Input array a:");

ReadData(a, n);

printf("Input array b:");

ReadData(b, n);

for (i = 0; i < n; i++)

{

Swap(&a[i], &b[i]);

}

printf("Output array a:");

PrintData(a, n);

printf("Output array b:");

PrintData(b, n);

return 0;

}

/\* 函数功能：输入数组a的n个元素值 \*/

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]);

}

}

/\* 函数功能：输出数组a的n个元素值 \*/

void PrintData(int a[], int n)

{

int i;

for (i = 0; i < n; i++)

{

printf("%5d", a[i]);

}

printf("\n");

}

/\* 函数功能：两整数值互换 \*/

void Swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

24．身高预测。每个做父母的都关心自己孩子成人后的身高，据有关生理卫生知识与数理统计分析表明，影响小孩成人后的身高的因素包括遗传、饮食习惯与体育锻炼等。小孩成人后的身高与其父母的身高和自身的性别密切相关。

设faHeight为其父身高，moHeight为其母身高，身高预测公式为

男性成人时身高 = (faHeight + moHeight) × 0.54 cm

女性成人时身高 = (faHeight × 0.923 + moHeight) / 2 cm

此外，如果喜爱体育锻炼，那么可增加身高2%；如果有良好的卫生饮食习惯，那么可增加身高1.5%。

请编程从键盘输入用户的性别（用字符型变量sex存储，输入字符F表示女性，输入字符M表示男性）、父母身高（用实型变量存储，faHeight为其父身高，moHeight为其母身高）、是否喜爱体育锻炼（用字符型变量sports存储，输入字符Y表示喜爱，输入字符N表示不喜爱）、是否有良好的饮食习惯等条件（用字符型变量diet存储，输入字符Y表示良好，输入字符N表示不好），利用给定公式和身高预测方法对身高进行预测。

运行示例：

二十四、#include <stdio.h>

main()

{

char sex; /\* 孩子的性别 \*/

char sports; /\* 是否喜欢体育运动 \*/

char diet; /\* 是否有良好的饮食习惯 \*/

float myHeight; /\* 孩子身高 \*/

float faHeight; /\* 父亲身高 \*/

float moHeight; /\* 母亲身高 \*/

printf("Are you a boy(M) or a girl(F)?");

scanf(" %c", &sex); /\* 在%c前加一个空格，将存于缓冲区中的回车符读走 \*/

printf("Please input your father's height(cm):");

scanf("%f", &faHeight);

printf("Please input your mother's height(cm):");

scanf("%f", &moHeight);

printf("Do you like sports(Y/N)?");

scanf(" %c", &sports);/\* %c前加一空格，读走缓冲区中的回车符 \*/

printf("Do you have a good habit of diet(Y/N)?");

scanf(" %c", &diet); /\* %c前加一空格，读走缓冲区中的回车符 \*/

if (sex == 'M' || sex == 'm')

myHeight = (faHeight + moHeight) \* 0.54;

else

myHeight = (faHeight \* 0.923 + moHeight) / 2.0;

if (sports == 'Y' || sports == 'y')

myHeight = myHeight \* (1 + 0.02);

if (diet == 'Y' || diet == 'y')

myHeight = myHeight \* (1 + 0.015);

printf("Your future height will be %.0f(cm)\n", myHeight);

}

Are you a boy(M) or a girl(F)?F↙

Please input your father's height(cm):182↙

Please input your mother's height(cm):162↙

Do you like sports(Y/N)?N↙

Do you have a good habit of diet(Y/N)?Y↙

Your future height will be 167(cm)

一．

有一电文，已按下面规律译成密码。

A->Z a->z

B->Y b->y

C->X c->x

… …

即第一个字母变成第26个字母，第i个字母变成第（26-i+1）个字母，非字母字符不变。要求编程将密码译回原文，并打印出密码和原文。

程序的运行示例如下：

请输入字符：hello,world!

密码是：hello,world!

原文是：svool,dliow!

参考答案：

#include <stdio.h>

int main()

{

    int i, n;

    char ch[80], tran[80];

    printf("请输入字符：");

    gets(ch);

    printf("\n密码是：%s", ch);

    i = 0;

    while (ch[i] != '\0')

    {

        if ((ch[i] >= 'A') && (ch[i] <= 'Z'))

        {

            tran[i] = 26 + 64 - ch[i] + 1 + 64;

        }

        else if ((ch[i] >= 'a') && (ch[i] <= 'z'))

        {

            tran[i] = 26 + 96 - ch[i] + 1 + 96;

        }

        else

        {

            tran[i] = ch[i];

        }

        i++;

    }

    n = i;

    printf("\n原文是：");

    for (i = 0; i < n; i++)

    {

        putchar(tran[i]);

    }

    return 0;

}

二．

编程实现两个数组中对应该元素值的交换（数组的长度定义成5）。

输出函数："%5d"

输入举例:注意输入的顺序

6 7 8 9 10 （输入第一个数组元素）

1 2 3 4 5 （输入第二个数组元素）

输出举例：

1 2 3 4 5 （输入第一个数组元素）

6 7 8 9 10 （输入第二个数组元素）

二．

#include <stdio.h>

int main()

{

    int a[5], b[5];

    int i, t;

    for (i = 0; i < 5; i++)

        scanf("%d", &a[i]);

    for (i = 0; i < 5; i++)

        scanf("%d", &b[i]);

    for (i = 0; i < 5; i++)

    {

        t = a[i];

        a[i] = b[i];

        b[i] = t;

    }

    for (i = 0; i < 5; i++)

        printf("%5d", a[i]);

    printf("\n");

    for (i = 0; i < 5; i++)

        printf("%5d", b[i]);

    printf("\n");

}

三．

从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str)==0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

三．

#include "stdio.h"

#include "string.h"

main()

{

    int i=0, findFlag=1;

    char x[13];

    char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

    printf("Input string:\n");

    gets(x);

    while( i<6 && findFlag)

    {

        if(strcmp(x,str[i])==0)

        {

            findFlag = 0;

        }

        i++;

    }

    if ( !findFlag )

    {

        printf("%s\n",x);

    }

    else

    {

        printf("Not find!\n");

    }

}

四．

魔术师的猜牌术。魔术师利用一副牌中的十三张黑桃，预先将它们排好后迭在一起，牌面朝下。对观众说：我不看牌，只数数就可以猜到每张牌是什么，我大声数数，你们听，不信？你们看看。魔术师将最上面的那张牌数为1，把它翻过来正好是黑桃A，将黑桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌下面，将第二张牌翻过来，正好是黑桃2，也将它放在桌子上。第三次数1、2、3，将前两张牌依次放在这迭牌下面，将第三张牌翻过来，正好是黑桃3。这样依次进行将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

\*\*输入格式要求：提示信息："The original order of cards is:"

\*\*输出格式要求："%d "

程序运行示例如下：

The original order of cards is:1 8 2 5 10 3 12 11 9 4 7 6 13

四．

#include <stdio.h>

int a[14];

main()

{int i,n,j=1;

 printf("The original order of cards is:");

 for(i=1;i<=13;i++)

 {n=1;

  do

  {if(j>13)

     j=1;

   if(a[j])

     j++;

   else

   {if(n==i)

       a[j]=i;

       j++;

       n++;

   }

  }while(n<=i);

 }

 for(i=1;i<=13;i++)

     printf("%d ",a[i]);

 printf("\n");

 }

五．

设数组a的定义如下：int a[20];

已存入数组中的数据值已经按由大到小的顺序存放，数组元素个数n和数组元素值通过键盘输入，现再从键盘输入一个数据x，把它插入到数组中，要求插入新数据以后，数组数据仍然保持有序。请编写一个程序实现上述功能。

\*\*输入格式要求："%d", 提示信息："输入数组中已有元素个数:" "输入数组中的元素:\n" "输入要插入的数据：" "输出插入数据后的所有数组元素:\n"

\*\*输出格式要求："%d "

程序运行示例如下：

输入数组中已有元素个数:5

输入数组中的元素:

8 7 5 3 1

输入要插入的数据：6

输出插入数据后的所有数组元素:

8 7 6 5 3 1

五．

#include <stdio.h>

int main()

{

    int a[20];

    int i, j, n,x;

    printf("输入数组中已有元素个数:");

    scanf("%d", &n);

    printf("输入数组中的元素:\n");

    for (i=0;i<n;i++)

        scanf("%d",&a[i]);

    printf("输入要插入的数据：");

    scanf("%d",&x);

    for (i = 0; i < n; i++)

        if (a[i] < x)

        {

            for (j = n; j > i; j--)

                a[j] = a[j - 1];

            a[i] = x;

            break;

        }

    if (i == n)

        a[i] = x;

    printf("输出插入数据后的所有数组元素:\n");

    for (i = 0; i < n+1; i++)

        printf("%d ", a[i]);

    printf("\n");

}

六．

编写一个程序，将十进制整数n转换成二进制数。

\*\*输入格式要求："%d" 提示信息："n="

\*\*输出格式要求："%d" "the number of binary is "

程序运行示例如下：

n=37

the number of binary is 0000000000100101

六．

#include <stdio.h>

void trans(int n, int b[]);

int main()

{

    int b[16], n, i;

    for (i = 0; i <= 15; i++)

    {

        b[i] = 0;

    }

    printf("n=");

    scanf("%d", &n);

    trans(n, b);

    printf("the number of binary is ");

    for (i = 15; i >= 0; i--)

    {

        printf("%d", b[i]);

    }

    printf("\n");

    return 0;

}

void trans(int n, int b[])

{

    int i = 0;

    while (n != 0)

    {

        b[i] = n % 2;

        i++;

        n /= 2;

    }

}

七．

创建并输出一个一维数组（含20个元素），数组元素的值分别是下标的3倍多2。

\*\*输出格式要求：" %5d"

程序运行示例如下：

2 5 8 11 14 17 20 23 26 29 32 35 38 41 44 47 50 53 56 59

七．

#include <stdio.h>

main()

{

    int a[20], k;

    for (k = 0; k < 20; k++)

    {

        a[k] = 3 \* k + 2;

        printf(" %5d", a[k]);

    }

}

八．

以下程序的功能是计算十个数据的平均值。找出其中的错误并改正之。

|  |
| --- |
| #include <stdio.h>  void main(void)  {      int i, sum;      float  avg;      int    sc[10], \*p = sc;        for (i=0, i<10, i++)      {              scanf("%d", p);              p++;              sum += \*p;      }      avg = sum / 10;      printf("avg=%f\n", avg);  } |

八．

#include <stdio.h>

void main(void)

{

    int i, sum=0;

float avg;

int sc[10], \*p=sc;

for (i=0; i<10; i++)

{

        scanf("%d", p);

        p++;

        sum+= \*(p-1);

}

avg = (float)sum/ 10;

printf("avg=%f", avg);

}

九．

从键盘输入10个整数，编程计算并输出其最大值、最小值及其所在元素的下标位置。

输入格式："%d"

输出格式："max=%d, pos=%d\n"

"min=%d, pos=%d\n"

输入样例：

1 2 3 4 5 6 7 8 9 10

输出样例：

max=10, pos=9

min=1, pos=0

九．

#include <stdio.h>

int main()

{

    int  a[10], n, max, min, maxPos, minPos;

    for (n = 0; n < 10; n++)

    {

        scanf("%d", &a[n]);

    }

    max = min = a[0];

    maxPos = minPos = 0;

    for (n = 0; n < 10; n++)

    {

        if (a[n] > max)

        {

            max = a[n];

            maxPos = n;

        }

        else if (a[n] < min)

        {

            min = a[n];

            minPos = n;

        }

    }

    printf("max=%d, pos=%d\n", max, maxPos);

    printf("min=%d, pos=%d\n", min, minPos);

    return 0;

}

十．

给定一维数组a，求a中各元素的平均值。

a[10]={1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9}

\*\*输出格式要求："The average of array is %f\n"

程序运行示例如下：

The average of array is 1.450000

十．

main()

{

    float average();

    static float a[10]={1.0,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8,1.9};

    printf("The average of array is %f\n",average(a,10));

}

float average(a,n)

    int n;

    float a[];

{

    int i;

    float sum=0;

    for(i=0;i<n;i++)

        sum+=a[i];

    return sum/n;

}

十一。

)\*约瑟夫问题。这是十七世纪的法国数学家加斯帕在《数目的游戏问题》中讲的一个故事：15个基督教徒和15个异教徒在海上遇险，必须将一半的人投入海中，其余的人才能幸免于难，于是想了个办法：30个人围成一个圆圈，从第一个人开始依次报数，每数到第9个人就将他扔入大海，如此循环进行直到仅余15个人为止。问怎样排法，才能使每次投入大海的都是异教徒。

\*\*输出格式要求："The original circle is (+:papandom, @:christian);\n" "%c "

程序运行示例如下：

The original circle is (+:papandom, @:christian);

@ @ @ @ + + + + + @ @ + @ @ @ + @ + + @ @ + + + @ + + @ @ +

十一。

#include <stdio.h>

struct node

{

  int nextp;

  int no\_out;

}link[31];

main()

{

  int i,j,k;

  printf("The original circle is (+:papandom,  @:christian);\n");

  for(i=1;i<=30;i++)

  {

    link[i].nextp=i+1;

    link[i].no\_out=1;

  }

  link[30].nextp=1;

  j=30;

  for(i=0;i<15;i++)

  {

     for(k=0;;)

     if(k<9)

     {

       j=link[j].nextp;

       k+=link[j].no\_out;

     }

     else break;

     link[j].no\_out=0;

  }

  for(i=1;i<=30;i++)

        printf("%c ",link[i].no\_out?'@':'+');

    printf("\n");

}

十二。

按如下函数原型编写程序，输入10个整数，用函数编程将其中最大数与最小数位置互换，然后输出互换后的数组。

void MaxMinExchang(int a[], int n);

int FindMaxPos(int s[], int n);

int FindMinPos(int s[], int n);

输入提示信息："Input 10 numbers:"

输入格式："%d"

输出提示信息："Exchang results:"

输出格式："%4d"

十二。

#include <stdio.h>

void MaxMinExchang(int a[], int n);

int FindMaxPos(int s[], int n);

int FindMinPos(int s[], int n);

int main()

{

    int i, a[10];

    printf("Input 10 numbers:");

    for (i = 0; i < 10; i++)

    {

        scanf("%d", &a[i]);

    }

    MaxMinExchang(a, 10);

    printf("Exchang results:");

    for (i = 0; i < 10; i++)

    {

        printf("%4d", a[i]);

    }

    printf("\n");

    return 0;

}

/\* 函数功能：将数组中的最大数与最小数位置互换 \*/

void MaxMinExchang(int a[], int n)

{

    int  maxPos, minPos, temp;

    maxPos = FindMaxPos(a, n);

    minPos = FindMinPos(a, n);

    temp = a[maxPos];

    a[maxPos] = a[minPos];

    a[minPos] = temp;

}

/\* 函数功能：计算数组中的最大值在数组中的下标位置 \*/

int FindMaxPos(int s[], int n)

{

    int maxPos = 0, max = s[0], i;

    for (i = 1; i < n; i++)

    {

        if (s[i] > max)

        {

            max = s[i];

            maxPos = i;

        }

    }

    return maxPos;

}

/\* 函数功能：计算数组中的最小值在数组中的下标位置 \*/

int FindMinPos(int s[], int n)

{

    int minPos = 0, min = s[0], i;

    for (i = 1; i < n; i++)

    {

        if (s[i] < min)

        {

            min = s[i];

            minPos = i;

        }

    }

    return minPos;

}

十三。

从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str)==0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

十三。

#include "stdio.h"

#include "string.h"

main()

{

    int i=0, findFlag=1;

    char x[13];

    char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

    printf("Input string:\n");

    gets(x);

    while( i<6 && findFlag)

    {

        if(strcmp(x,str[i])==0)

        {

            findFlag = 0;

        }

        i++;

    }

    if ( !findFlag )

    {

        printf("%s\n",x);

    }

    else

    {

        printf("Not find!\n");

    }

}

十四。

假设有40个学生被邀请来给餐厅的饮食和服务质量打分，分数划分为1～10这10个等级（1表示最低分，10表示最高分），编程统计并按如下格式输出餐饮服务质量调查结果。

Grade Count Histogram

1 5 \*\*\*\*\*

2 10 \*\*\*\*\*\*\*\*\*\*

3 7 \*\*\*\*\*\*\*

...

\*\*输入格式要求："%d" 提示信息："Input the feedbacks of 40 students:\n" "input error!\n"

\*\*输出格式要求："Feedback\tCount\tHistogram\n" "%8d\t%5d\t"

程序运行示例如下：

Input the feedbacks of 40 students:

10 9 10 8 7 6 5 10 9 8

8 9 7 6 10 9 8 8 7 7

6 6 8 8 9 9 10 8 7 7

9 8 7 9 7 6 5 9 8 7

Feedback Count Histogram

1 0

2 0

3 0

4 0

5 2 \*\*

6 5 \*\*\*\*\*

7 9 \*\*\*\*\*\*\*\*\*

8 10 \*\*\*\*\*\*\*\*\*\*

9 9 \*\*\*\*\*\*\*\*\*

10 5 \*\*\*\*\*

十四。

#include <stdio.h>

#define  M   40

#define  N   11

int main()

{

    int  i, j, grade, feedback[M], count[N] = {0};

    printf("Input the feedbacks of 40 students:\n");

    for (i = 0; i < M; i++)

    {

        scanf("%d", &feedback[i]);

    }

    for (i = 0; i < M; i++)

    {

        switch (feedback[i])

        {

        case 1:

            count[1]++;

            break;

        case 2:

            count[2]++;

            break;

        case 3:

            count[3]++;

            break;

        case 4:

            count[4]++;

            break;

        case 5:

            count[5]++;

            break;

        case 6:

            count[6]++;

            break;

        case 7:

            count[7]++;

            break;

        case 8:

            count[8]++;

            break;

        case 9:

            count[9]++;

            break;

        case 10:

            count[10]++;

            break;

        default:

            printf("input error!\n");

            break;

        }

    }

    printf("Feedback\tCount\tHistogram\n");

    for (grade = 1; grade <= N - 1; grade++)

    {

        printf("%8d\t%5d\t", grade, count[grade]);

        for (j = 0; j < count[grade]; j++)

        {

            printf("%c", '\*');

        }

        printf("\n");

    }

    return 0;

}

十五。

计算分数的精确值

使用数组精确计算M/N（0<M<N<=100）的值。如果M/N是无限循环小数，则计算并输出它的第一循环节，同时要求输出循环节的起止位置（小数位的序号）。

\*\*输入格式要求："%d/%d" 提示信息："please input a fraction (m/n)) (0< m< n<=100):"

\*\*输出格式要求："%d/%d it's accuracy value is:0." "\n\tand it is a infinite cyclic fraction from %d\n" "\tdigit to %d digit after decimal point.\n"

程序运行示例如下：

please input a fraction (m/n)) (0< m< n<=100):1/6

1/6 it's accuracy value is:0.16

and it is a infinite cyclic fraction from 2

digit to 2 digit after decimal point.

十五。

#include <stdio.h>

int remainder[101],quotient[101];

main()

{

    int m,n,i,j;

    printf("please input a fraction (m/n))  (0< m< n<=100):");

    scanf("%d/%d",&m,&n);

    printf("%d/%d it's accuracy value is:0.",m,n);

    for (i=1;i<=100;i++)

    {

        remainder[m]=i;

        m\*=10;

        quotient[i]=m/n;

        m=m%n;

        if(m==0)

        {

            for(j=1;j<=i;j++)

                printf("%d",quotient[j]);

            break;

        }

        if(remainder[m]!=0)

        {

            for (j=1;j<=i;j++)

                printf("%d",quotient[j]);

            printf("\n\tand it is a infinite cyclic fraction from %d\n",remainder[m]);

            printf("\tdigit to %d digit after decimal point.\n",i);

            break;

        }

    }

}

十六。

选美比赛

在选美大奖赛的半决赛现场，有一批选手参加比赛，比赛的规则是最后得分越高，名次

越低。当半决赛结束时，要在现场按照选手的出场顺序宣布最后得分和最后名次，获得

相同分数的选手具有相同的名次，名次连续编号，不用考虑同名次的选手人数。例如：

选手序号： 1，2，3，4，5，6，7

选手得分： 5，3，4，7，3，5，6

则输出名次为：3，1，2，5，1，3，4

请编程帮助大奖赛组委会完成半决赛的评分排名工作。

\*\*输出格式要求：" Player\_NO Score Rank\n" " %3d %4d %4d\n"

程序运行示例如下：

Player\_NO Score Rank

1 5 3

2 3 1

3 4 2

4 7 5

5 3 1

6 5 3

7 6 4

十六。

#include <stdio.h>

#define NUM 7

int a[NUM+1]={0,5,3,4,7,3,5,6};

int m[NUM+1],l[NUM+1];

main()

{

  int i,n,smallest,num,k,j;

  num=1;

  for(i=1;i<=NUM;i++)

    if(m[i]==0)

    {

        smallest=a[i];

        k=1;

        l[k]=i;

        for(j=i+1;j<=NUM;j++)

          if(m[j]==0)

             if(a[j]<smallest)

             {

                smallest=a[j];

                k=0;

                l[++k]=j;

             }

             else if(a[j]==smallest)

                    l[++k]=j;

         for(j=1;j<=k;j++)

           m[l[j]]=num;

           num++;

           i=0;

    }

    printf("  Player\_NO  Score  Rank\n");

    for(j=1;j<=NUM;j++)

       printf("   %3d    %4d   %4d\n",j,a[j],m[j]);

}

十七。

从键盘输入一个长度为N（比如10）的整型数组，而后将数组中小于零的元素移动到数组的前端，大于零的元素移动到数组的后端，等于零的元素留在数组中间。比如原始数组为：2 -5 -89 75 0 -89 0 93 48 0，经过处理后的数组为：-5 -89 -89 0 0 0 75 93 48 2。由于不要求数组有序，所以不允许用排序方法。

\*\*输入格式要求："%d" 提示信息："Please input %d interger number\n"

\*\*输出格式要求："%5d"

程序运行示例如下：

Please input 10 interger number

2 -5 -89 75 0 -89 0 93 48 0

-5 -89 -89 0 0 0 75 93 48 2

十七。

#include <stdio.h>

#define N 10

int main()

{

    int a[N];

    int i, p1 = 0, p2 = 9, p, temp;

    printf("Please input %d interger number\n", N);

    for (i = 0; i <= N - 1; i++)

    {

        scanf("%d", &a[i]);

    }

    for (; p1 < p2 ; )

    {

        if (a[p1] == 0)

        {

            p = p1;

            while (a[p1 + 1] == 0)

            {

                p1++;

            }

            if (p1 + 1 < p2)

            {

                temp = a[p1 + 1];

                a[p1 + 1] = a[p];

                a[p] = temp;

                p1 = p;

                continue;

            }

            else

            {

                break;

            }

        }

        if (a[p1] > 0)

        {

            temp = a[p1];

            a[p1] = a[p2];

            a[p2] = temp;

            p2--;

        }

        else

        {

            p1++;

        }

        if (a[p2] == 0)

        {

            p = p2;

            while (a[p2 - 1] == 0)

            {

                p2--;

            }

            if (p1 < p2 - 1)

            {

                temp = a[p2 - 1];

                a[p2 - 1] = a[p];

                a[p] = temp;

                p2 = p;

                continue;

            }

            else

            {

                break;

            }

        }

        if (a[p2] < 0)

        {

            temp = a[p2];

            a[p2] = a[p1];

            a[p1] = temp;

            p1++;

        }

        else

        {

            p2--;

        }

    }

    for (i = 0; i <= N - 1; i++)

    {

        printf("%5d", a[i]);

    }

    printf("\n");

}

十八。

以下程序的功能是从键盘输入10个整型数据，放入数组a 中，求其最大值、最小值及其所在元素的下标位置并输出。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  main()  {      int a[10], n, max, min, maxPos, minPos;        for (n=0, n<10, n++)      {          scanf("%d", a[n]);      }      max = min = a;      maxPos = minPos = 0;      for (n=1; n<10; n++)      {          if (a[n] > max)          {                  max = a[n];                  maxPos = n;          }          else if (a[n] < min)          {                  min = a[n];                  minPos = n;          }      }      printf("max=%d, pos=%d\n",max, maxPos);      printf("min=%d, pos=%d\n",min, minPos);  } |

十八。

#include <stdio.h>

main()

{   int a[10], n, max, min, maxPos, minPos;

    for (n=0; n<10; n++)

    {

        scanf("%d", &a[n]);

    }

    max = min = a[0];   //应初始化为数组中的某一元素

    maxPos = minPos = 0;

    for (n=1; n<10; n++)

    {

        if (a[n] > max)

        {

                max = a[n];

                maxPos = n;

        }

        else if (a[n] < min)

        {

                min = a[n];

                minPos = n;

        }

    }

    printf("max=%d, pos=%d\n",max, maxPos);

    printf("min=%d, pos=%d\n",min, minPos);

}

十九。

从键盘输入10个整型数据，放入数组a中，求其最大值，并输出。

\*\*要求输入提示信息为：无输入提示信息

\*\*要求输入格式为："%d"

\*\*要求输出格式为："max=%d\n"

十九。

#include <stdio.h>

main()

{

    int a[10],i,max;

    for(i=0;i<10;i++)

    {

        scanf("%d",&a[i]);  //2

    }

    max=a[0];  //2

    for(i=1;i<10;i++)  //2

    {

        if(max<a[i])  //2

        {

            max=a[i];  //2

        }

    }

    printf("max=%d\n",max);  //2

}

二十。

#include <stdio.h>

#include <math.h>

float a[5] = {2.33, 2.56, 2.65, 2.66, 2.30};

float t(float x,int n)

{

    float x1, x2;

    int i;

    for (x1 = x, i = 0; i < n - 1; i++)

        x1 \*= 10.0;

    x2 = floor((double)x1);

    x1 = floor((double)(x1 - x2) \* 10);

    if (x1 >= 5)

        x2 += 1;

    for (i = 0; i < n - 1; i++)

        x2 /= 10.0;

    return x2;

}

main()

{

    float t();

    int i;

    printf("Old array:\n");

    for (i = 0; i < 5; i++)

        printf("a[%d]=%.3f\t", i, a[i]);

    printf("\nnew array:\n");

    for (i = 0; i < 5; i++)

        printf("a[%d]=%.2f\t", i, t(a[i], 2));

}

二十。

编写一个C函数，实现保留小数点后第N-1位，从第N位四舍五入。用该函数对数组a[5]中的各元素从小数点后第2位开始四舍五入。

a[5]={2.33,2.56,2.65,2.66,2.30}

\*\*输出格式要求："Old array:\n" "a[%d]=%.3f\t" "\nnew array:\n" "a[%d]=%.2f\t"

程序运行示例如下：

Old array:

a[0]=2.330 a[1]=2.560 a[2]=2.650 a[3]=2.660 a[4]=2.300

new array:

a[0]=2.30 a[1]=2.60 a[2]=2.70 a[3]=2.70 a[4]=2.30

#include <stdio.h>

#include <math.h>

float a[5] = {2.33, 2.56, 2.65, 2.66, 2.30};

float t(float x,int n)

{

    float x1, x2;

    int i;

    for (x1 = x, i = 0; i < n - 1; i++)

        x1 \*= 10.0;

    x2 = floor((double)x1);

    x1 = floor((double)(x1 - x2) \* 10);

    if (x1 >= 5)

        x2 += 1;

    for (i = 0; i < n - 1; i++)

        x2 /= 10.0;

    return x2;

}

main()

{

    float t();

    int i;

    printf("Old array:\n");

    for (i = 0; i < 5; i++)

        printf("a[%d]=%.3f\t", i, a[i]);

    printf("\nnew array:\n");

    for (i = 0; i < 5; i++)

        printf("a[%d]=%.2f\t", i, t(a[i], 2));

}

二十一。

有n个人围成一圈，顺序排号。从第一人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

程序的运行示例如下：

请输入人数：30

30个人中最后留下的是第29号。

二十一。

/\* n个人循环报数逢3退出求最后一人是第几号 \*/

#include <stdio.h>

#define     NMAX    50 /\* 定义最多人数为50人 \*/

int main(void)

{

    int i, k, m, n, num[NMAX], \*p;

    printf("请输入人数：");

    scanf("%d", &n);

    p = num;

    for (i = 0; i < n; i++)

    {

        \*(p + i) = i + 1; /\* 以1至n为序给每个人编号 \*/

    }

    i = 0;

    k = 0; /\* k为按1,2,3报数时的计数变量 \*/

    m = 0; /\* m为退出人数 \*/

    while (m < n - 1)

    {

        if (\*(p + i) != 0)

        {

            k++;

        }

        if (k == 3)

        {

            \*(p + i) = 0; /\* 对退出人员的编号置为0 \*/

            k = 0;

            m++;

        }

        i++;

        if (i == n)

        {

            i = 0; /\* 报数到尾后i恢复为0 \*/

        }

    }

    while (\*p == 0)

    {

        p++;

    }

    printf("%d个人中最后留下的是第%d号。", n, \*p);

    return 0;

}

二十二。

)输入一行字符，分别统计其中每个元音字母（a、e、i、o、u不分大小写）的个数。

\*\*输入格式要求：信息提示："Input a line of characters:\n"

\*\*输出格式要求："4d"

程序的运行示例如下：

Input a line of characters:

How old are you?

1 1 0 3 1

二十二。

#include<stdio.h>

int main()

{

    char ch[80], \*p;

    int i, count[5] = {0};

    printf("Input a line of characters:\n");

    gets(ch);

    for (p = ch; \*p != '\0'; p++)

    {

        switch (\*p)

        {

        case 'a':

            count[0]++;

            break;

        case 'e':

            count[1]++;

            break;

        case 'i':

            count[2]++;

            break;

        case 'o':

            count[3]++;

            break;

        case 'u':

            count[4]++;

        }

    }

    for (i = 0; i < 5; i++)

    {

        printf("%4d", count[i]);

    }

    return 0;

}

二十三。

选择排序法。用选择排序法将N个数从小到大排序后输出。

\*\*输入格式要求："%d" 提示信息："Enter No.%2d:"

\*\*输出格式要求："%d"

程序运行示例如下：

Enter No. 1:5

Enter No. 2:7

Enter No. 3:3

Enter No. 4:9

Enter No. 5:8

35789

二十三。

#include <stdio.h>

#define N 5

main()

{ int i,j,t,p,a[N];

  for(i=0;i<N;i++)

 { printf("Enter No.%2d:",i+1);

   scanf("%d",&a[i]);

 }

 for(i=0;i<N-1;i++)

 { p=i;

   for(j=i+1;j<N;j++)

     if(a[p]>a[j])

     p=j;

     if(p!=i)

     {t=a[p];a[p]=a[i];a[i]=t;

     }

  }

  for(i=0;i<N;i++)

      printf("%d",a[i]);

}

二十四。

有n个人围成一圈，顺序排号。从第一个人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

\*\*输入格式要求："%d" 提示信息："please input the total of numbers:"

\*\*输出格式要求："%d is left\n"

程序运行示例如下：

please input the total of numbers:30

2 is left

二十四。

#include <stdio.h>

#define nmax 50

int main()

{

    int i, k, m, n, num[nmax], \*p;

    printf("please input the total of numbers:");

    scanf("%d", &n);

    p = num;

    for (i = 0; i < n; i++ )

        \*(p + i) = i + 1;

    i = 0;

    k = 0;

    m = 0;

    while (m < n)

    {

        if (\*(p + i) != 0)

            k++;

        if (k % 3 == 0)

        {

            \*(p + i) = 0;

            m++;

        }

        i++;

        if (i == n)

            i = 0;

    }

    while (\*p == 0)

        p++;

    printf("%d is left\n", \*p);

}

二十五。

假设今年的工业产值为100万元，产值增长率为每年c%，请编程计算当c分别为6、8、10、12时工业产值分别过多少年可实现翻一番（即增加一倍）。

输入格式：无

输出格式："When grow rate is %d%%, the output can be doubled after %d years.\n"

二十五。

#include <stdio.h>

#define  CURRENT  100

#define  N      4

main()

{

    int  i;

    int  year;                              /\* 产值翻番所需年数 \*/

    int   growRate[N] = {6, 8, 10, 12};     /\* 工业产值的增长率 \*/

    double  output;                         /\* 工业总产值 \*/

    for (i = 0; i < N; i++)

    {

        output = CURRENT;                /\* 当年产值为100万元 \*/

        for (year = 0; output < 2 \* CURRENT; year++)

        {

            output = output \* (1 + growRate[i] / 100.0);

        }

        printf("When grow rate is %d%%, the output can be doubled after

               %d years.\n", growRate[i], year);

    }

}

二十六。

程序改错。以下程序用于打印输出结构体数组的所有元素。

struct s

{

    char name[10];

    int age;

}

main()

{

    struct s a[3] = {”John”,19,”Paul”,17,”Marry”,18};

    int \*p;

    for(p = a; p < 3; p++)

    {

        printf("%s,%d\n", p->name, p->age );

    }

}

二十六。

struct s

{

    char name[10];

    int age;

};

main()

{

    struct s a[3] = {"John",19,"Paul",17,"Marry",18};

    struct s \*p;

    for(p = a; p < a+3; p++)

    {

        printf("%s,%d\n", p->name, p->age );

    }

}

二十七。

输入10个学生的C语言成绩（假设成绩为整数值），求最高成绩、最低成绩和平均值。

\*\*输入格式要求："%d" 提示信息："请输入10个学生的C成绩：\n"

\*\*输出格式要求："%d " "10个学生的C成绩分别为：\n" "最高成绩是%d,是第%d个学生\n" "最低成绩是%d,是第%d个学生\n" "10名学生的C成绩的平均分为:%.2f\n"

程序运行示例如下：

请输入10个学生的C成绩：

80 90 93 78 83 67 79 60 64 45

10个学生的C成绩分别为：

80 90 93 78 83 67 79 60 64 45

最高成绩是93,是第3个学生

最低成绩是45,是第10个学生

10名学生的C成绩的平均分为:73.90

二十七。

#include <stdio.h>

int main()

{

    int cScore[10], cMax, cMin, cMaxIndex, cMinIndex, cSum;

    //cMax,cMine,cSum分别存放10个学生中最高成绩及最低成绩、成绩总和

    int i;

    /\*输入10个学生的成绩\*/

    printf("请输入10个学生的C成绩：\n");

    for (i = 0; i < 10; i++)

        scanf("%d", &cScore[i]);

    /\*计算平均值、最高分数和最低分数,假设第一个学生的成绩是最高和最低\*/

    cMax = cMin = cScore[0];

    cSum = cScore[0];

    cMaxIndex = cMinIndex = 0;

    for (i = 1; i < 10; i++)

    {

        cSum = cSum + cScore[i];

        if (cScore[i] > cMax)

        {

            cMax = cScore[i];

            cMaxIndex = i;

        }

        if (cScore[i] < cMin)

        {

            cMin = cScore[i];

            cMinIndex = i;

        }

    }

    /\*打印结果\*/

    printf("10个学生的C成绩分别为：\n");

    for (i = 0; i < 10; i++)

        printf("%d ", cScore[i]);

    printf("\n");

    printf("最高成绩是%d,是第%d个学生\n", cMax, cMaxIndex + 1);

    printf("最低成绩是%d,是第%d个学生\n", cMin, cMinIndex + 1);

    printf("10名学生的C成绩的平均分为:%.2f\n", (1.0 \* cSum) / 10);

    return 0;

}

二十八。

编程实现显示用户输入的月份（不包括闰年的月份）拥有的天数。

\*\*输入格式要求："%d" 提示信息："Input a month:" "The number of days is %d\n"

\*\*输出格式要求："%d"

程序的运行示例如下：

Input a month:13

Input a month:2

The number of days is 28

二十八。

#include <stdio.h>

#define  MONTHS 12

int main()

{

    int days[MONTHS] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

    int month;

    do

    {

        printf("Input a month:");

        scanf("%d", &month);

    }

    while (month < 1 || month > 12); /\* 处理不合法数据的输入 \*/

    printf("The number of days is %d\n", days[month - 1]);

    return 0;

}

二十九。

编程实现从键盘任意输入20个整数，统计非负数个数，并计算非负数之和。

输入提示信息："Input 20 Numbers:\n"

输入格式："%d"

输出格式："sum = %d, counter = %d\n"

运行示例：

Input 20 Numbers:

1 2 3 4 5 6 7 8 -9 -10↙

12 -45 56 -99 34 87 90 -23 0 65↙

sum = 380, counter = 15

二十九。

#include  <stdio.h>

int main(void)

{

    int i, n, sum = 0, counter = 0;

    printf("Input 20 Numbers:\n");

    for (i = 0; i < 20; i++)

    {

        scanf("%d", &n);

        if (n >= 0)              //判断是否为非负数

        {

            sum += n;           //非负数求和

            counter++;          //非负数个数计算

        }

    }

    printf("sum = %d, counter = %d\n", sum, counter);

    return 0;

}

二十九。

#include  <stdio.h>

int main(void)

{

    int i, n, sum = 0, counter = 0;

    printf("Input 20 Numbers:\n");

    for (i = 0; i < 20; i++)

    {

        scanf("%d", &n);

        if (n >= 0)              //判断是否为非负数

        {

            sum += n;           //非负数求和

            counter++;          //非负数个数计算

        }

    }

    printf("sum = %d, counter = %d\n", sum, counter);

    return 0;

}

三十。

乘式还原。有乘法算式如下：

 ×     o   o o  o o    o  o o  o o  o o  o    o  o  o  o  o

18个O的位置上全部是素数（2、3、5或7），请还原此算式。

程序运行示例如下：

775

\* 33

-----

2325

2325

-----

25575

三十。

#include <stdio.h>

#define NUM 5

#define C\_NUM 4

int a[NUM+1];

int b[]={0,2,3,5,7};

main()

{

    int i,not\_finish=1;

    i=2;

    a[1]=1;

    while(not\_finish)

    {

        while(not\_finish&&i<=NUM)

            if(a[i]>=C\_NUM)

                if(i==1&&a[i]==C\_NUM)

                    not\_finish=0;

                else a[i--]=0;

            else a[i++]++;

        if(not\_finish)

        {

            long int sum1,sum2,sum3,sum4;

            sum1=b[a[1]]\*100+b[a[2]]\*10+b[a[3]];

            sum2=sum1\*b[a[5]];

            sum3=sum1\*b[a[4]];

            if(sum2>=2222&&sum2<=7777&&f(sum2)&&sum3>=2222&&sum3<=7777&&f(sum3))

                if((sum4=sum2+sum3\*10)>=22222&&sum4<=77777&&f(sum4))

                {

                    printf("  %d\n",sum1);

                    printf("\*  %d%d\n",b[a[4]],b[a[5]]);

                    printf("-----\n");

                    printf(" %d\n",sum2);

                    printf("%d\n",sum3);

                    printf("-----\n");

                    printf("%d\n",sum4);

                }

             i=NUM;

        }

    }

}

f(sum)

long sum;

{   int i,k,flag;

    while(sum>0)

    {

        i=sum%10;

        for(flag=0,k=1;!flag&&k<=C\_NUM;k++)

        if(b[k]==i)

        {

            flag=1;

            break;

        }

        if(!flag)   return(0);

        else      sum=sum/10;

    }

    return (1);

}

三十一。

)魔术师利用一副牌中的13张红桃，预先将它们排好后迭在一起，牌面朝下。对观众说：“我不看牌”，只数数就可以猜到每张牌是什么，我大声数，你们听，不信？你们就看。魔术师将最上面的那张牌数为1，把它翻过来正好是红桃A，将红桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌的下面，将第二张牌翻过来，正好是红桃2，也将它放在桌子上。第三次数1、2、3，将前面两张依次放在这迭牌的下面，再翻第三张牌正好是红桃3.这样依次将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

输入格式要求：无

输出格式要求："%d "（%d后面有一个空格），全部数据输出后再输出一个换行"\n"

程序运行示例：

1 8 2 5 10 3 12 11 9 4 7 6 13

三十一。

#include <stdio.h>

int main()

{

    int cards[14] = {0};

    int i, j = 1, n;

    for (i = 1 ; i <= 13 ; i++)

    {

        n = 1;

        do

        {

            if (j > 13)

                j = 1;

            if (cards[j] > 0)

                j++;

            else

            {

                if (n == i)

                    cards[j] = i;

                j++;

                n++;

            }

        }

        while (n <= i);

    }

    for (i = 1; i <= 13; i++)

        printf("%d ", cards[i]);

    printf("\n");

    return 0;

}

三十二。

对候选人得票的统计程度。设由3个候选人，每次输入一个得票的候选人的名字，要求最后输入各人得票结果。（总票数为n）请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

struct person

{

char name[20];

int count;

}

person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

int i, j;

char leader\_name[20];

scanf("%d\n", &n);

for (i = 0; i < n; i++)

{

scanf("%s", leader\_name);

for (j = 0; j < 3; j++)

if (leader\_name == leader[j].name)

leader[j].count++;

}

printf("\n");

for (i = 0; i < 3; i++)

printf("%5s:%d\n", leader[i].name, leader[i].count);

}

三十二。

#include <string.h>

struct person

{

    char name[20];

    int count;

};

struct person leader[3] = {"Li", 0, "Zhang", 0, "Fun", 0};

main()

{

    int i, j, n;

    char leader\_name[20];

    scanf("%d\n", &n);

    for (i = 0; i < n; i++)

    {

        scanf("%s", leader\_name);

        for (j = 0; j < 3; j++)

            if (!strcmp(leader\_name, leader[j].name))

                leader[j].count++;

    }

    printf("\n");

    for (i = 0; i < 3; i++)

        printf("%5s:%d\n", leader[i].name, leader[i].count);

}

三十二。

数字排列 在较中的九个顶点上，空出中间的点，其余的点上任意填入数字1至8；1的位置固定不动，然后移动其余数字，使1到8顺时针从小到大排列。移动的规则是：只能将数字沿线移向空白的点。请编程显示数字移动过程。

○—○—○

︱＼︱／︱

○—◇—○

︱／︱＼︱

○—○—○

\*\*输入格式要求："%d" 提示信息："Please enter original order of digits 1~8:"

\*\*输出格式要求："%2d\n" "The sorting process is as felow:\n" "---%2d---\n"

程序运行示例如下：

Please enter original order of digits 1~8:1 2 3 4 5 6 7 8

The sorting process is as felow:

1 2 3

8 0 4

7 6 5

--- 0---

三十二。

int a[9]={0,1,2,5,8,7,6,3};

int b[9];

int c[9];

int count=0;

main()

{int i,j,k,t;

void print();

printf("Please enter original order of digits 1~8:");

for (i=0;i<8;i++)

    scanf("%d",&b[a[i]]);

printf("The sorting process is as felow:\n");

print();

for(t=-1,j=0;j<8 && t==-1;j++)

   if(b[a[j]==1]) t=j;

for(j=0;j<8;j++)

    c[j]=a[(j+t)%8];

for(i=2;i<9;i++)

    for(j=i-1;j<8;j++)

    if(b[c[j]]==i&&j!=i-1)

    { b[4]=i;

      b[c[j]]=0;

      print();

      for (k=j;k!=i-1;k--)

      {

          b[c[k]]=b[c[k-1]];

          b[c[k-1]]=0;

          print();

      }

      b[c[k]]=i;

      b[4]   =0;

      print();

      break;

    }

    else if(b[c[j]==i])

         break;

}

void print(void)

{ int c;

  for(c=0;c<9;c++)

    if(c%3==2)

        printf("%2d\n",b[c]);

    else

        printf("%2d",b[c]);

  printf("---%2d---\n",count++);

}

三十三。

反向输出有5个元素的int数组中的各元素的值。数组为{10,20,30,40,50}。

\*\*输出格式要求："\ta[%d]=%d,"

程序运行示例如下：

a[4]=50, a[3]=40, a[2]=30, a[1]=20, a[0]=10,

三十三。

#include <stdio.h>

main()

{

    static int a[5]={10,20,30,40,50};

    int i=5;

    while(--i>=0)

        printf("\ta[%d]=%d,",i,a[i]);

}

三十四。

求有10个整型元素的数组中最大元素及其下标。

\*\*输入格式要求："%d" 提示信息："请输入10个元素的值：\n"

\*\*输出格式要求："值最大的元素值为：%d 下标为%d\n"

程序运行示例1如下：

请输入10个元素的值：

23 45 36 11 20 90 31 66 30 33

值最大的元素值为：90 下标为5

三十四。

#include <stdio.h>

int main()

{

    int a[10];

    int i, index;//i表示循环变量，index用于存放值最大的元素下标

    //输入数组元素

    printf("请输入10个元素的值：\n");

    for (i = 0; i < 10; i++)

    {

        scanf("%d", &a[i]);

    }

    //求值最大的元素及其对应下标

    index = 0;

    for (i = 1; i < 10; i++)

    {

        if (a[i] > a[index])

            index = i;

    }

    //输出最大的元素值及其下标

    printf("值最大的元素值为：%d  下标为%d\n", a[index], index);

    return 0;

}

三十五。

某个公司采用公用电话传递数据，数据是四位的整数，在传递过程中是加密的，加密规则如下： 每位数字都加上5,然后用和除以10的余数代替该数字，再将第一位和第四位交换，第二位和第三位交换。

\*\*输入格式要求："%d"

\*\*输出格式要求："%d"

程序运行示例如下：

输入:3421

输出:8976

三十五。

int main()

{

    int i, aa[4], t;

    for (i = 0; i <= 3; i++)

    {

        scanf("%d", &aa[i]);

    }

    for (i = 0; i <= 3; i++)

    {

        aa[i] += 5;

        aa[i] %= 10;

    }

    for (i = 0; i <= 3 / 2; i++)

    {

        t = aa[i];

        aa[i] = aa[3 - i];

        aa[3 - i] = t;

    }

    for (i = 3; i >= 0; i--)

        printf("%d", aa[i]);

}

三十六。

编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

三十六。

#include <stdio.h>

#include <stdlib.h>

#define N 100

int getline(char \*s, int len);

int main()

{

    char line[N], b[20], \*ptr1, \*ptrb;

    int n = 0, j, a[50];

    printf("Please enter a string:");

    getline(line, N);

    ptr1 = line;

    while (\*ptr1 != '\0')

    {

        ptrb = b;

        for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)

        {

            \*ptrb = \*ptr1;

        }

        if (j)

        {

            \*ptrb = '\0';

            \*(a + n) = atoi(b);

            n++;

        }

        if (\*ptr1 != '\0')ptr1++;

    }

    printf("the result of output:\n");

    for (j = 0; j < n; j++)

    {

        printf("%10d\n", \*(a + j));

    }

    return 0;

}

int getline(char \*s, int len)

{

    int c;

    char \*p = s;

    while (--len > 0 && (c = getchar()) != '\n')

    {

        \*s++ = c;

    }

    \*s = '\0';

    return s - p;

}

三十七。

#include <stdio.h>

#include <math.h>

#define NUM 320

int number[NUM];

main()

{

    int i,j,count=0;

    printf("There are follwing primes in first row :\n");

    for(j=0,i=3;i<=1999;i+=2)

        if (fflag(i)) number[j++]=i;

    for(j--;number[j]>1898;j--)

    {

        for(i=0;number[j]-number[i]>1898;i++);

        if(number[j]-number[i]==1898)

            printf ("(%d).%3d,......,%d\n",++count,number[i],number[j]);

    }

}

fflag(i)

int i;

{

    int j;

    if (i<=1) return(0);

    if (i==2)  return(1);

    if(!(i%2)) return(0);

    for(j=3;j<=(int)( sqrt((double)i)+1);j+=2)

        if (!(i%j))  return(0);

    return(1);

}

三十七。

求1898

现將不超过2000的所有素数从到大排成第一行，第二行上的每个数都等于它“右肩”上的素数与“左肩”上的素数之差。请编程求出：第二行数中是否存在这样的若干个连续的整数，它们的和恰好是1898？假如存在的话，又有几种这样的情况？

\*\*输入格式要求：提示信息："There are follwing primes in first row :\n"

\*\*输出格式要求："(%d).%3d,......,%d\n"

程序运行示例如下：

There are follwing primes in first row :

(1).101,......,1999

(2). 89,......,1987

(3). 53,......,1951

(4). 3,......,1901

三十八。

编写程序：用整数型数组a存放一个8位长整型数（每个数组元素为一个一位整数），判断该数是否为对称数。该数由键盘输入。例如：132231为对称数。

\*\*输入格式要求："%d"

\*\*输出格式要求："no,it is not\n" "yes,it is\n"

程序运行示例1如下：

1

2

3

4

4

3

2

1

输出

yes,it is

程序运行2示例如下：

2

3

4

4

3

2

1

0

输出

no,it is not

三十八。

#include <stdio.h>

main()

{

    int i, j, a[8];

    for (i = 0; i < 8; i++)

        scanf("%d", &a[i]);

    for (i = 0, j = 7; i <= j; i++, j--)

        if (a[i] != a[j])  break;

    if (i < j)  printf("no,it is not\n");

    else  printf("yes,it is\n");

}

三十九。

输入一行字符，统计其中的英文字符、数字字符、空格和其他字符的个数。

输入提示信息："Please input a string:"

输入字符串用gets()

输出提示信息和格式：

"English character: %d\n"

"digit character: %d\n"

"space: %d\n"

"other character: %d\n"

三十九。

#include <stdio.h>

#include <string.h>

#define ARR\_SIZE 80

int main()

{

    char str[ARR\_SIZE];

    int  len, i, letter = 0, digit = 0, space = 0, others = 0;

    printf("Please input a  string:");

    gets(str);

    for (i = 0; str[i] != '\0'; i++)

    {

        if (str[i] >= 'a' && str[i] <= 'z' || str[i] >= 'A' && str[i] <= 'Z')

            letter ++;              //统计英文字符

        else if (str[i] >= '0' && str[i] <= '9' )

            digit ++;               //统计数字字符

        else if (str[i] == ' ' )

            space ++;               //统计空格

        else

            others ++;              //统计其他字符的个数

    }

    printf("English character:  %d\n", letter);

    printf("digit character:  %d\n", digit);

    printf("space:  %d\n", space);

    printf("other character:  %d\n", others);

    return 0;

}

四十。

编程统计候选人的得票数。设有3个候选人zhang、li、wang，10个选民，选民每次输入一个得票的候选人的名字，若选民输错候选人姓名，则按废票处理。选民投票结束后程序自动显示各候选人的得票结果和废票信息。要求用结构体数组candidate表示3个候选人的姓名和得票结果。

四十。

#include  <stdio.h>

#include  <string.h>

#define NUM\_ELECTORATE 10

#define NUM\_CANDIDATE 3

struct candidate

{

    char  name[20];

    int   count;

} candidate[3] = {"li", 0, "zhang", 0, "wang", 0};

int main()

{

    int  i, j, flag = 1, wrong = 0;

    char  name[20];

    for (i = 1; i <= NUM\_ELECTORATE; i++)

    {

        printf("Input vote %d:", i);

        scanf("%s", name);

        flag = 1;

        for (j = 0; j < NUM\_CANDIDATE; j++)

        {

            if (strcmp(name, candidate[j].name) == 0)

            {

                candidate[j].count++;

                flag = 0;

            }

        }

        if (flag)

        {

            wrong++;  /\* 废票计数 \*/

            flag = 0;

        }

    }

    printf("Election results:\n");

    for (i = 0; i < NUM\_CANDIDATE; i++)

    {

        printf("%8s:%d\n", candidate[i].name, candidate[i].count);

    }

    printf("Wrong election:%d\n", wrong);

    return 0;

}

四十一。

满足特异条件的数列。输入m和n（20≥m≥n≥0），求出满足以下方程式的正整数数列i1,i2,…,in，使得i1+i2+…+in=m，且i1≥i2≥…≥in。例如：

当n=4，m=8时，将得到如下5个数列：

5 1 1 1 4 2 1 1 3 3 1 1 3 2 2 1 2 2 2 2

\*\*输入格式要求："%d" 提示信息："Please enter requried terms (<=10):"

\*\*输出格式要求："There are following possible series:\n" "[%d]:"

四十一。

#define NUM 10

#include <stdio.h>

int i[NUM];

main()

{ int sum, n,total,k,flag,count=0;

  printf("Please enter requried terms (<=10):");

  scanf("%d",&n);

  printf("                             their sum:");

  scanf("%d",&total);

  sum=0;

  k=n;

  i[n]=1;

  printf("There are following possible series:\n");

  while(1)

  {

    if(sum+i[k]<total)

        if(k<=1)

        {i[1]=total-sum;flag=1;}

        else

        {sum+=i[k--];

         i[k]=i[k+1];

         continue;

         }

    else if(sum+i[k]>total||k!=1)

        {sum-=i[++k];

        flag=0;

        }

        else

        flag=1;

    if(flag)

    {

        printf("[%d]:",++count);

        for(flag=1;flag<=n;++flag)

            printf("%d",i[flag]);

        printf("\n");

    }

    if(++k>n)

        break;

    sum-=i[k];

    i[k]++;

  }

}

四十二。

魔术师利用一副牌中的13张红桃，预先将它们排好后迭在一起，牌面朝下。对观众说：“我不看牌”，只数数就可以猜到每张牌是什么，我大声数，你们听，不信？你们就看。魔术师将最上面的那张牌数为1，把它翻过来正好是红桃A，将红桃A放在桌子上，然后按顺序从上到下数手中的余牌，第二次数1、2，将第一张牌放在这迭牌的下面，将第二张牌翻过来，正好是红桃2，也将它放在桌子上。第三次数1、2、3，将前面两张依次放在这迭牌的下面，再翻第三张牌正好是红桃3.这样依次将13张牌全翻出来，准确无误。问魔术师手中的牌原始次序是怎样安排的？

\*\*输出格式要求："%d "

四十二。

#include<stdio.h>

int main()

{

    int cards[14] = {0};

    int i, j = 1, n;

    for (i = 1 ; i <= 13 ; i++)

    {

        n = 1;

        do

        {

            if (j > 13)

                j = 1;

            if (cards[j] > 0)

                j++;

            else

            {

                if (n == i)

                    cards[j] = i;

                j++;

                n++;

            }

        }

        while (n <= i);

    }

    for (i = 1; i <= 13; i++)

        printf("%d ", cards[i]);

    printf("\n");

    return 0;

}

四十三。

下面程序的功能是将数组元素倒置 ，找出其中的4处错误并改正之。

#include "stdio.h"

#define M 5;

main()

{   int a[M]={1,2,3,4,5};

    int i,j,\*t;

    i=0;j=M-1;

    while(i)

    {

        t=\*(a+i);

        \*(a+i)=\*(a+j);

        \*(a+j)=t;

        i++;j++

    }

    for(i=0;i<M;i++) printf("%d",(a+i));

}

四十三。

#include "stdio.h"

#define M 5

main()

{   int a[M]={1,2,3,4,5};

    int i,j,t;

    i=0;j=M-1;

    while(i<=j)

    {

        t=\*(a+i);

        \*(a+i)=\*(a+j);

        \*(a+j)=t;

        i++;j--;

    }

    for(i=0;i<M;i++) printf("%d",\*(a+i));

}

四十四。

下面程序实现折半查找算法，当找到输入元素后显示其在数组中的下标。找出其中的错误，并改正之

|  |
| --- |
| #include <stdio.h>  main()  {      int  up=10, low=1, mid, found, find;      int  a[10]={1, 5, 6, 9, 11, 17, 25, 34, 38, 41};        scanf("%d", find);      printf("\n");        while (up>=low  ||  !found)      {          mid=(up+low)/2;          if( a[mid] = find )          {                      found=1;                  break;          }           else  if(a[mid]>find)                  up=mid+1;              else                          low=mid+1;     }        if(found)  printf("found  number  is  %dth", mid);      else    printf("no  found");  } |

四十四。

#include <stdio.h>

main( )

{

int  up=9, low=0, mid,found=0, find;

  int  a[10]={1, 5, 6, 9, 11, 17, 25, 34, 38, 41};

    scanf("%d", &find);

  printf("\n");

  while (up>=low  &&  !found)

       {

                mid=(up+low)/2;

            if(a[mid]==find)

                {

                            found=1;

                            break;

                        }

            else  if(a[mid]>find)

                up=mid-1;

            else

                        low=mid+1;

        }

if(found) printf("found number is %dth",mid);

else  printf("no  found");

}

四十五。

13个人围成一圈，从第1个人开始顺序报号1、2、3，凡报到3的人退出圈子。找出最后留在圈子里的人原来的序号。要求用结构体编程实现。

程序的运行示例如下：

出圈成员及顺序： 3 6 9 12 2 7 11 4 10 5 1 8

最后的成员是： 13

四十五。

#include <stdio.h>

#define     N   13

struct person

{

    int number;

    int nextp;

} link[N + 1];

int main(void)

{

    int i, count, h;

    /\* 建立队列 \*/

    for (i = 1; i <= N; i++)

    {

        if (i == N)

        {

            link[i].nextp = 1;

        }

        else

        {

            link[i].nextp = i + 1;

        }

        link[i].number = i;

    }

    printf("\n");

    count = 0;

    h = N;

    printf("出圈成员及顺序：");

    while (count < N - 1)

    {

        i = 0;

        while (i != 3)

        {

            h = link[h].nextp;

            if (link[h].number)

            {

                i++;

            }

        }

        printf("%3d", link[h].number);

        link[h].number = 0;

        count++;

    }

    printf("\n最后的成员是：");

    for (i = 1; i <= N; i++)

    {

        if (link[i].number)

        {

            printf("%3d", link[i].number);

        }

    }

    return 0;

}

四十六。

设有a[n],b[n]。求a[0]\*b[0]+a[1]\*b[1]+...+a[n-1]\*b[n-1] 。其中n，及数组a和b的元素值由键盘输入。

程序运行示例如下：

Input n

3

Input array a

1 2 3

Input array b

4 5 6

sum=32.000000

四十六。

#include <stdio.h>

float product(float\*pa, float \*pb, int n)

{

    int i;

    float s;

    for (i = 0, s = 0; i < n; i++)

    {

        s = s + (\*pa \* \*pb);

        pa++;

        pb++;

    }

    return s;

}

int main(int argc, char\* argv[])

{

    float \*pa, \*pb, sum;

    int i, n;

    printf("Input n\n");

    scanf("%d", &n);

    pa = (float\*) malloc(n \* sizeof(float));

    pb = (float\*) malloc(n \* sizeof(float));

    printf("Input array a\n");

    for (i = 0; i < n; i++)

    {

        scanf("%f", pa);

        pa++;

    }

    printf("Input array b\n");

    for (i = 0; i < n; i++)

    {

        scanf("%f", pb);

        pb++;

    }

    pa = pa - n;

    pb = pb - n;

    sum = product(pa, pb, n);

    printf("\nsum=%f\n", sum);

    return 0;

}

四十七。

在下面给出的程序中，共有5处错误（包括语法错误和逻辑错误），请找出其中的错误，并改正之。本程序的功能是对a数组中的元素按从大到小的顺序排序。

#include <stdio.h>

#define N 10;

main()

{

    int a[N]={2,17,8,3,24,53,82,1,29,101};

    int i,j,k,t;

    for(i=0;i<9;i++)

    {       k=i;

            for(j=i+1;j<10;j++)

                if(a[j]<a[k])    k=j;

            if(k=i)

            {

                t=a[j];

                a[j]=a[i];

                a[i]=t;

                        }

    }

        for(i=0;i<9;i++)

                printf("%d ",a[i]);

}

四十七。

#include <stdio.h>

#define N 10

main()

{

    int a[N]={2,17,8,3,24,53,82,1,29,101};

    int i,j,k,t;

    for(i=0;i<9;i++)

    {       k=i;

            for(j=i+1;j<10;j++)

                if(a[j]>a[k])    k=j;

            if(k!=i)

            {

                t=a[k];

                a[k]=a[i];

                a[i]=t;

            }

    }

    for(i=0;i<9;i++)

            printf("%d ",a[i]);

}

四十八。

程序：检查数中重复出现的数字。

用户输入数后，程序显示信息Repeated digit或No Repeated digit：

Enter a number: 28212

Repeated digit

数28212有一个重复的数字（2），而数9357则没有。

\*\*输入格式要求："%ld" 提示信息："Enter a number :"

\*\*输出格式要求："Repeated digit\n\n" "No Repeated digit\n\n"

四十八。

#include <stdio.h>

#define TRUE    1

#define FALSE   0

typedef int Bool;

int main()

{

    Bool digit\_seen[10] = {0};

    int digit;

    long int n;

    printf("Enter a number :");

    scanf("%ld", &n);

    while (n > 0)

    {

        digit = n % 10;

        if (digit\_seen[digit])

            break;

        digit\_seen[digit] = TRUE;

        n /= 10;

    }

    if (n > 0)

        printf("Repeated digit\n\n");

    else

        printf("No Repeated digit\n\n");

    return 0;

}

四十九。

如果一个正整数等于其各个数字的立方和，则该数称为阿姆斯特朗数（亦称为自恋性数）。如407=4^3+0^3+7^3就是一个阿姆斯特朗数。试编程求1000以内的所有阿姆斯特朗数。

\*\*输出格式要求："There are following Armstrong number smaller than 1000:\n" " %d "

程序运行示例如下：

There are following Armstrong number smaller than 1000:

153 370 371 407

四十九。

#include <stdio.h>

main()

{

    int i,t,k,a[4];

    printf("There are following Armstrong number smaller than 1000:\n");

    for(i=2;i<1000;i++)

    {

        for(t=0,k=1000;k>=10;t++)

        {

            a[t]=(i%k)/(k/10);

            k/=10;

        }

        if(a[0]\*a[0]\*a[0]+a[1]\*a[1]\*a[1]+a[2]\*a[2]\*a[2]==i)

            printf(" %d ",i);

    }

}

五十。

程序改错。下面程序的功能是从键盘输入一行字符，统计其中有多少单词。假设单词之间以空格分开。[提示：判断是否有新单词出现的方法是——当前被检验的字符不是空格，而前一被检验字符是空格，则表示有新单词出现。]

#include<stdio.h>

#define ARR\_SIZE 80;

main()

{

    char str[ARR\_SIZE];

    int i, num;

    gets(str);

    if(str[0] != ' ' && str[0] != '\0')

    {

        num = 1;

    }

    else

    {

        num = 0;

    }

    for(i = 1 ; str[i] != '\0'; i++)

    {

        if(str[i]  != ' ' && str[i-1]  = ' ')

        {

            num++;

        }

    }

    printf("num=%d\n", num);

}

五十。

#include<stdio.h>

#define ARR\_SIZE 80

main()

{

    char str[ARR\_SIZE];

    int i, num;

    gets(str);

    if(str[0] != ' ' && str[0] != '\0')

    {

        num = 1;

    }

    else

    {

        num = 0;

    }

    for(i = 1 ; str[i] != '\0'; i++)

    {

        if(str[i]  != ' ' && str[i-1]  == ' ')

        {

            num++;

        }

    }

    printf("num=%d\n", num);

}

五十一。

)按如下函数原型编写程序，用一个整型数组feedback保存调查的40个反馈意见。用函数编程计算反馈意见的平均数（Mean）、中位数（Median）和众数（Mode）。中位数指的是排列在数组中间的数。如果原始数据的个数是偶数，那么中位数等于中间那两个元素的算术平均值。众数是数组中出现次数最多的那个数（不考虑两个或两个以上的反馈意见出现次数相同的情况）。

int Mean(int answer[], int n);

int Median(int answer[], int n);

int Mode(int answer[], int n);

void DataSort(int a[], int n);

输入提示信息："Input the feedbacks of 40 students:\n"

输入格式："%d"

输出提示信息和输出格式：

"Mean value = %d\n"

"Median value = %d\n"

"Mode value = %d\n"

五十一。

#include <stdio.h>

#define  M   40

#define  N   11

int Mean(int answer[], int n);

int Median(int answer[], int n);

int Mode(int answer[], int n);

void DataSort(int a[], int n);

int main()

{

    int  i, feedback[M];

    printf("Input the feedbacks of 40 students:\n");

    for (i = 0; i < M; i++)

    {

        scanf("%d", &feedback[i]);

    }

    printf("Mean value = %d\n", Mean(feedback, M));

    printf("Median value = %d\n", Median(feedback, M));

    printf("Mode value = %d\n", Mode(feedback, M));

    return 0;

}

/\* 函数功能：若n>0则计算并返回n个数的平均数，否则返回-1 \*/

int Mean(int answer[], int n)

{

    int i, sum = 0;

    for (i = 0; i < n; i++)

    {

        sum += answer[i];

    }

    return  n > 0 ? sum / n : -1;

}

/\* 函数功能：计算n个数的中位数 \*/

int Median(int answer[], int n)

{

    DataSort(answer, n);

    if (n % 2 == 0)

        return  (answer[n / 2] + answer[n / 2 - 1]) / 2;

    else

        return  answer[n / 2];

}

/\* 函数功能：计算n个数的众数 \*/

int Mode(int answer[], int n)

{

    int  i, grade, max = 0, modeValue = 0, count[N] = {0};

    for (i = 0; i < n; i++)

    {

        count[answer[i]]++;

    }

    for (grade = 1; grade <= N - 1; grade++)

    {

        if (count[grade] > max)

        {

            max = count[grade];

            modeValue = grade;

        }

    }

    return modeValue;

}

/\*  函数功能：按选择法对数组a中的n个元素进行排序 \*/

void DataSort(int a[], int n)

{

    int i, j, k, temp;

    for (i = 0; i < n - 1; i++)

    {

        k = i;

        for (j = i + 1; j < n; j++)

        {

            if (a[j] > a[k]) k = j;

        }

        if (k != i)

        {

            temp = a[k];

            a[k] = a[i];

            a[i] = temp;

        }

    }

}

五十二。

程序改错。

以下程序的功能是统计字符数。判断一个由’0’ ~ ‘9’这10个字符组成的字符串中哪个字符出现的次数最多。

输入数据：第一行是测试数据的组数m，每组测试数据占1行，每行数据不超过1000个字符且非空。

输出要求：m行，每行对应一组输入，包括出现次数最多的字符和该字符出现的次数。如果有多个字符出现的次数相同且最多，那么输出ASCII码最小的那一个。

|  |
| --- |
| #include <stdio.h>  #include <string.h>  main( )  {  int  cases, sum[10], i, max;    char str[1000];      scanf("%d", case);    while (cases > 0)         {                  scanf("%c", str);              for( i = 0; i < 10; i++)                   sum[i] = 0;                      for(i < 0; i < strlen(str); i++)                  ++sum[str[i] – 0];              max = 0;              for (i = 1; i < 10; i++)                  if(sum[i] >＝ sum[max]) max = i;              printf("%c %d\n", max + '0', sum[0]);              cases --;         }  } |

五十二。

#include <stdio.h>

#include <string.h>

main( )

{

    int  cases, sum[10], i, max;

    char str[1001];

    scanf("%d", &cases);

    while (cases > 0)

    {

        scanf("%s", str);

        for( i = 0; i < 10; i++)

            sum[i] = 0;

        for(i = 0; i < strlen(str); i++)

            ++sum[str[i] - '0'];

        max = 0;

        for (i = 1; i < 10; i++)

             if(sum[i] > sum[max]) max = i;

        printf("%c %d\n", max + '0', sum[max]);

        cases --;

    }

}

五十三。

从一个字符串删除元音字母，例如，如果输入的字符串是“C programming”，那么输出将是“C prgrmmng”。

\*\*输入格式要求：提示信息："Enter a string to delete vowels\n"

\*\*输出格式要求："String after deleting vowels: %s\n"

程序运行示例如下：

Enter a string to delete vowels

c programming

String after deleting vowels: c prgrmmng

五十三。

#include <stdio.h>

#include <string.h>

int check\_vowel(char);

int main()

{

    char s[100], t[100];

    int i, j = 0;

    printf("Enter a string to delete vowels\n");

    gets(s);

    for (i = 0; s[i] != '\0'; i++)

    {

        if (check\_vowel(s[i]) == 0)        //not a vowel

        {

            t[j] = s[i];

            j++;

        }

    }

    t[j] = '\0';

    strcpy(s, t);    //We are changing initial string

    printf("String after deleting vowels: %s\n", s);

    return 0;

}

int check\_vowel(char c)

{

    switch (c)

    {

    case 'a':

    case 'A':

    case 'e':

    case 'E':

    case 'i':

    case 'I':

    case 'o':

    case 'O':

    case 'u':

    case 'U':

        return 1;

    default:

        return 0;

    }

}

五十四。

冒泡排序法。设有N个杂乱无序的整数，要求将这N个整数从小到大排序后输出。

\*\*输入格式要求："%d" 提示信息："Enter No.%2d:"

\*\*输出格式要求："%d"

程序运行示例如下：

Enter No. 1:5

Enter No. 2:7

Enter No. 3:3

Enter No. 4:9

Enter No. 5:8

35789

五十四。

#include <stdio.h>

#define N 5

main()

{int i,j,t,a[N];

 for(i=0;i<N;i++)

 { printf("Enter No.%2d:",i+1);

   scanf("%d",&a[i]);

 }

 for(i=0;i<N-1;i++)

   for(j=0;j<N-i-1;j++)

     if(a[j]>a[j+1])

     { t=a[j];

       a[j]=a[j+1];

       a[j+1]=t;

     }

 for(i=0;i<N;i++)

   printf("%d",a[i]);

}

五十五。

程序：数列反向。

要求用户录入一串数（假设为10个整数），然后按反向顺序输出这些数：

Enter 10 numbers: 34 82 49 102 7 94 23 11 50 31

In reverse order: 31 50 11 23 94 7 102 49 82 34

\*\*输入格式要求："%d" 提示信息："Enter %d numbers:"

\*\*输出格式要求："In reverse order:" " %d"

五十五。

#include <stdio.h>

#define N 10

int main()

{

    int a[N], i;

    printf("Enter %d numbers:", N);

    for (i = 0; i < N; i++)

        scanf("%d", &a[i]);

    printf("In reverse order:");

    for (i = N - 1; i >= 0; i--)

        printf(" %d", a[i]);

    printf("\n");

    return 0;

}

五十六。

创建并输出一个一维数组（含20个元素），数组元素的值分别是下标的3倍多2。

\*\*输出格式要求："%3d"

程序的运行示例如下：

2 5 8 11 14 17 20 23 26 29 32 35 38 41 44 47 50 53 56 59

五十六。

#include <stdio.h>

main()

{

    int a[20], k;

    for (k = 0; k < 20; k++)

    {

        a[k] = 3 \* k + 2;

        printf("%3d", a[k]);

    }

}

1.(30分)编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void main()

{

int n;

char str[N], min[N];

printf("Input five countries' names:\n");

gets(str);//2

strcpy(min, str); //2

for (n=1; n<5; n++)//3

{

gets(str); //2

if (strcmp(str, min) < 0) //4

{

strcpy(min, str); //2

}

}

printf("The minimum is:%s\n", min);//2

}

2.(10分)用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

#include <stdio.h>

#include <string.h>

#define N 100

void Squeeze(char \*s, char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char \*s, char c)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; t++)

{

if (\*t != c)

{

\*s = \*t;

s++;

}

}

\*s = '\0'; /\* 在字符串t2的末尾添加字符串结束标志 \*/

}

3.(10分)用字符数组作函数参数编程实现在字符串每个字符间插入一个空格的功能。

\*\*输入提示信息："Input a string:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Insert results:%s\n"

程序运行示例1如下：

Input a string:Howareyou

Insert results:H o w a r e y o u

#include <stdio.h>

#include <string.h>

#define N 100

void Insert(char s[]);

main()

{

char str[N];

printf("Input a string:");

gets(str);

Insert(str);

printf("Insert results:%s\n", str);

}

void Insert(char s[])

{

char t[N];

int i, j;

strcpy(t, s);

for (i = 0, j = 0; t[i] != '\0'; i++, j++)

{

s[j] = t[i];

j++;

s[j] = ' ';

}

s[j] = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

4.(15分)从键盘任意输入一个字符串，计算其实际字符个数并打印输出，即不使用字符串处理函数strlen(),编程实现strlen()的功能。

\*\*输入格式要求："%s" 提示信息："Please enter a string:"

\*\*输出格式要求："The length of the string is: %u\n"

程序的运行示例如下：

Please enter a string:Hello China

The length of the string is: 11

#include <stdio.h>

unsigned int MyStrlen(char str[]);

/\* 函数功能：用字符型数组作函数参数，计算字符串的长度 \*/

unsigned int MyStrlen(char str[])

{

int i ;

unsigned int len = 0; /\* 计数器置0 \*/

for (i = 0; str[i] != '\0'; i++)

{

len++; /\* 利用循环统计不包括’\0’在内的字符个数 \*/

}

return (len); /\* 返回实际字符个数 \*/

}

int main()

{

char a[80];

unsigned int len;

printf("Please enter a string:");

gets(a);

len = MyStrlen(a); /\* 计算字符串实际字符个数 \*/

printf("The length of the string is: %u\n", len);

return 0;

}

5.(10分)按如下函数原型编写程序，用字符数组作函数参数编程实现字符串逆序存放功能。

void Inverse(char str[], char ptr[]);

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[], char ptr[]);

int main()

{

char a[N], b[N];

printf("Input a string:");

gets(a);

Inverse(a, b);

printf("Inversed results:%s\n", b);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char str[], char ptr[])

{

int len, i, j;

len = strlen(str);

for (i = 0, j = len - 1; str[i] != '\0'; i++, j--)

{

ptr[j] = str[i];

}

ptr[i] = '\0'; /\* 在字符串ptr的末尾添加字符串结束标志 \*/

}

6.(10分)任意输入英文的星期几，通过查找星期表，输出其对应的数字，

若查到表尾，仍未找到，则输出错误提示信息。

\*\*输入格式要求："%s" 提示信息："Please enter a string:\n"

\*\*输出格式要求："%s is %d\n" "Not found!\n"

查找表中信息：

"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"

程序运行示例1如下：

Please enter a string:

Monday

Monday is 1

程序运行示例2如下：

Monkey

Not found!

#include <stdio.h>

#include <string.h>

#define WEEKDAYS 7

#define MAX\_STR\_LEN 10

int main()

{

int i, pos;

int findFlag = 0;

char x[MAX\_STR\_LEN];

char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};

printf("Please enter a string:\n");

scanf("%s", x);

for (i = 0; i < WEEKDAYS && !findFlag; i++)

{

if (strcmp(x, weekDay[i]) == 0)

{

pos = i;

findFlag = 1;

}

}

if (findFlag)

{

printf("%s is %d\n", x, pos);

}

else

{

printf("Not found!\n");

}

return 0;

}

7.(10分)请编程实现按奥运会参赛国国名在字典中的顺序对其入场次序进行排序。假设参赛国不超过150个。要求通过二维字符数组传参调用函数实现排序,输入和输出在主函数中实现。

\*\*输入格式要求："%d" 提示信息："How many countries?" "Input their names:\n"

\*\*输出格式要求："Sorted results:\n"

程序的运行示例如下：

How many countries? 5

Input their names:

America

England

Australia

Sweden

Finland

Sorted results:

America

Australia

England

Finland

Sweden

#include <stdio.h>

#include <string.h>

#define MAX\_LEN 10 /\* 字符串最大长度 \*/

#define N 150 /\* 字符串个数 \*/

void SortString(char str[][MAX\_LEN], int n);

int main()

{

int i, n;

char name[N][MAX\_LEN]; /\* 定义二维字符数组 \*/

printf("How many countries?");

scanf("%d", &n);

getchar(); /\* 读走输入缓冲区中的回车符 \*/

printf("Input their names:\n");

for (i = 0; i < n; i++)

{

gets(name[i]); /\* 输入n个字符串 \*/

}

SortString(name, n); /\* 字符串按字典顺序排序 \*/

printf("Sorted results:\n");

for (i = 0; i < n; i++)

{

puts(name[i]); /\* 输出排序后的n个字符串 \*/

}

return 0;

}

/\* 函数功能：交换法实现字符串按字典顺序排序 \*/

void SortString(char str[][MAX\_LEN], int n)

{

int i, j;

char temp[MAX\_LEN];

for (i = 0; i < n - 1; i++)

{

for (j = i + 1; j < n; j++)

{

if (strcmp(str[j], str[i]) < 0)

{

strcpy(temp, str[i]);

strcpy(str[i], str[j]);

strcpy(str[j], temp);

}

}

}

}

8.(10分)用字符指针作函数参数编程实现字符串逆序存放功能。

输入提示信息："Input a string:"

输入字符串用gets()

输出提示信息："Inversed results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char \*pStr);

int main()

{

char a[N];

printf("Input a string:");

gets(a);

Inverse(a);

printf("Inversed results:%s\n", a);

return 0;

}

/\* 函数功能： 实现将字符数组中的字符串逆序存放 \*/

void Inverse(char \*pStr)

{

int len;

char temp;

char \*pStart; /\* 指针变量pStart指向字符串的第一个字符 \*/

char \*pEnd; /\* 指针变量pEnd指向字符串的最后一个字符 \*/

len = strlen(pStr); /\* 求出字符串长度 \*/

for (pStart = pStr, pEnd = pStr + len - 1; pStart < pEnd; pStart++, pEnd--)

{

temp = \*pStart;

\*pStart = \*pEnd;

\*pEnd = temp;

}

}

9.(30分)编程实现如下功能：在字符串中删除与某字符相同的字符。

要求输入和输出在主函数中实现，

在字符串中删除与某字符相同的字符用函数实现

\*\*提示信息：

"Input a string:"

"Input a character:"

\*\*输入格式要求："%s"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

#include <stdio.h>

void Squeeze(char s[], char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char s[], char c)

{

int i, j;

for (i = j = 0; s[i] != '\0'; i++)

{

if (s[i] != c)

{

s[j] = s[i];

j++;

}

}

s[j] = '\0'; /\* 在字符串s的末尾添加字符串结束标志 \*/

}

10.(10分)不使用字符串处理函数strcat()编程实现strcat()的功能，即任意输入两个字符串，然后连接这两个字符串，返回连接后字符串的首地址。

程序的运行示例如下：

Input the first string:Hello

Input the second string:China

The result is: HelloChina

#include <stdio.h>

#define N 80

char \*MyStrcat(char \*dstStr, char \*srcStr);

int main()

{

char first[2 \* N]; /\* 这个数组应该足够大，以便存放连接后的字符串 \*/

char second[N];

char \*result = NULL;

printf("Input the first string:");

gets(first);

printf("Input the second string:");

gets(second);

result = MyStrcat(first, second);

printf("The result is: %s\n", result);

return 0;

}

/\* 函数功能：连接两个字符串 \*/

char \*MyStrcat(char \*dstStr, char \*srcStr)

{

char \*pStr = dstStr; /\* 保存字符串首地址指针 \*/

/\* 将指针移到字符串末尾 \*/

while (\*dstStr != '\0')

{

dstStr++;

}

/\* 将字符串2内容复制到字符串1的后面 \*/

for (; \*srcStr != '\0'; dstStr++, srcStr++)

{

\*dstStr = \*srcStr;

}

\*dstStr = '\0'; /\* 在连接后的字符串的末尾添加字符串结束标志 \*/

return pStr; /\* 返回连接后的字符串的指针 \*/

}

11.(10分)从键盘输入一个字符串a，将字符串a复制到字符串b中，再输出字符串b，即编程实现字符串处理函数strcpy()的功能，但要求不能使用字符串处理函数strcpy()。

\*\*输入格式要求："%s" 提示信息："Input a string:"

\*\*输出格式要求："The copy is:"

程序的运行示例如下：

Input a string:Hello China

The copy is:Hello China

#include <stdio.h>

#define N 80

void MyStrcpy(char dstStr[], char srcStr[]);

int main()

{

char a[N], b[N];

printf("Input a string:");

gets(a); /\* 输入字符串 \*/

MyStrcpy(b, a); /\* 将字符数组a中的字符串拷贝到b中 \*/

printf("The copy is:");

puts(b); /\* 输出复制后的字符串 \*/

return 0;

}

/\* 函数功能：用字符数组作为函数参数，字符串拷贝 \*/

void MyStrcpy(char dstStr[], char srcStr[])

{

int i = 0; /\* 数组下标初始化为0 \*/

while (srcStr[i] != '\0') /\* 若当前取出的字符不是字符串结束标志 \*/

{

dstStr[i] = srcStr[i];/\* 复制字符 \*/

i++; /\* 移动下标 \*/

}

dstStr[i] = '\0'; /\* 在字符串dstStr的末尾添加字符串结束标志 \*/

}

12.(35分)按如下函数原型编程实现字符串连接函数strcat()的功能。

void MyStrcat(char \*dstStr, char \*srcStr);

将srcStr字符串连接到dstStr字符串后面，；

输入提示信息：

"Input a string:"

"Input another string:"

输入字符串用gets()

输出提示信息和格式："Concatenate results:%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void MyStrcat(char \*dstStr, char \*srcStr);

int main()

{

char s[N], t[N];

printf("Input a string:");

gets(s);

printf("Input another string:");

gets(t);

MyStrcat(s, t);

printf("Concatenate results:%s\n", s);

return 0;

}

/\* 函数功能：将字符串srcStr连接到字符串dstStr之后 \*/

void MyStrcat(char \*dstStr, char \* srcStr)

{

while (\*dstStr != '\0')

{

dstStr++;

}

while (\*srcStr != '\0')/\* 若srcStr所指字符不是字符串结束标志 \*/

{

\*dstStr = \*srcStr; /\* 将srcStr所指字符复制到dstStr所指内存中\*/

srcStr++; /\* 使srcStr指向下一个字符 \*/

dstStr++; /\* 使dstStr指向下一个存储单元 \*/

}

\*dstStr = '\0'; /\* 在字符串dstStr的末尾添加字符串结束标志 \*/

}

13.(10分)用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*输入格式要求："%s"

输入提示信息：

"Input a string:"

"Input a character:"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

#include <stdio.h>

#include <string.h>

#define N 100

void Squeeze(char \*s, char c);

int main()

{

char str[20], ch;

printf("Input a string:");

gets(str);

printf("Input a character:");

ch = getchar();

Squeeze(str, ch);

printf("Results:%s\n", str);

return 0;

}

void Squeeze(char \*s, char c)

{

char str[N];

char \*t = str;

strcpy(t, s);

for (; \*t != '\0'; t++)

{

if (\*t != c)

{

\*s = \*t;

s++;

}

}

\*s = '\0'; /\* 在字符串t2的末尾添加字符串结束标志 \*/

}

一．

输入一行字符，统计其中的英文字符、数字字符、空格字符，以及其他字符的个数。请找出以下程序的错误，并改正之。

#include <stdio.h>

#include <string.h>

#define ARR\_SIZE = 80;

main()

{

char str[ARR\_SIZE];

int len, i;

int letter=0,digit=0,space=0,other=0;

printf("请输入一个字符串:");

gets(str);

len = strlen(str);

for (i=0; i<len; i++)

{

if (a=<str[i]<=z || A=<str[i]<=Z)

{

letter ++;

}

else if (0=<str[i]<=9)

{

digit ++;

}

else if (str[i]=' ' )

{

space ++;

}

else

other ++;

}

printf("英文字符数:%d\n", letter);

printf("数字字符数:%d\n", digit);

printf("空格数:%d\n", space);

printf("其他字符数:%d\n", other);

}

参考答案

|  |
| --- |
| #include <stdio.h>  #include <string.h>    #define ARR\_SIZE  80  main()  {      char str[ARR\_SIZE];      int  len, i;      int  letter=0,digit=0,space=0,other=0;        printf("请输入一个字符串:");      gets(str);        len = strlen(str);        for (i=0; i<len; i++)      {          if (str[i]>='a'&&str[i]<='z' || str[i]>='A'&&str[i]<='Z')          {              letter ++;          }          else if (str[i]>='0'&&str[i]<='9')          {              digit ++;          }          else if (str[i]==' ' )          {              space ++;          }          else              other ++;      }        printf("英文字符数:%d\n", letter);      printf("数字字符数:%d\n", digit);      printf("空格数:%d\n", space);      printf("其他字符数:%d\n", other);  } |

二．

编写程序实现将字符串str2拷贝到字符串str1。

\*\*输入格式要求："%s" 提示信息："Enter string 2:"

\*\*输出格式要求："Output string 1:%s\n"

程序运行的输入输出样例：

输入为

Enter string 2:good morning

输出为

Output string 1:good morning

参考答案：

#include <stdio.h>

main()

{

    char str1[100],str2[100];

    int i;

    printf("Enter string 2:");

    gets(str2);

    for(i=0;(str1[i]=str2[i])!='\0';i++)

        ;

    printf("Output string 1:%s\n",str1);

}

三．

简单的口令检查程序

按下述要求编写口令检查程序（假设正确的口令为8888）。

（1）若输入口令正确，则提示“You are welcome!”，程序结束。

（2）若输入口令不正确，则提示“Wrong password!”，同时检查口令是否已输入3次，若未输入3次，则提示“Enter again：”，且允许用户再次输入口令；若已输入3次，则提示“You have entered three times! You are not welcome!”，且不允许用户再输入口令，程序结束。

参考答案：

#include  <stdio.h>

#include  <string.h>

#define MAXLEN  20

int main(void)

{

    int  n, flag = 0;

    char str[MAXLEN];

    printf("Please enter password:");

    gets(str);

    n = 1;

    while ((n <= 3) && (flag == 0))

    {

        if (strcmp(str, "8888") == 0)           //实际口令为8888

        {

            printf("You are welcome!\n");

            flag = 1;

        }

        else

        {

            printf("Wrong password!\n");

            if (n < 3)

            {

                printf("Please enter again:");

                gets(str);

                n++;

            }

            else

            {

                printf("You have entered three times!\n");

                printf("You are not welcome!\n");

                flag = 1;

            }

        }

    }

    return 0;

}

四．

任意输入英文的星期几，通过查找星期表，输出其对应的数字，

若查到表尾，仍未找到，则输出错误提示信息。

\*\*输入格式要求："%s" 提示信息："Please enter a string:\n"

\*\*输出格式要求："%s is %d\n" "Not found!\n"

查找表中信息：

"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"

程序运行示例1如下：

Please enter a string:

Monday

Monday is 1

程序运行示例2如下：

Monkey

Not found!

参考答案：

#include <stdio.h>

#include <string.h>

#define WEEKDAYS 7

#define MAX\_STR\_LEN 10

int main()

{

    int i, pos;

    int findFlag = 0;

    char x[MAX\_STR\_LEN];

    char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};

    printf("Please enter a string:\n");

    scanf("%s", x);

    for (i = 0; i < WEEKDAYS && !findFlag; i++)

    {

        if (strcmp(x, weekDay[i]) == 0)

        {

            pos = i;

            findFlag = 1;

        }

    }

    if (findFlag)

    {

        printf("%s is %d\n", x, pos);

    }

    else

    {

        printf("Not found!\n");

    }

    return 0;

}

五．

编程实现找出字符串中最大字符元素并输出该元素及其对应的ASCII值.

\*\*\*\*要求输入提示信息为：

"Input a string:\n"

\*\*\*\*输出格式要求为：

"The largest character of \"%s\" is \'%c\' ,The ASCII is %d."

参考答案：

#include <stdio.h>

#include <stdlib.h>

main()

{

    char string[81];

    int i=0,max;

    printf("Input a string:");  //1

    printf("\n");

    gets(string);

    max=string[i];      //1

    while(string[i]!='\0')      //2

    {

        i=i+1;

        if(string[i]>max)        //2

           max=string[i];       //1

    }

    printf("The largest character of \"%s\" is \'%c\' ,The ASCII is %d.",string,max,max);   //2

}

六．

有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。

程序的运行示例如下：

请输入一个字符串：1234567890

从第几个字符开始复制？3

复制结果为：34567890

参考答案：

#include <stdio.h>

copystr(char \*p1, char \*p2, int m)

{

    int n;

    n = 0;

    while (n < m - 1)

    {

        n++;

        p1++;

    }

    while (\*p1 != '\0')

    {

        \*p2 = \*p1;

        p1++;

        p2++;

    }

    \*p2 = '\0';

}

int main(void)

{

    int m;

    char str1[20], str2[20];

    printf("请输入一个字符串：\n");

    scanf("%s", str1);

    printf("从第几个字符开始复制？");

    scanf("%d", &m);

    if (strlen(str1) < m)

    {

        printf("输入错误。");

    }

    else

    {

        copystr(str1, str2, m);

        printf("复制结果为：%s", str2);

    }

    return 0;

}

七．

编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

参考答案：

#include  <stdio.h>

#include  <string.h>

#define N 80

void main()

{

    int    n;

    char  str[N], min[N];

    printf("Input five countries' names:\n");

    gets(str);//2

    strcpy(min, str);  //2

    for (n=1; n<5; n++)//3

    {

        gets(str); //2

        if (strcmp(str, min) < 0)    //4

        {

            strcpy(min, str);  //2

        }

    }

    printf("The minimum is:%s\n", min);//2

}

八．

下面的程序将数字进行加密，其中的每一个数字转化为另一个不同的数字。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

char s[] = "24635", c, i;

for (i = 0; c = s[i]; i++)

{

switch ((c - '0')

{

case 2:

case 3:

putchar(c + 4);

continue;

case 4:

putchar(c + 4);

break;

case 5:

putchar(c + 3);

default:

putchar(c + 2);

}

putchar('\n');

}

}

参考答案：

#include <stdio.h>

main()

{

    char s[] = "24635", c, i;

    for (i = 0; c = s[i]; i++)

    {

        switch (c - '0')

        {

        case 2:

        case 3:

            putchar(c + 4);

            continue;

        case 4:

            putchar(c + 4);

            break;

        case 5:

            putchar(c + 3);

            break;

        default:

            putchar(c + 2);

        }

        putchar('\n');

    }

}

九．

输入6个字符串，输出最小串及最大串。

\*\*输入格式要求："%s" 提示信息："请输入6行字符串：\n"

\*\*输出格式要求："The max string is: %s\n" "The min string is: %s\n"

程序示例运行如下：

请输入6行字符串：

hello,world

vb

vc

Java

c++

c#

The max string is: vc

The min string is: Java

参考答案：

#include <stdio.h>

#include <string.h>

#define N 6

#define N1 20

int main()

{

    char str[N1], min[N1], max[N1];

    int i;

    printf("请输入6行字符串：\n");

    gets(min);  //假设第1个串最小

    strcpy(max, min);//假设最大串的值也为min

    //循环输入其它串，并与最小的串及最大的串比较

    for (i = 2; i <= N; i++)

    {

        gets(str);

        if (strcmp(str, min) < 0)

            strcpy(min, str);

        if (strcmp(str, max) > 0)

            strcpy(max, str);

    }

    printf("The max string is: %s\n", max);

    printf("The min string is: %s\n", min);

    return 0;

}

十．

将字符串s1从第m个字符开始剩余的所有字符，送入字符数组s2中。

\*\*输入格式要求："%d" 提示信息："input a string:\n" "input start point:\n"

程序运行示例如下：

input a string:hello,world!

input start point:7

world!

参考答案：

#include <stdio.h>

main()

{

    int i, j, m;

    char s1[80], s2[80];

    printf("input a string:\n");

    gets(s1);

    printf("input start point:\n");

    scanf("%d", &m);

    i = m - 1;

    j = 0;

    while (s1[i] != '\0')

    {

        s2[j] = s1[i];

        i++;

        j++;

    }

    s2[j] = '\0';

    puts(s2);

}

十一。

指针和字符数组的应用,用五种方法输出字符串“China”。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

int i = 0;

char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[5];

while (str[i] != '\0')

{

putchar(\*str);

str++;

}

scanf("%s", ptr);

puts(ptr);

ptr1 = str;

puts(ptr1);

ptr2 = "China";

puts(ptr2);

str2 = "China";

printf("%s", str2);

}

参考答案：

#include <stdio.h>

main()

{

    int i = 0;

    char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[6];

    while (str[i] != '\0')

    {

        putchar(str[i]);

        i++;

    }

    ptr = str1;

    scanf("%s", ptr);

    puts(ptr);

    ptr1 = str;

    puts(ptr1);

    ptr2 = "China";

    puts(ptr2);

    strcpy(str2, "China");

    printf("%s", str2);

}

十二。

将字符串“Hello World”读入字符数组中去。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

#include <string.h>

main()

{

char ch[10];

scanf("%s", ch);

if (!strcmp(ch, "Hello World"))

printf("OK!");

else

printf("Error!");

}

参考答案：

#include <stdio.h>

#include <string.h>

main()

{

    char ch[10];

    gets(ch);

    if (!strcmp(ch, "Hello World"))

        printf("OK!");

    else

        printf("Error!");

}

十三。

)输入英文的星期几，查找星期表后输出对应的数字。具体的算法是：若发现与表中某项相同，则输出该字符串在表中的位置（序号）；若查到表尾仍不同，则输出错误信息。设星期表的内容如下图所示。

0 Sunday

1 Monday

2 Tuesday

3 Wednesday

4 Thursday

5 Friday

6 Saturday

\*\*输入格式要求："%s" 提示信息："Enter a string:"

\*\*输出格式要求："%s is %d\n" "Error\n"

程序运行示例如下：

Enter a string:wednesday

wednesday is 3

参考答案：

#include <stdio.h>

char week\_day[][10]={"sunday","monday","tuesday","wednesday","thursday","friday","saturday"};

main()

{

    int i,j;

    char id[10],c;

    printf("Enter a string:");

    scanf("%s",id);

    for(i=0;i<7;i++)

    {

        for(j=0;(c=week\_day[i][j])==id[j]&&c!='\0';j++)

            ;

        if(c=='\0')

            break;

    }

    if(i<7)

        printf("%s is %d\n",id,i);

    else

        printf("Error\n");

}

十四。

编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

参考答案：

#include  <stdio.h>

#include  <string.h>

#define N 80

void main()

{

    int    n;

    char  str[N], min[N];

    printf("Input five countries' names:\n");

    gets(str);//2

    strcpy(min, str);  //2

    for (n=1; n<5; n++)//3

    {

        gets(str); //2

        if (strcmp(str, min) < 0)    //4

        {

            strcpy(min, str);  //2

        }

    }

    printf("The minimum is:%s\n", min);//2

}

十五。

编写一个程序，将一个字符串s2插入到字符串s1中，其起始插入位置为n。

\*\*输入格式要求："%d" 提示信息："main string:" "sub string:" "site of begining:(<=%d)"

\*\*输出格式要求："After instert:%s\n"

程序运行示例如下：

main string:hello,world!

sub string:c

site of begining:(<=12)6

After instert:hello,c world!

参考答案：

#include <stdio.h>

#include <string.h>

#define N 100

int main()

{

    int n, i, j, k, len;

    char s1[N], s2[N], s3[2 \* N];

    printf("main string:");

    gets(s1);

    printf("sub string:");

    gets(s2);

    len = strlen(s1);

    do

    {

        printf("site of begining:(<=%d)", len);

        scanf("%d", &n);

    }

    while (n > len);

    for (i = 0; i < n; i++)

    {

        s3[i] = s1[i];

    }

    for (j = 0; s2[j] != '\0'; j++)

    {

        s3[i + j] = s2[j];

    }

    for (k = n; s1[k] != '\0'; k++)

    {

        s3[j + k] = s1[k];

    }

    s3[j + k] = '\0';

    printf("After instert:%s\n", s3);

    return 0;

}

1.(10分)题目：去除字符串中的空格并分解单词。

要求：在main()中调用下述各函数，完成字符串中去除空格并分解单词的功能。

函数原型如下：

函数一：单词内是否包含数字 int IsNumIn(char word[])

函数二：单词内是否包含空格 int IsSpcIn(char word[])

函数三：去掉单词的前后空格，tab键和换行符 Trim(char oldWord[], char newWord[])

函数四：单词内部有空格，分解成多个单词 Seg(char words[], char wArray[][100] ) 假设单词内部只有一个空格，没有两个连续空格的情况发生。

要求利用gets函数来获得用户输入的字符串，不能用scanf来获得用户输入的字符串

参考输入，输出

参考输入

ros2e

参考输出

error

参考输入

︺︺hello︺（ ︺代表一个空格）

参考输出

hello

参考输入

︺︺hello world︺

参考输出

hello

world

#include <stdio.h>

#include <ctype.h>

#define N 100

int IsNumIn(char word[]);

int IsSpcIn(char word[]);

void Trim(char oldWord[], char newWord[]);

void Seg(char words[], char wArray[][100] );

int main()

{

char str1[N],str[N],w\_Arr[20][100];

int i,j;

for(i=0;i<N;i++)

{

\*(str1+i)=0;

\*(str+i)=0;

}

for(i=0;i<20;i++)

{

for(j=0;j<100;j++)

w\_Arr[i][j]=0;

}

gets(str1);

Trim(str1,str);

if(IsNumIn(str))

printf("error");

else

{

if(IsSpcIn(str))

{

Seg(str,w\_Arr);

for(i=0;w\_Arr[i][0]!='\0';i++)

{

puts(w\_Arr[i]);

}

}

else

puts(str);

}

return 0;

}

int IsNumIn(char word[])

{

int i;

for(i=0;\*(word+i)!='\0';i++)

{

if(isdigit(\*(word+i)))

return 1;

}

return 0;

}

int IsSpcIn(char word[])

{

int i;

for(i=0;\*(word+i)!='\0';i++)

{

if(isspace(\*(word+i)))

return 1;

}

return 0;

}

void Trim(char oldWord[], char newWord[])

{

int i=0,j=0;

while(isspace(\*(oldWord+i)))

i++;

while( \*(oldWord+i) != '\0')

{

\*(newWord+j)=\*(oldWord+i);

i++;

j++;

}

i--;

j--;

for(;isspace(\*(oldWord+i));i--,j--)

{

\*(newWord+j)=0;

}

}

void Seg(char words[], char wArray[][100] )

{

int i,j=0,k=0;

for(i=0;i<N;i++)

{

if(isalpha(\*(words+i)))

{

wArray[j][k]=\*(words+i);

k++;

}

else

{

j++;

k=0;

}

}

}

2.(20分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能。

要求：

（1）按如下函数原型进行编程：

int Mystrlen(char str[]);

（2）在主函数中：

任意输入一个字符串（长度不超过80个字符）;

调用函数Mystrlen()计算输入字符串的实际长度;

然后打印计算结果。

\*\*要求输入提示信息为："Please input the string:\n"

\*\*要求输出格式为:"%d\n"

注意：

（1）字符串中的空格计入字符串实际长度，

但是字符串结束标志不计入字符串实际长度。

（2）必须使用函数编程。否则没有成绩。

#include <stdio.h>

int Mystrlen(char str[]); //2

main()

{

char str[80];

int len;

printf ("Please input the string:\n"); //2

gets(str); //2

len = Mystrlen(str); //2

printf("%d\n", len); //2

}

int Mystrlen(char str[])

{

int len = 0, i;

for (i=0; str[i]!=0; i++) //2

{

len++; //2

}

return len; //2

}

3.(5分)调用GetInput()函数从键盘任意输入一个字符串，然后分别在GetInput()和main()中输出该字符串。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，

修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

|  |
| --- |
| #include  <stdio.h>  void GetInput(char\* s);    int main(void)  {     char \*ptr = NULL;     GetInput(ptr);     puts(ptr);     return 0;  }    void GetInput(char\* s)  {      scanf("%s", s);      puts(s);  } |

#include <stdio.h>

void GetInput(char\* s);

int main(void)

{

char str[80];

char \*ptr = str;

GetInput(ptr);

puts(ptr);

return 0;

}

void GetInput(char\* s)

{

scanf("%s", s);

puts(s);

}

4.(10分)以下程序实现从键盘输入一个字符串，将其字符顺序颠倒后重新存放，并输出这个字符串。（用字符数组实现）。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  #include <string.h>  void Inverse(char rstr[])    main()  {      char  str[80];        printf("Input a string:\n");      gets(str);      Inverse(str);      printf("The inversed string is:\n");      puts(str);  }    void Inverse(char rstr[])  {      int i,n;      char temp;        for(i=0, n=(strlen(rstr)); i<n; i++, n--)      {          temp = rstr[i];          rstr[i] = rstr[n];          rstr[n] = temp;      }  } |

#include <stdio.h>

#include <string.h>

void Inverse(char rstr[]);

main()

{

char str[80];

printf("Input a string:\n");

gets(str);

Inverse(str);

printf("The inversed string is:\n");

puts(str);

}

void Inverse(char rstr[])

{

int i,n;

char temp;

for(i=0, n=(strlen(rstr)-1); i<n; i++,n--)

{

temp = rstr[i];

rstr[i] = rstr[n];

rstr[n] = temp;

}

}

5.(10分)有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。

程序的运行示例如下：

请输入一个字符串：1234567890

从第几个字符开始复制？3

复制结果为：34567890

#include <stdio.h>

copystr(char \*p1, char \*p2, int m)

{

int n;

n = 0;

while (n < m - 1)

{

n++;

p1++;

}

while (\*p1 != '\0')

{

\*p2 = \*p1;

p1++;

p2++;

}

\*p2 = '\0';

}

int main(void)

{

int m;

char str1[20], str2[20];

printf("请输入一个字符串：\n");

scanf("%s", str1);

printf("从第几个字符开始复制？");

scanf("%d", &m);

if (strlen(str1) < m)

{

printf("输入错误。");

}

else

{

copystr(str1, str2, m);

printf("复制结果为：%s", str2);

}

return 0;

}

6.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q108.(15分)用字符数组作函数参数编程，

利用一个数组实现字符串（允许输入带空格的字符串）的逆序存放。

要求如下：

（1）在主函数中从键盘输入字符串，字符串的最大长度为80个字符

调用Inverse函数，并输出逆序后的字符串。

（2）在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(3)\*\*输入提示信息格式要求："Input a string:\n"

\*\*输出提示信息要求："Inversed results:\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);//1

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])//1

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

Q139.(15分)请按给定的函数原型编程实现将字符数组中的字符串的第m个字符开始的n个字符逆序存放。要求在主函数读入字符串，且逆序存放后的字符串也在主函数打印。函数原型：

void inverse(char str[],int m,int n);

输入要求: 输入的一行字符串，应包含字母，数字以及空格字符

友情提示:

在执行输入字符串的函数之前,请用getchar();把输入缓冲区中的换行符读出!!!

\*\*\*\*输入提示信息和格式要求为：

"input m,n:"

"%d,%d"

"input the string:"

\*\*\*\*输出提示信息为:"the inverse string:"

#include <stdio.h>

#define N 100

void inverse(char str[],int m,int n);

main()

{

char str[N];

int m,n;

printf("input m,n:"); //1

scanf("%d,%d",&m,&n);

getchar();

printf("input the string:"); //1

gets(str);

inverse(str,m,n); //2

printf("the inverse string:%s",str); //1

}

void inverse(char str[],int m,int n)

{

int i;

char a[N];

for(i=0;i<n;i++)

{

a[i]=str[m-1+i]; //1

}

for(i=0;i<n;i++)

{

str[m+i-1]=a[n-1-i]; //1

}

}

Q1323.(10分)写一函数，输入一个十六进制数，输出相应的十进制数。

程序的运行示例如下：

输入十六进制数：

3a

十进制数58

继续吗？

y

输入十六进制数：

2f

十进制数47

继续吗？

N

#include <stdio.h>

#define MAX 1000

int main()

{

int c, i, flag, flag1;

char t[MAX];

i = 0;

flag = 0;

flag1 = 1;

printf("\n输入十六进制数：");

while ((c = getchar()) != '\0' && i < MAX && flag1)

{

if (c >= '0' && c <= '9' || c >= 'a' && c <= 'f' || c >= 'A' && c <= 'F')

{

flag = 1;

t[i++] = c;

}

else if (flag)

{

t[i] = '\0';

printf("\n十进制数%d\n", htoi(t));

printf("继续吗？");

c = getchar();

if (c == 'N' || c == 'n')

{

flag1 = 0;

}

else

{

flag = 0;

i = 0;

printf("\n输入十六进制数：");

}

}

}

return 0;

}

htoi(char s[])

{

int i, n;

n = 0;

for (i = 0; s[i] != '\0'; i++)

{

if (s[i] >= '0' && s[i] <= '9')

{

n = n \* 16 + s[i] - '0';

}

if (s[i] >= 'a' && s[i] <= 'f')

{

n = n \* 16 + s[i] - 'a' + 10;

}

if (s[i] >= 'A' && s[i] <= 'F')

{

n = n \* 16 + s[i] - 'A' + 10;

}

}

return n;

}

Q167.(20分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q1113.(10分)输入一个字符串到数组S将S的前9个位数字的数存储到T，并将其转换为整数!例如：用户输入字符串1243abc3,则将12433取出以整数形式输出。阅读程序，找出其中的错误，并改正之。

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int k)

{

int i;

k = 0;

for (i = 0; s[i]; i++)

if (0 <= s[i] && s[I] <= 9)

{

t[\*k] = s[i];

\*k += 1;

}

}

void main()

{

char str[100], tOrigin[9];

long int n;

int i, k;

printf("\nPlease enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

printf("The result is: %d", n);

}

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int \*k)

{

int i;

\*k = 0;

for (i = 0; s[i]; i++)

if ('0' <= s[i] && s[i] <= '9')

{

t[\*k] = s[i];

\*k += 1;

}

}

main()

{

char str[100], tOrigin[9];

long int n = 0L;

int i, k;

printf("\nPlease enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

printf("The result is: %d", n);

}

Q1125.(10分)下列给定程序中，函数fun()的功能是：从s所指字符串中，找出t所指字符串的个数作为函数值返回。如，当s所指字符串中的内容为abcdabfab，t所指字符串的内容为ab，则函数返回整数3。请改正程序中的错误，使它得出正确的结果：

#include <stdio.h>

int fun(char \*s, char \*t)

{

int n;

char \*p, \*r;

n = 0;

while (\*s)

{

p = s;

r = t;

while (\*r)

if (\*r == \*p)

{

r++;

p++

}

else break;

if (r == '\0')

n++;

s++;

}

return n;

}

main()

{

char s[100], t[100];

int m;

printf("\nPlease enter string s: ");

scanf("%s", s);

printf("\nPlease enter substring t:");

scanf("%s", t);

m = fun(s, t);

printf("\nThe result is: m=%d\n", m);

}

#include <stdio.h>

int fun(char \*s, char \*t)

{

int n;

char \*p, \*r;

n = 0;

while (\*s)

{

p = s;

r = t;

while (\*r)

if (\*r == \*p)

{

r++;

p++;

}

else break;

if (\*r == '\0')

n++;

s++;

}

return n;

}

main()

{

char s[100], t[100];

int m;

printf("\nPlease enter string s: ");

scanf("%s", s);

printf("\nPlease enter substring t:");

scanf("%s", t);

m = fun(s, t);

printf("\nThe result is: m=%d\n", m);

}

Q114.(20分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

char s[SIZE],t[SIZE],i;

printf("Input s\n");

gets(s);//1

printf("Input t\n");

gets(t);//1

i=MyStrcmp(s,t);//2

if(i>0)//1

printf("string s>string t.\n");

else if(i<0)//1

printf("string s<string t.\n");

else//1

printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

int i;

for (i=0; s[i] == t[i]; i++ ) //6

{

if (s[i] == '\0' ) //2

return 0 ;//2

}

return (s[i]-t[i]); //2

}

Q499.(10分)实现验证输入的必须全部是数字0～9构成的字符串，利用函数checkNumber验证输入的字符串中的每个字符是否全为数字0～9之一，如果是则返回1，否则返回0。然后利用函数getNumberString容错地读取一个全由数字0～9组成的数字字符串。要求编写完整的程序测试运行你所写的函数。

程序的运行示例如下：

输入一个全为数字的字符串: abc

请输入一个全为数字的字符串:a123

请输入一个全为数字的字符串:1a23

请输入一个全为数字的字符串:123

你输入的正确的数字串为：123

#include <stdio.h>

//检查是否全是数字构成的串

int checkNumber(char \*str)

{

int i;

for (i = 0; i < strlen(str); i++)

{

if (str[i] < '0' || str[i] > '9')

{

return 0;

}

}

return 1;

}

char \*getNumberString()

{

static char string[100];

do

{

printf("请输入一个全为数字的字符串:\n");

scanf("%s", string);

}

while (!checkNumber(string));

return string;

}

int main(int argc, char \*argv[])

{

char input[100];

strcpy(input, getNumberString());

printf("你输入的正确的数字串为：%s\n", input);

return 0;

}

Q1133.(10分)下面程序的Squeeze函数的功能是删除字符串s中所出现的与变量c相同的字符。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

void Squeeze(char s[], char c);

main()

{

char a[80], c;

scanf("%s", a);

scanf("%c", &c);

Squeeze(char a[], char c);

printf("%s\n", a);

}

void Squeeze(char s[], char c);

{

int i, j;

for (i = 0; s[i] != '\0'; i++);

{

if (s[i] != 'c')

{

s[j] = s[i];

j++;

}

}

s[i] = '\0';

}

#include <stdio.h>

void Squeeze(char s[], char c);

main()

{

char a[80], c;

gets(a);

scanf("%c", &c);

Squeeze(a, c);

printf("%s\n", a);

}

void Squeeze(char s[], char c)

{

int i, j;

for (i = 0, j = 0; s[i] != '\0'; i++)

{

if (s[i] != c)

{

s[j] = s[i];

j++;

}

}

s[j] = '\0';

}

Q128.(5分)调用GetInput()函数从键盘任意输入一个字符串，然后分别在GetInput()和main()中输出该字符串。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，

修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

|  |
| --- |
| #include  <stdio.h>  void GetInput(char\* s);    int main(void)  {     char \*ptr = NULL;     GetInput(ptr);     puts(ptr);     return 0;  }    void GetInput(char\* s)  {      scanf("%s", s);      puts(s);  } |

#include <stdio.h>

void GetInput(char\* s);

int main(void)

{

char str[80];

char \*ptr = str;

GetInput(ptr);

puts(ptr);

return 0;

}

void GetInput(char\* s)

{

scanf("%s", s);

puts(s);

}

Q173.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

参考答案：

#include <stdio.h>

#define SIZE 80

int MyStrcmp(char s[],char t[]);

main()

{

    char s[SIZE],t[SIZE],i;

    printf("Input s\n");

    gets(s);//1

    printf("Input t\n");

    gets(t);//1

    i=MyStrcmp(s,t);//2

    if(i>0)//1

        printf("string s>string t.\n");

    else if(i<0)//1

        printf("string s<string t.\n");

    else//1

        printf("string s=string t.\n");

}

int MyStrcmp(char s[], char t[])

{

    int i;

    for (i=0; s[i] == t[i]; i++ ) //6

    {

            if (s[i] ==  '\0' )  //2

            return 0 ;//2

    }

    return (s[i]-t[i]); //2

}

Q1316.(10分)写一个函数，将一个字符串中的元音字母复制到另一个字符串，然后输出。

程序的运行示例如下：

输入字符串：Hello,world!

字符串中的元音字母是eoo

|  |
| --- |
| #include <stdio.h>    int main()  {      char str[80], c[80];      void cpy();        printf("\n输入字符串：");      gets(str);      printf("%s", str);      cpy(str, c);      printf("\n字符串中的元音字母是%s", c);        return 0;  }    void cpy(char s[], char c[])  {      int i, j;        for (i = 0, j = 0; s[i] != '\0'; i++)      {          if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||                  s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||                  s[i] == 'u' || s[i] == 'U')          {              c[j] = s[i];              j++;          }      }      c[j] = '\0';  } |

Q498.(20分)对输入的字符串进行长度验证，保证输入的字符串的长度在指定的范围内，如果不在指定的范围内，则一直提示用户输入，直到输入合法长度的字符串为止。

程序的示例运行如下：

请输入一个字符串(长度为[3..5]个字符)：a

请输入一个字符串(长度为[3..5]个字符)：ab

请输入一个字符串(长度为[3..5]个字符)：abcdef

请输入一个字符串(长度为[3..5]个字符)：abc

你输入的字符串为：abc

|  |
| --- |
| #include <stdio.h>    void getString(char \*str, int min, int max)  {      int len;        printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);      scanf("%s", str);      len = strlen(str);        while (len < min || len > max)      {          printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);          scanf("%s", str);          len = strlen(str);      }  }    int main(int argc, char \*argv[])  {      char string[100];        getString(string, 3, 5);        printf("你输入的字符串为：%s\n", string);        return 0;  } |

Q1273.(10分)

|  |
| --- |
| 该程序的作用是将一个字符串中的元音字母复制到另一个字符串，然后输出。改正错误。    #include<stdio.h>    main()  {      char str1[80], str2[80];        printf("输入一个字符串\n");      gets(str1);        Cpy(str1[], str2[]);        printf("输出一个字符串\n");      puts(str2);  }  void Cpy(char s[], char c[])  {      int i, j;        for(i=0; s[i] != '\0'; i++)      {          if(s[i]= 'a'||s[i]= 'A'||s[i]= 'e'||s[i]= 'E'||            s[i]= 'i'||s[i]= 'I'||s[i]= 'o'||s[i]= 'O'||s[i]= 'u'||s[i]= 'U')          {              s[i]=c[j];              j++;          }      }    } |
| #include<stdio.h>  void Cpy(char s[], char c[]);  main()  {      char str1[80], str2[80];        printf("输入一个字符串\n");      gets(str1);        Cpy(str1, str2); //改为：Cpy(str1, str2)        printf("输出一个字符串\n");      puts(str2);  }  void Cpy(char s[], char c[])  {      int i, j = 0; //改为：j=0;        for (i = 0; s[i] != '\0'; i++)      {          if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||              s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||              s[i] == 'u' || s[i] == 'U')              //将=改为：==          {              c[j] = s[i]; //改为:c[j] = s[i];              j++;          }      }      c[j] = '\0';  } |

Q276.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

|  |
| --- |
| #include <stdio.h>  #define SIZE 80  int MyStrcmp(char s[],char t[]);  main()  {      char s[SIZE],t[SIZE],i;      printf("Input s\n");      gets(s);//1      printf("Input t\n");      gets(t);//1      i=MyStrcmp(s,t);//2      if(i>0)//1          printf("string s>string t.\n");      else if(i<0)//1          printf("string s<string t.\n");      else//1          printf("string s=string t.\n");  }  int MyStrcmp(char s[], char t[])  {      int i;      for (i=0; s[i] == t[i]; i++ ) //6      {              if (s[i] ==  '\0' )  //2              return 0 ;//2      }      return (s[i]-t[i]); //2  } |

Q1160.(10分)

|  |
| --- |
| 下面程序的功能是删除字符串中第一次出现的a字符。找出其中错误，并改正之。  #include <stdio.h>  #include <string.h>  void fun(char \*x,int n,\*t);  {   int i,k=0;      x[n]='a';x[n+1]='\0';      while(x[k]!=’a’) k++;      if(k==n){\*t=0;break;}      else      {   for(i=k;i<n;i++)              x[i]=x[i+1];          x[i]='\0';      }  }    main()  {      char x[20];      int len,t;      gets(x);      puts(x);      len=strlen(x);      fun(x,len,t);      if(t==0) printf("Not exist!\n");      else    puts(x);  } |
| #include <stdio.h>  #include <string.h>  void fun(char \*x,int n,int \*t)  {   int i,k=0;      x[n]='a';x[n+1]='\0';      while(x[k]!='a') k++;      if(k==n){\*t=0;}      else      {   for(i=k;i<n;i++)              x[i]=x[i+1];          x[i-1]='\0';      }  }    main()  {      char x[20];      int len,t;      gets(x);      puts(x);      len=strlen(x);      fun(x,len,&t);      if(t==0) printf("Not exist!\n");      else    puts(x);  } |

Q1115.(10分)下面的程序判断是不是回文，是回文返回1，否则返回0。回文是顺读和倒读都一样。请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

#include <stdio.h>

hui(char \*a)

{

int i, k, j = 0;

for (i = 0; i < 15; i++)

if (a[i] != '\0')

j++;

if (j % 2 == 0)

return 0;

for (i = 0, k = j; i <= (j - 1) / 2; i++, k--)

if (a[i] != a[k]) return 0;

return 1;

}

main()

{

int x;

char a[15];

gets(a);

puts(a);

x = hui(a);

if (x == 1)

printf("this is huiwen!!!");

else

printf("this is not huiwen!!!");

}

|  |
| --- |
| #include <string.h>  #include <stdio.h>    hui(char \*a)  {      int i, k, j = 0;        for (i = 0; i < 15; i++)          if (a[i] != '\0')              j++;          else              break;      if (j % 2 == 0)          return 0;      for (i = 0, k = j - 1; i <= (j - 1) / 2; i++, k--)          if (a[i] != a[k])              return 0;      return 1;  }    main()  {      int x;      char a[15];        gets(a);      puts(a);      x = hui(a);      if (x == 1)          printf("this is huiwen!!!");      else          printf("this is not huiwen!!!");  } |

Q1118.(10分)输入一个字符，如果此字符在已知的字符串中出现，则删除此字符，如删除成功就返回被删字符，否则返回'\0'。请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

#include <stdio.h>

char sh(char \*a, char x)

{

int i = 0, k;

char q;

while (a[i] != '\0') i++;

for (int j = 0; j < i; j++);

{

if (a[j] = x)

{

q = a[j];

for (k = j; k <= i; k++)

a[k] = a[k + 1];

return q;

}

}

return '\0';

}

main()

{

char a[6] = "seven", x, ch;

printf("The Original string is:");

puts(a);

printf("Enter x: ");

scanf("%d", &x);

ch = sh(a, x);

if (ch = '\0')

printf("%c not found!!!", x);

else

{

printf("The deleted charactor is :");

puts(x);

}

printf("\nThe Original string changes to :");

puts(a);

}

|  |
| --- |
| #include <string.h>  #include <stdio.h>    char sh(char \*a, char x)  {      int i = 0, k, j;      char q;        while (a[i] != '\0') i++;      for (j = 0; j < i; j++)      {          if (a[j] == x)          {              q = a[j];              for (k = j; k < i; k++)                  a[k] = a[k + 1];              return q;          }      }      return '\0';  }  main()  {      char a[6] = "seven", x, ch;      printf("The Original string is:");      puts(a);      printf("Enter x: ");      scanf("%c", &x);      ch = sh(a, x);      if (ch == '\0')          printf("%c not found!!!", x);      else      {          printf("The deleted charactor is :");          putchar(x);      }      printf("\nThe Original string changes to :");      puts(a);  } |

Q1384.(10分)下面的函数MyStrcmp()用于实现函数strcmp()的功能，将两个字符串s和t进行比较，要求将两个字符串中第一个不相同字符的ASCII码值之差作为MyStrcmp()函数的返回值。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  int MyStrcmp(char s[], char t[]);  main()  {      char  str1[20],str2[20];        printf("Input string:");      gets(str1);      printf("Input another string:");      gets(str2);        if (MyStrcmp(str1[],str2[]) > 0)      {          printf("str1 大于 str2\n");      }      else if (MyStrcmp(str1[],str2[]) < 0)      {          printf("str1 小于 str2\n");      }      else      {          printf("str1 等于 str2\n");      }  }    int MyStrcmp(char s[], char t[])  {      int i;        for (i=0; s[i] == t[i]; i++);      {              if (s[i] = '\0' )  return 0 ;      }      return (s[i] - t[i]);  } |
| #include <stdio.h>  int MyStrcmp(char s[], char t[]);  main()  {      char  str1[20],str2[20];      printf("Input string:");      gets(str1);      printf("Input another string:");      gets(str2);      if (MyStrcmp(str1,str2) > 0)      {          printf("str1 大于 str2\n");      }      else if (MyStrcmp(str1,str2) < 0)      {          printf("str1 小于 str2\n");      }      else      {          printf("str1 等于 str2\n");      }  }    int MyStrcmp(char s[], char t[])  {      int i;        for (i=0; s[i] == t[i]; i++)      {              if (s[i] == '\0' )  return 0 ;      }      return (s[i] - t[i]);  } |

Q1317.(10分)写一个函数，输入一行字符，将此字符串中最长的单词输出。

程序的运行示例如下：

输入一行文本：I am a student.

最长的单词是：student

|  |
| --- |
| #include <stdio.h>    int alphabetic(char c)  {      if ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z'))      {          return 1;      }      else      {          return 0;      }  }    /\* 寻找最长单词 \*/  int longest(char string[])  {      int len = 0, i, length = 0, flag = 1, place, point;        for (i = 0; i <= strlen(string); i++)      {          if (alphabetic(string[i]))          {              if (flag)              {                  point = i;                  flag = 0;              }              else              {                  len++;              }          }          else          {              flag = 1;              if (len > length)              {                  length = len;                  place = point;                  len = 0;              }          }      }      return place;  }    int main()  {      int i;      char line[100];        printf("输入一行文本：\n");      gets(line);      printf("\n最长的单词是：");      for (i = longest(line); alphabetic(line[i]); i++)      {          printf("%c", line[i]);      }      printf("\n");        return 0;  } |

Q1294.(10分)

|  |
| --- |
| 程序改错。编写一个函数Inverse()，实现将字符数组中的字符串逆序存放的功能。（4个错误）  #include<string.h>  #include<stdio.h>    #define ARR\_SIZE = 80;    void Inverse(char str[]);    main()  {      char a[ARR\_SIZE] ;        printf("Please enter a string: ");      gets(a);      Inverse(char a[]);      printf("The inversed string is: ");      puts(a);  }    void Inverse(char str[])  {      int   len, i, j;      char  \*temp;        len = strlen(str);      for (i=0; j=len-1; i<j; i++; j--)        {      temp = str[i];          str[i] = str[j];          str[j] = temp;      }  } |
| #include<string.h>  #include<stdio.h>    #define ARR\_SIZE 80    void Inverse(char str[]);    main()  {      char a[ARR\_SIZE] ;        printf("Please enter a string: ");      gets(a);      Inverse(a);      printf("The inversed string is: ");      puts(a);  }    void Inverse(char str[])  {      int   len, i, j;      char  temp;        len = strlen(str);      for (i=0, j=len-1; i<j; i++, j--)      {      temp = str[i];          str[i] = str[j];          str[j] = temp;      }  } |

Q170.(15分)不用标准库函数strcmp，自己编写一个函数MyStrcmp

实现将两个字符串s和t进行比较，

然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。

要求如下：

(1)函数MyStrcmp函数原型为：

int MyStrcmp(char s[],char t[])

(2)在主函数中

从键盘分别输入两个字符串s和t，

(每个字符串的最大长度为80个字符)

然后调用MyStrcmp函数，

接下来，判定该函数的返回值：

如果大于0，输出"string s>string t.\n"

如果小于0，输出"string s<string t.\n"

如果等于0，输出"string s=string t.\n"

(3)\*\*用gets输入字符串。

\*\*输入提示信息：

输入s之前提示："Input s\n"

输入t之前提示："Input t\n"

|  |
| --- |
| #include <stdio.h>  #define SIZE 80  int MyStrcmp(char s[],char t[]);  main()  {      char s[SIZE],t[SIZE],i;      printf("Input s\n");      gets(s);//1      printf("Input t\n");      gets(t);//1      i=MyStrcmp(s,t);//2      if(i>0)//1          printf("string s>string t.\n");      else if(i<0)//1          printf("string s<string t.\n");      else//1          printf("string s=string t.\n");  }  int MyStrcmp(char s[], char t[])  {      int i;      for (i=0; s[i] == t[i]; i++ ) //6      {              if (s[i] ==  '\0' )  //2              return 0 ;//2      }      return (s[i]-t[i]); //2  } |

Q124.(20分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能。

要求：

（1）按如下函数原型进行编程：

int Mystrlen(char str[]);

（2）在主函数中：

任意输入一个字符串（长度不超过80个字符）;

调用函数Mystrlen()计算输入字符串的实际长度;

然后打印计算结果。

\*\*要求输入提示信息为："Please input the string:\n"

\*\*要求输出格式为:"%d\n"

注意：

（1）字符串中的空格计入字符串实际长度，

但是字符串结束标志不计入字符串实际长度。

（2）必须使用函数编程。否则没有成绩。

|  |
| --- |
| #include <stdio.h>  int Mystrlen(char str[]);  //2  main()  {      char str[80];      int len;          printf ("Please input the string:\n");  //2      gets(str);  //2      len = Mystrlen(str);  //2      printf("%d\n", len);  //2  }  int Mystrlen(char str[])  {      int len = 0, i;      for (i=0; str[i]!=0; i++)  //2      {          len++;  //2      }      return len;  //2  } |

Q564.(10分)请编写一个查找子字符串的程序，并统计子字符串出现的次数。

\*\*输入格式要求："%s" 提示信息："请输入主串:" "请输入要查找的串:"

\*\*输出格式要求："%s,%s:" "子串出现的次数：%d\n" "子串不在主串中\n"

程序运行示例1如下：

请输入主串:Hello,world!

请输入要查找的串:l

Hello,world!,l:子串出现的次数：3

程序运行示例2如下：

请输入主串:Hello,world!

请输入要查找的串:abc

Hello,world!,abc:子串不在主串中

|  |
| --- |
| #include <stdio.h>  #define Max 100  int count(char mstr[], char sstr[])  {      int i, j, k, n = 0;      for (i = 0; mstr[i] != '\0'; i++)      {          j = i;          k = 0;          while (sstr[k] == mstr[j])          {              j++;              k++;          }          if (sstr[k] == '\0')              n++;      }      return n;  }    int main()  {      int n;      char mstr[Max], sstr[Max];      printf("请输入主串:");      scanf("%s", mstr);      printf("请输入要查找的串:");      scanf("%s", sstr);      printf("%s,%s:", mstr, sstr);      n = count(mstr, sstr);      if (n > 0)      {          printf("子串出现的次数：%d\n", n);      }      else          printf("子串不在主串中\n");    } |

Q156.(15分)自己定义一个函数，用字符数组作为函数参数的类型，

以实现和函数strlen相同的功能，即在主函数中任意输入

一个字符串（长度不超过80个字符），调用该函数计算

输入字符串的实际长度，然后打印计算结果。

要求按如下函数原型编程实现计算字符串长度的功能。

int Mystrlen(char str[]);

要求必须按照题目要求和用函数编程，否则不给分。

\*\*要求输入提示信息格式为："Enter a string:"

\*\*输出格式为："The length of the string is:%d\n"

|  |
| --- |
| #include  <stdio.h>  int Mystrlen(char str[]);  main()  {      char str[80];      int len;         printf("Enter a string:");      gets(str);                        //1      len = Mystrlen(str);              //1      printf("The length of the string is:%d\n",len);  }  int Mystrlen(char str[])                   //1  {      int len = 0, i;      for (i=0; str[i]!='\0'; i++)           //1      {          len++;                          //1      }      return len;                           //1  } |

Q307.(15分)利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 80  void Inverse(char str[]);  main()  {      char a[N];      printf("Input a string:\n");      gets(a);//1      Inverse(a);//2      printf("Inversed results:\n");      puts(a);//1  }  void Inverse(char str[])  {      int   len, i, j;      char  temp;//1      len = strlen(str);//1      for (i = 0, j = len - 1; i < j; i++, j--) //4      {          temp = str[i];//1          str[i] = str[j];//1          str[j] = temp;//1      }  } |

Q607.(10分)请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);

\*\*输入格式要求：无

\*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身

程序运行的输入输出样例1：

输入：

abc

输出：

No!

程序运行的输入输出样例2：

输入：

12321

输出：

12321

|  |
| --- |
| #include <stdio.h>  #include <string.h>  int  Judger( char  a[ ]);  int main()  {      char a[80];      int flag;      gets(a);      flag = Judger(a);      if (flag == 1)      {          puts(a);      }      else      {          printf("No!");      }      return 0;  }  int  Judger( char  a[ ])  {      char \*pStar, \*pEnd;      int i, j, flag = 1;      j = strlen(a);      pStar = &a[0];      pEnd = &a[j - 1];      for (i = 0; i < (j / 2); i++)      {          if (\*pStar != \*pEnd)          {              flag = 0;          }          pStar++;          pEnd--;      }      return flag;  } |

Q565.(30分)程序中函数 transform()的功能: 将一个由二进制数字字符组成的字符串转换为与其值相等的十进制整数。规定输入的字符串最多只能包含10位二进制数字字符。

\*\*输入格式要求：

要求对输入的二进制数字字符组成的字符串进行长度和内容确为二进制的判断。

提示信息："输入一个二进制字符串（10位）:" "错误：字符串太长\n" "错误:%c 不是一个二进制字符\n"

\*\*输出格式要求："输入的二进制字符串为" "\n%s 转换成十进制整数为：%d\n"

程序运行示例如下：

输入一个二进制字符串（10位）:10010

输入的二进制字符串为10010

10010 转换成十进制整数为：18

提示：exit(0)函数在stdlib.h库中定义。

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #include <stdlib.h>    int transform(char ch[])  {      int n=0,i=0;      while (ch[i] != '\0')      {          n = n \* 2 + ch[i] - '0';          i++;      }      return n;  }    void main()  {      char s[10];      int i;      int n;      printf("输入一个二进制字符串（10位）:");      gets(s);      if (strlen(s) > 10)      {          printf("错误：字符串太长\n");          exit(0);      }      for (i=0;s[i]!='\0';i++)      {          if (s[i] != '0' && s[i] != '1')          {              printf("错误:%c 不是一个二进制字符\n", s[i]);              exit(0);          }      }      printf("输入的二进制字符串为");      puts(s);      n = transform(s);      printf("\n%s 转换成十进制整数为：%d\n", s, n);  } |

Q594.(10分)编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #define N 100  int getline(char \*s, int len);  int main()  {      char line[N], b[20], \*ptr1, \*ptrb;      int n = 0, j, a[50];      printf("Please enter a string:");      getline(line, N);      ptr1 = line;      while (\*ptr1 != '\0')      {          ptrb = b;          for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)          {              \*ptrb = \*ptr1;          }          if (j)          {              \*ptrb = '\0';              \*(a + n) = atoi(b);              n++;          }          if (\*ptr1 != '\0')ptr1++;      }      printf("the result of output:\n");      for (j = 0; j < n; j++)      {          printf("%10d\n", \*(a + j));      }      return 0;  }    int getline(char \*s, int len)  {      int c;      char \*p = s;      while (--len > 0 && (c = getchar()) != '\n')      {          \*s++ = c;      }      \*s = '\0';      return s - p;  } |

Q1242.(10分)

|  |
| --- |
| 下面程序用于将输入的两个字符串连接起来，然后打印连接后的字符串。找出其中错误并改正之。  #include <stdio.h>  #include <string.h>    main()  {      char    first[80];      char    second[80];        printf("Input the first string:");      gets(first);      printf("Input the second string:");      gets(second);      MyStrCat(first, second);      printf("The result is : %s\n", first);  }    void MyStrCat(char \*dest, char \*source);  {      int i = 0, j = 0;        while (\*(dest+i)!='\0')      {          i++;      }      for (; \*(source+i)!='\0'; i++,j++)      {          \*(dest+j) = \*(source+i);      }      \*(dest+j) = '\0';  } |
| #include <stdio.h>  #include <string.h>  void MyStrCat(char \*dest, char \*source);  main()  {      char    first[80];      char    second[80];      printf("Input the first string:");      gets(first);      printf("Input the second string:");      gets(second);      MyStrCat(first, second);      printf("The result is : %s\n", first);  }    void MyStrCat(char \*dest, char \*source)  {      int i = 0, j = 0;        while (\*(dest+i)!='\0')      {          i++;      }      for (; \*(source+j)!='\0'; i++,j++)      {          \*(dest+i) = \*(source+j);        }      \*(dest+i) = '\0';  } |

1.(15分)编写函数，通过指针连接两个字符串。

\*\*输入格式要求："%s" 提示信息："Enter string 1:" "Enter string 2:"

\*\*输出格式要求："a+b=%s\n"

程序运行示例如下：

Enter string 1:happy

Enter string 2:hollween

a+b=happyhollween

#include <stdio.h>

char \*strcat(char \*str1,char \*str2)

{

char \*p=str1;

while(\*p!='\0') p++;

while(\*p++=\*str2++);

return str1;

}

main()

{

char a[50],b[30];

printf("Enter string 1:");

scanf("%s",a);

printf("Enter string 2:");

scanf("%s",b);

printf("a+b=%s\n",strcat(a,b));

}

2.(10分)编写函数，通过指针将一个字符串反向。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："a=%s\n"

程序运行示例如下：

Enter string:love

a=evol

#include <stdio.h>

revstr(char \*s)

{

char \*p=s,c;

while(\*p) p++;

p--;

while(s<p)

{

c=\*s;

\*s++=\*p;

\*p--=c;

}

}

main()

{

char a[50];

printf("Enter string:");

scanf("%s",a);

revstr(a);

printf("a=%s\n",a);

}

***.***(10分)编程计算圆的面积。

要求：

（1）圆的半径r的值为1到10（包括1和10）之间的正整数；

（2）用符号常量定义PI为3.14；

（3）当圆的面积小于50时输出圆的面积并对圆的面积求累加和，

大于等于50时结束循环；

（4）输出累加和的结果

（5）不用数组编程

\*\*要求输入提示信息为：无输入提示信息和输入数据

\*\*要求输出格式为：

（1）"area=%.2f\n"

（2）"sum=%.2f\n"

# include <stdio.h>

# define PI 3.14

main()

{

int r;

float area,sum=0;

for(r=1;r<=10;r++) //2

{

area=PI\*r\*r; //2

if(area<=50.0) //2

{

printf("area=%.2f\n",area); //2

sum=sum+area; //2

}

else

break; //2

}

***Q2296.***(10分)写一个函数返回三个整数中的中间数。函数原型：int mid(int a, int b, int c)，返回a，b，c三数中大小位于中间的一个数。在main函数中调用该函数进行测试。

\*\*输入格式要求："%d%d%d" 提示信息："请输入三个不同的整数："

\*\*输出格式要求："中间的一个数为%d\n" #include <stdio.h>

int mid(int a, int b, int c)

{

if (a > b)

{

if (b > c)

{

return b;

}

else if (a > c)

{

return c;

}

else

{

return a;

}

}

else if (a > c)

{

return a;

}

else if (b > c)

{

return c;

}

else

{

return b;

}

}

int main()

{

int n1, n2, n3;

printf("请输入三个不同的整数：");

scanf("%d%d%d", &n1, &n2, &n3);

printf("中间的一个数为%d\n", mid(n1, n2, n3));

}

堆栈（Strack）是指这样一段内存，它可以理解为一个筒结构，先放进筒中的数据被后放进筒中的数据“压住”，只有后放进筒中的数据都取出后，先放进去的数据才能被取出，称为“后进先出”。堆栈的长度可随意增加。堆栈结构可用链表实现。设计一个链表结构需包含两个成员：一个存放数据，一个为指向下一个节点的指针。当每次有一个新数据要放入堆栈时，称为“压入堆栈”，这时动态建立一个链表的节点，并连接到链表的结尾；当每次从堆栈中取出一个数据时，称为“弹出堆栈”，这意味着从链表的最后一个节点中取出该节点的数据成员，同时删除该节点，释放该节点所占的内存。 编程用链表方法实现堆栈数据结构。程序至少应该包含三个函数： 1、主函数 2、压栈操作：将5个数据依次压入堆栈 3、出栈操作：将5个数据依次弹出堆栈（注意：因为后进先出的原因，弹出时是逆序，另外，弹出后要删除该节点内存）; \*\*输入提示：printf("请输入要压栈的整数："); \*\*输入格式："%d" \*\*压栈后输出格示："Push %dth Data:%d\n" \*\*出栈后输出格式："Pop %dth Data:%d\n" 程序运行示例： 请输入要压栈的整数：1↙ Push 1th Data:1↙ 请输入要压栈的整数：2↙ Push 2th Data:2↙ 请输入要压栈的整数：3↙ Push 3th Data:3↙ 请输入要压栈的整数：4↙ Push 4th Data:4↙ 请输入要压栈的整数：5↙ Push 5th Data:5↙ Pop 5th Data:5↙ Pop 4th Data:4↙ Pop 3th Data:3↙ Pop 2th Data:2↙ Pop 1th Data:1↙#include <stdio.h>

#include <stdlib.h>

typedef struct stack

{

    int data;

    struct stack \*next;

}                     STACK;

STACK \*head, \*pr;

int nodeNum = 0;                            /\* 堆栈节点数寄存器 \*/

STACK \*CreateNode(int num);

STACK \*PushStack(int num);

int PopStack(void);

int main()

{

    int pushNum[5] , popNum[5], i;

    for (i = 0; i < 5; i++)

    {

        printf("请输入要压栈的整数：");

        scanf("%d",&pushNum[i]);

        PushStack(pushNum[i]);

        printf("Push %dth Data:%d\n", i + 1, pushNum[i]);

    }

    for (i = 0; i < 5; i++)

    {

        popNum[i] = PopStack();

        printf("Pop %dth Data:%d\n", 5 - i, popNum[i]);

    }

    return 0;

}

/\* 函数功能：生成一个新的节点，并为该节点赋初值，返回指向新的节点的指针 \*/

STACK \*CreateNode(int num)

{

    STACK \*p;

    p = (STACK \*)malloc(sizeof(STACK));

    if (p == NULL)

    {

        printf("No enough memory!\n");

        exit(0);

    }

    p->next = NULL;              /\* 为新建的节点指针域赋空指针 \*/

    p->data = num;                   /\* 为新建的节点数据区赋值 \*/

    return p;

}

/\*  函数功能：将整型变量num的值压入堆栈，返回指向链表新节点的指针 \*/

STACK \*PushStack(int num)

{

    if (nodeNum == 0) /\* 若为首节点，则保留该节点地址在head中\*/

    {

        head = CreateNode(num);

        pr = head;

        nodeNum++;  /\* 堆栈节点数寄存器+1 \*/

    }

    else             /\* 若不是首节点，则将新建节点连到链表的结尾处 \*/

    {

        pr->next = CreateNode(num);

        pr = pr->next;

        nodeNum++;  /\* 堆栈节点数寄存器+1 \*/

    }

    return pr;

}

/\*  函数功能：将当前栈顶的数据弹出堆栈，返回从堆栈中弹出的数据 \*/

int PopStack(void)

{

    STACK \*p = head;

    int result;

    for (;;)

    {

        if (p->next == NULL)     /\* 查找最后一个节点 \*/

        {

            break;

        }

        else

        {

            pr = p;             /\* 记录最后一个节点的前一个节点的地址 \*/

            p = p->next;

            nodeNum--;          /\* 堆栈节点数寄存器-1 \*/

        }

    }

    pr->next = NULL;          /\* 将末节点的前一个节点置成末节点 \*/

    result = p->data;

    free(p);

    return result;

}

将一个链表中元素值为x的结点删除。（链表数据域为整数，初始长为6个元素） \*\*输入提示信息："输入数组%d个元素的值。\n" \*\*输入格式："%d" \*\*输出提示："此链表各个结点的数据域为：" \*\*输出格式："%d " \*\*输入提示信息："输入要删除的数据x: " \*\*输入格式："%d" \*\*输出提示： "删除后链表各个结点的数据域为：" \*\*输出格式："%d " 程序运行示例1： 输入数组6个元素的值。 11 22 33 44 55 66 此链表各个结点的数据域为：11 22 33 44 55 66 输入要删除的数据x: 33 删除后链表各个结点的数据域为：11 22 44 55 66 程序运行示例2： 输入数组6个元素的值。↙ 6 2 7 4 5 9 此链表各个结点的数据域为：6 2 7 4 5 9 ↙ 输入要删除的数据x: 8 删除后链表各个结点的数据域为：6 2 7 4 5 9 ↙

#include <stdio.h>

#include <stdlib.h>

#define N 6

struct LNode

{

    int data;

    struct LNode \*next;

}                 ;

struct LNode\* create\_rear(int a[], int n);

void output(struct LNode \*h) ;

struct LNode\* delete\_node(struct LNode\* h, int x);

int main(int argc, char \*argv[])

{

    int a[N], i, x;

    struct LNode\* head;

    printf("输入数组%d个元素的值。\n", N);

    for (i = 0; i < N; i++)

        scanf("%d", &a[i]);

    /\*创建链表head,其结点的值依次为数组a元素的值\*/

    head = create\_rear(a, N);

    /\*删除前输出链表head\*/

    printf("此链表各个结点的数据域为：");

    output(head);

    printf("输入要删除的数据x: ");

    scanf("%d", &x);

    head = delete\_node(head, x);   /\*调用删除函数\*/

    printf("删除后链表各个结点的数据域为：");

    output(head);  /\*删除后输出链表head\*/

    return 0;

}

struct LNode\* create\_rear(int a[], int n)

{

    /\*新建一个链表h，每个结点依次插入到链尾，将链表的头指针返回 \*/

    struct LNode \*h = NULL;

    struct LNode \*s, \*r; /\*用s指向要插入结点，r指向链表的尾结点\*/

    int i;

    for (i = 0; i < n; i++)

    {

        s = (struct LNode \*) malloc(sizeof(struct LNode));

        s->data = a[i];

        s->next = NULL;

        if (h == NULL)

            h = s;        /\*如果链表为空，则头指针h指向s \*/

        else

            r->next = s;  /\*否则将s链接到尾结点r之后     \*/

        r = s;                /\*将r指向尾结点                \*/

    }

    return h;  /\*返回链表的头指针\*/

}

void output(struct LNode \*h)

{

    /\*将链表h的各个结点的数据域依次输出，即遍历该链表\*/

    struct LNode \*p = h;/\*从第一个结点开始，用p依次指向各个结点\*/

    while (p)

    {

        /\*只要p是一个非空结点，则输出其数据域，然后将p后移\*/

        printf("%d ", p->data);

        p = p->next;  //将p后移

    }

    printf("\n");

}

struct LNode\* delete\_node(struct LNode\* h, int x)

{

    /\*将链表h中值为x的结点第一个结点删除，并返回头指针。\*/

    struct LNode \*pre, \*p;/\*pre所指结点为p所指结点的前驱\*/

    p = h;

    while (p && x != p->data)

    {

        /\*如果p不空，且x不等于p所指结点的数据域,p后移，pre为p的前驱\*/

        pre = p;

        p = p->next;

    }

    if (p)

    {

        /\*在链表中找到了要删除的结点p，即p->data为x\*/

        if (p == h)

        {

            /\*p为链首结点，由于p没有前驱，删除后p的后继结点成为链首，需修改头指针\*/

            h = p->next;

        }

        else

        {

            /\*删除的p非链首结点，则p有前驱pre，删除时需将pre后面链接到p的后继结点\*/

            pre->next = p->next;

        }

    }

    return h;

}

ISBN识别码判断

每一本正式出版的图书都有一个ISBN号码与之对应，ISBN码包括9位数字、1位识别码和3位分隔符，其规定格式如“x-xxx-xxxxx-x”，其中符号“-”就是分隔符（键盘上的减号），最后一位是识别码，例如0-670-82162-4就是一个标准的ISBN码。ISBN码的首位数字表示书籍的出版语言，例如0代表英语；第一个分隔符“-”之后的三位数字代表出版社，例如670代表维京出版社；第二个分隔符后的五位数字代表该书在该出版社的编号；最后一位为识别码。

识别码的计算方法如下：

首位数字乘以1加上次位数字乘以2……以此类推，用所得的结果mod 11，所得的余数即为识别码，如果余数为10，则识别码为大写字母X。例如ISBN号码0-670-82162-4中的识别码4是这样得到的：对067082162这9个数字，从左至右，分别乘以1，2，...,9,再求和，即0×1+6×2+……+2×9=158，然后取158 mod 11的结果4作为识别码。

你的任务是编写程序判断输入的ISBN号码中识别码是否正确，如果正确，则仅输出“Right”；如果错误，则输出你认为是正确的ISBN号码。

程序运行结果示例1：

0-123-41562-4↙

Right

程序运行结果示例2：

0-123-41562-7↙

0-123-41562-4

输入格式: 用gets()输入字符串

输入只有一行，是一个字符序列，表示一本书的ISBN号码（保证输入符合ISBN的格式要求）。

输出格式：

输入的ISBN号码的识别码正确，输出信息: "Right"

输入的ISBN号码的识别码错误，按照规定的格式，输出正确的ISBN号码（包括分隔符“-”），输出格式："%s"

#include <stdio.h>

int main()

{

    char isbn[13];

    int i, k = 0, last, p;

    gets(isbn);

    for (i = 0, p = 0; i < 11; i ++)

    {

        if (isbn[i] == '-') continue;

        p ++;

        k += (isbn[i] - 48) \* p;

    }

    last = k % 11;

    if (last == 10)

        last = 'X';

    else

        last += 48;

    if (last == isbn[12])

        printf("Right");

    else

    {

        isbn[12] = last;

        printf("%s", isbn);

    }

    return 0;

}

编程实现从键盘输入5个国名（每个国名最长80个字符），

找出并输出按字典顺序排在最前面的国名

要求：

(1)用gets输入字符串。

(2)

\*\*输入提示信息为："Input five countries' names:\n"

\*\*输出格式为:"The minimum is:%s\n"

#include  <stdio.h>

#include  <string.h>

#define N 80

void main()

{

    int    n;

    char  str[N], min[N];

    printf("Input five countries' names:\n");

    gets(str);

    strcpy(min, str);

    for (n=1; n<5; n++)

    {

        gets(str);

        if (strcmp(str, min) < 0)

        {

            strcpy(min, str);

        }

    }

    printf("The minimum is:%s\n", min);

}

有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。

程序的运行示例如下：

请输入一个字符串：

1234567890

从第几个字符开始复制？3

复制结果为：34567890

输入一行字符，统计其中的英文字符、数字字符、空格字符，以及其他字符的个数。请找出以下程序的错误，并改正之。

|  |
| --- |
| #include <stdio.h>  #include <string.h>    #define ARR\_SIZE = 80;    main()  {      char str[ARR\_SIZE];      int  len, i;      int  letter=0,digit=0,space=0,other=0;        printf("请输入一个字符串:");      gets(str);        len = strlen(str);        for (i=0; i<len; i++)      {          if (a=<str[i]<=z || A=<str[i]<=Z)          {              letter ++;          }          else if (0=<str[i]<=9)          {              digit ++;          }          else if (str[i]=' ' )          {              space ++;          }          else              other ++;      }        printf("英文字符数:%d\n", letter);      printf("数字字符数:%d\n", digit);      printf("空格数:%d\n", space);      printf("其他字符数:%d\n", other);  } |

#include <stdio.h>

#include <string.h>

#define ARR\_SIZE  80

main()

{

    char str[ARR\_SIZE];

    int  len, i;

    int  letter=0,digit=0,space=0,other=0;

    printf("请输入一个字符串:");

    gets(str);

    len = strlen(str);

    for (i=0; i<len; i++)

    {

        if (str[i]>='a'&&str[i]<='z' || str[i]>='A'&&str[i]<='Z')

        {

            letter ++;

        }

        else if (str[i]>='0'&&str[i]<='9')

        {

            digit ++;

        }

        else if (str[i]==' ' )

        {

            space ++;

        }

        else

            other ++;

    }

    printf("英文字符数:%d\n", letter);

    printf("数字字符数:%d\n", digit);

    printf("空格数:%d\n", space);

    printf("其他字符数:%d\n", other);

}

编写一个程序，将一个字符串s2插入到字符串s1中，其起始插入位置为n。

\*\*输入格式要求："%d" 使用gets()接收字符串s1和s2 提示信息："main string:" "sub string:" "site of begining:(<=%d)"

\*\*输出格式要求："After instert:%s\n"

程序运行示例如下：

main string:hello,world!

sub string:c

site of begining:(<=12)6

After instert:hello,c world!

#include <stdio.h>

#include <string.h>

#define N 100

int main()

{

    int n, i, j, k, len;

    char s1[N], s2[N], s3[2 \* N];

    printf("main string:");

    gets(s1);

    printf("sub string:");

    gets(s2);

    len = strlen(s1);

    do

    {

        printf("site of begining:(<=%d)", len);

        scanf("%d", &n);

    }

    while (n > len);

    for (i = 0; i < n; i++)

    {

        s3[i] = s1[i];

    }

    for (j = 0; s2[j] != '\0'; j++)

    {

        s3[i + j] = s2[j];

    }

    for (k = n; s1[k] != '\0'; k++)

    {

        s3[j + k] = s1[k];

    }

    s3[j + k] = '\0';

    printf("After instert:%s\n", s3);

    return 0;

}

指针和字符数组的应用,用五种方法输出字符串“China”。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

int i = 0;

char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[5];

while (str[i] != '\0')

{

putchar(\*str);

str++;

}

scanf("%s", ptr);

puts(ptr);

ptr1 = str;

puts(ptr1);

ptr2 = "China";

puts(ptr2);

str2 = "China";

printf("%s", str2);

}

#include <stdio.h>

main()

{

    int i = 0;

    char str[6] = {'C', 'h', 'i', 'n', 'a', '\0'}, str1[6] , \*ptr, \*ptr1, \*ptr2 , str2[6];

    while (str[i] != '\0')

    {

        putchar(str[i]);

        i++;

    }

    ptr = str1;

    scanf("%s", ptr);

    puts(ptr);

    ptr1 = str;

    puts(ptr1);

    ptr2 = "China";

    puts(ptr2);

    strcpy(str2, "China");

    printf("%s", str2);

}

下面的程序将数字进行加密，其中的每一个数字转化为另一个不同的数字。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

main()

{

char s[] = "24635", c, i;

for (i = 0; c = s[i]; i++)

{

switch ((c - '0')

{

case 2:

case 3:

putchar(c + 4);

continue;

case 4:

putchar(c + 4);

break;

case 5:

putchar(c + 3);

default:

putchar(c + 2);

}

putchar('\n');

}

}

#include <stdio.h>

main()

{

    char s[] = "24635", c, i;

    for (i = 0; c = s[i]; i++)

    {

        switch (c - '0')

        {

        case 2:

        case 3:

            putchar(c + 4);

            break;

        case 4:

            putchar(c + 4);

            break;

        case 5:

            putchar(c + 3);

            break;

        default:

            putchar(c + 2);

        }

        putchar('\n');

    }

}

简单的口令检查程序

按下述要求编写口令检查程序（假设正确的口令为8888,正确口令"8888"为字符串）。

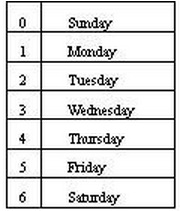
（1）若输入口令正确，则提示“You are welcome!”，程序结束。

（2）若输入口令不正确，则提示“Wrong password!”，同时检查口令是否已输入3次，若未输入3次，则提示“Enter again：”，且允许用户再次输入口令；若已输入3次，则提示“You have entered three times! You are not welcome!”，且不允许用户再输入口令，程序结束。

\*\*\*\*\*\*\*\*？？？？？？没有答案？？？

星期查找

任意输入英文的星期几，通过查找如图所示的星期表，输出其对应的数字，若查到表尾，仍未找到，则输出错误提示信息。



提示：用一个二维字符数组weekDay来存放如图所示的星期表的内容（字符串）。输入待查找的字符串，然后在星期表中顺序查找与输入字符串相匹配的字符串。找到的字符串在星期表数组中的第一维下标（行号）即为题目所求。

程序运行结果示例1：

Please enter a string:

Friday↙

Friday is 5

程序运行结果示例2：

Please enter a string:

Fruday↙

Not found!

**输入格式:**字符串输入采用gets()函数

**输出格式：**

输入提示信息："Please enter a string:\n"

找到了，输出："%s is %d\n"

没找到，输出："Not found!\n"

#include  <stdio.h>

#include  <string.h>

#define   WEEKDAYS  7             /\*每星期天数\*/

#define   MAX\_STR\_LEN  10         /\*字符串最大长度\*/

int main()

{

    int    i, pos;

    int    findFlag = 0;          /\* 置找到标志为假 \*/

    char   x[MAX\_STR\_LEN];

    char   weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",

                                     "Wednesday", "Thursday", "Friday",                                  "Saturday"

                                    };

    printf("Please enter a string:\n");

    gets(x);

    /\*scanf("%s", x);                /\* 输入待查找的字符串 \*/

    for (i = 0; i < WEEKDAYS && !findFlag; i++)

    {

        if (strcmp(x, weekDay[i]) == 0)

        {

            pos = i;          /\*记录找到的位置\*/

            findFlag = 1;    /\*若找到，则置找到标志为真，退出循环\*/

        }

    }

    if (findFlag)        /\*找到标志为真，说明找到\*/

        printf("%s is %d\n", x, pos);

    else                  /\*找到标志为假，说明未找到\*/

        printf("Not found!\n");

    return 0;

}

从键盘输入一个字符串（最长不超过80字符），然后从键盘输入一个字符，删除字符串中与该字符相同的字符。其中，在字符串中删除与某字符相同的字符，要求用字符数组作函数参数编程实现。已知该函数的原型为：

void Squeeze(char s[], char c);

程序运行结果示例：

Input a string:

hello, my friend!↙

Input a character:

!↙

Results:hello, my friend

输入格式:

输入字符串用 gets()函数

输入单个字符用 getchar()函数

输出格式：

输入字符串的提示信息： "Input a string:\n"

输入单个字符的提示信息： "Input a character:\n"

输出格式： "Results:%s\n"

注意：不能使用指针、结构体和goto语句编程。

#include <stdio.h>

void  Squeeze(char s[], char c);// 2

int main()

{

    char  str[20], ch;

    printf("Input a string:\n");  //1

    gets(str);                    //1

    printf("Input a character:\n");//1

    ch = getchar();

    Squeeze(str, ch);      //2

    printf("Results:%s\n", str);//1

    return 0;

}

void  Squeeze(char s[], char c)//2

{

    int i, j;

    for (i = j = 0; s[i] != '\0'; i++)//1

    {

        if (s[i] != c)//1

        {

            s[j] = s[i];//1

            j++;//1

        }

    }

    s[j] = '\0'; //1

}

从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

若找到，则打印该字符串，否则打印"Not find!"。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include "stdio.h"

#include "string.h"

main()

{

int i, findFlag;

char x[13];

char \*str[13]={ "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag);

{

if(strcmp(x,str)=0)

{

findFlag = 0;

}

i++;

str++;

}

if (findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

#include "stdio.h"

#include "string.h"

main()

{

    int i=0, findFlag=1;

    char x[13];

    char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

    printf("Input string:\n");

    gets(x);

    while( i<6 && findFlag)

    {

        if(strcmp(x,str[i])==0)

        {

            findFlag = 0;

        }

        i++;

    }

    if ( !findFlag )

    {

        printf("%s\n",x);

    }

    else

    {

        printf("Not find!\n");

    }

}

创建动态链表并打印链表内容。要求通过键盘读入学生的学号和分数，当输入的学号为0时表示链表创建结束，并将读入的学生学号和成绩打印输出。下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。改错时不能改变程序原有的意图。

注意：将修改后的完整的源程序写在答题区内。

#include <stdio.h>

#define SIZE sizeof(struct node)

struct node

{

long num;

float score;

struct node \*next;

}

int main()

{

int n = 0;

struct node \*head = NULL, \*p1, \*p2;

p1 = p2 = (struct node\*)malloc(SIZE);

printf("Input %d node data:\n", n + 1);

scanf("%d%f", &p1->num, &p1->score);

while (p1->num != 0)

{

if (++n == 1)

head = p1;

else

p2->next = p1;

p1 = p2;

p1 = (struct node\*)malloc(SIZE);

printf("Input %d node data:\n", n + 1);

scanf("%d%f", &p1->num, &p1->score);

}

free(p1);

printf("Prit list:\n");

p2->next = NULL;

if (head != NULL)

{

p2=head;

while (p1 != NULL)

{

printf("num:%d\tscore:%.2f\n", p1->num, p1->score);

p1 = p1->next;

}

}

return 0;

}

#include <stdio.h>

#include <stdlib.h>

#define SIZE sizeof(struct node)

**struct** node

{

**long** num;

**float** score;

**struct** node \*next;

}              ;

**int** main()

{

**int** n = 0;

**struct** node \*head = NULL, \*p1, \*p2;

    p1 = p2 = (**struct** node\*)**malloc**(SIZE);

**printf**("Input %d node data:\n", n + 1);

**scanf**("%d%f", &p1->num, &p1->score);

**while** (p1->num != 0)

    {

**if** (++n == 1)

            head = p1;

**else**

            p2->next = p1;

        p2 = p1;

        p1 = (**struct** node\*)**malloc**(SIZE);

**printf**("Input %d node data:\n", n + 1);

**scanf**("%d%f", &p1->num, &p1->score);

    }

**free**(p1);

**printf**("Prit list:\n");

    p2->next = NULL;

**if** (head != NULL)

    {

        p1 = head;

**while** (p1 != NULL)

        {

**printf**("num:%d\tscore:%.2f\n", p1->num, p1->score);

            p1 = p1->next;

        }

    }

**return** 0;

}

编写程序将如下的一个数组a向右旋转90度，构成新的数组b，并打印出来。

a =

1 2

3 4

5 6

7 8

旋转后

b =

7 5 3 1

8 6 4 2

\*\*输出提示信息："Array a:\n" "Array b:\n"

\*\*输出格式要求："%4d"

程序运行示例如下：

Array a:

1 2

3 4

5 6

7 8

Array b:

7 5 3 1

8 6 4 2

#include <stdio.h>

#define M 4

#define N 2

main()

{

**int** a[M][N] = {{1, 2}, {3, 4}, {5, 6}, {7, 8}};

**int** b[N][M], i, j;

**printf**("Array a:\n");

**for** (i = 0; i < M; i++)

    {

**for** (j = 0; j < N; j++)

        {

**printf**("%4d", a[i][j]);

            b[j][M - 1 - i] = a[i][j];

        }

**printf**("\n");

    }

**printf**("Array b:\n");

**for** (i = 0; i < N; i++)

    {

**for** (j = 0; j < M; j++)

        {

**printf**("%4d", b[i][j]);

        }

**printf**("\n");

    }

}

用指针编程实现3X4的二维数组的元素读入以及求此二维数组的最大值及最大值下标

请用以下函数实现：

void Input(int \*p,int m, int n); /\*数组元素读入函数\*/

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol); /\*求最大值及下标函数\*/

\*\*\*输入提示信息："Please input your data:\n"

\*\*\*输入格式要求：无格式要求

\*\*\*输出格式要求："The maximum is %d, which is in row %d, colum %d\n"

样例：

Please input your data:

\*\*输入样例：

3 5 2 7 1 6 12 11 4 10 8 9

\*\*输出样例：

The maximum is 12, which is in row 1, colum 2

参考答案：

#include <stdio.h>

#define M 3

#define N 4

void Input(int \*p, int m, int n); //1

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol); //1

main()

{

int score[M][N], maxScore, row, col;

printf("Please input your data:\n");

Input(\*score, M, N); //1

maxScore = FindMax(\*score, M, N, &row, &col); //2

printf("The maximum is %d, which is in row %d, colum %d\n", maxScore, row, col);

}

void Input(int \*p, int m, int n)

{

int i, j;

for (i = 0; i < m; i++) //1

{

for (j = 0; j < n; j++) //1

scanf("%d", &p[i \* n + j]);

}

}

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol )

{

int i, j, max;

max = p[0]; //1

\*pRow = 0;

\*pCol = 0;

for ( i = 0; i < m; i++ ) //1

{

for ( j = 0; j < n; j++ ) //1

{

if ( p[ i \* n + j ] > max ) //1

{

max = p[ i \* n + j]; //1

\*pRow = i;

\*pCol = j;

}

}

}

return max;

}

题目：去除字符串中的空格并分解单词。

要求：在main()中调用下述各函数，完成字符串中去除空格并分解单词的功能。

函数原型如下：

函数一：单词内是否包含数字 int IsNumIn(char word[])

函数二：单词内是否包含空格 int IsSpcIn(char word[])

函数三：去掉单词的前后空格，tab键和换行符 Trim(char oldWord[], char newWord[])

函数四：单词内部有空格，分解成多个单词 Seg(char words[], char wArray[][100] ) 假设单词内部只有一个空格，没有两个连续空格的情况发生。

要求利用gets函数来获得用户输入的字符串，不能用scanf来获得用户输入的字符串

参考输入，输出

参考输入

ros2e

参考输出

error

参考输入

︺︺hello︺（ ︺代表一个空格）

参考输出

hello

参考输入

︺︺hello world︺

参考输出

hello

world

参考答案：

#include <stdio.h>

#include <ctype.h>

#define N 100

int IsNumIn(char word[]);

int IsSpcIn(char word[]);

void Trim(char oldWord[], char newWord[]);

void Seg(char words[], char wArray[][100] );

int main()

{

char str1[N],str[N],w\_Arr[20][100];

int i,j;

for(i=0;i<N;i++)

{

\*(str1+i)=0;

\*(str+i)=0;

}

for(i=0;i<20;i++)

{

for(j=0;j<100;j++)

w\_Arr[i][j]=0;

}

gets(str1);

Trim(str1,str);

if(IsNumIn(str))

printf("error");

else

{

if(IsSpcIn(str))

{

Seg(str,w\_Arr);

for(i=0;w\_Arr[i][0]!='\0';i++)

{

puts(w\_Arr[i]);

}

}

else

puts(str);

}

return 0;

}

int IsNumIn(char word[])

{

int i;

for(i=0;\*(word+i)!='\0';i++)

{

if(isdigit(\*(word+i)))

return 1;

}

return 0;

}

int IsSpcIn(char word[])

{

int i;

for(i=0;\*(word+i)!='\0';i++)

{

if(isspace(\*(word+i)))

return 1;

}

return 0;

}

void Trim(char oldWord[], char newWord[])

{

int i=0,j=0;

while(isspace(\*(oldWord+i)))

i++;

while( \*(oldWord+i) != '\0')

{

\*(newWord+j)=\*(oldWord+i);

i++;

j++;

}

i--;

j--;

for(;isspace(\*(oldWord+i));i--,j--)

{

\*(newWord+j)=0;

}

}

void Seg(char words[], char wArray[][100] )

{

int i,j=0,k=0;

for(i=0;i<N;i++)

{

if(isalpha(\*(words+i)))

{

wArray[j][k]=\*(words+i);

k++;

}

else

{

j++;

k=0;

}

}

}

单词接龙  
阿泰和女友小菲用英语短信玩单词接龙游戏。一人先写一个英文单词，然后另一个人回复一个英文单词，要求回复单词的开头有若干个字母和上一个人所写单词的结尾若干个字母相同，重合部分的长度不限。（如阿泰输入happy，小菲可以回复python，重合部分为py。）现在，小菲刚刚回复了阿泰一个单词，阿泰想知道这个单词与自己发过去的单词的重合部分是什么。他们两人都是喜欢写长单词的英语大神，阿泰觉得用肉眼找重合部分实在是太难了，所以请你编写程序来帮他找出重合部分。  
  
程序运行结果示例1：  
happyC:\Users\花间独酌\AppData\Local\Temp\{G]K_~VQ2`J3FND@QJ6C9GT.png  
pythenC:\Users\花间独酌\AppData\Local\Temp\{G]K_~VQ2`J3FND@QJ6C9GT.png  
py  
  
程序运行结果示例2：  
sunC:\Users\花间独酌\AppData\Local\Temp\{G]K_~VQ2`J3FND@QJ6C9GT.png  
unknownC:\Users\花间独酌\AppData\Local\Temp\{G]K_~VQ2`J3FND@QJ6C9GT.png  
un  
  
输入格式: "%s%s"  
输出格式： "%s\n"  
  
  
  
  
#include <stdio.h>  
#include <string.h>  
char a[81], b[81];  
int main()  
{                            
    int maxlen, testlen, la;  
    scanf("%s%s", a, b);  
    la = strlen(a);  
    maxlen = strlen(b);  
    for (testlen = maxlen; testlen > 0; testlen--)  
        if (!strncmp(b, a + la - testlen, testlen))  
        {                            
            printf("%s\n", a + la - testlen);  
            break;  
        }  
    return 0;  
}

用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。  
\*\*输入格式要求："%s"    
输入提示信息：  
"Input a string:"    
"Input a character:"  
\*\*输出格式要求："Results:%s\n"  
程序运行示例1如下：  
Input a string:hello,world!  
Input a character:o  
Results:hell,wrld!  
  
  
  
#include <stdio.h>  
#include <string.h>  
#define N 100  
void  Squeeze(char \*s, char c);  
int main()  
{                    
    char  str[20], ch;  
    printf("Input a string:");  
    gets(str);  
    printf("Input a character:");  
    ch = getchar();  
    Squeeze(str, ch);  
    printf("Results:%s\n", str);  
    return 0;  
}                    
void  Squeeze(char \*s, char c)  
{                    
    char str[N];  
    char \*t = str;  
    strcpy(t, s);  
    for (; \*t != '\0'; t++)  
    {                    
        if (\*t != c)  
        {                    
            \*s = \*t;  
            s++;  
        }  
    }  
    \*s = '\0';  /\* 在字符串t2的末尾添加字符串结束标志 \*/  
}

输入一行字符（第一个字符有可能是空格），用函数编程统计其中有多少单词。假设单词之间以空格分开。要求如下：

（1）在主函数中从键盘输入字符串，字符串的最大长度为80个字符

调用CountWords函数，并统计字符串中的单词个数。

（2）在子函数CountWords中统计字符串中的单词个数。函数原型为：

int CountWords(char str[]);

(3)\*\*输入提示信息："Input a string:"

\*\*输出提示信息和格式："Numbers of words = %d\n"

\*\*用gets()输入字符串

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

#include <stdio.h>

int CountWords(char str[]);

int main()

{

char str[80];

printf("Input a string:");

gets(str);//1

printf("Numbers of words = %d\n", CountWords(str));//1

return 0;

}

int CountWords(char str[])//1

{

int i, num;

num = (str[0] != ' ') ? 1 : 0;//1

for (i = 1; str[i] != '\0'; i++)//1

{

if (str[i] != ' ' && str[i - 1] == ' ')//1

{

num++;//1

}

}

return num;//1

}

按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%5d"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

#include<stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

int main()

{

int n,a[10][10];

printf("Input n:\n");

scanf("%d",&n);

InputMatrix(a,n);

Transpose(a,n);

printf("The transposed matrix is:\n");

PrintMatrix(a,n);

return 0;

}

void Transpose(int a[][N], int n)

{

int b[10][10];

int i,j;

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

b[j][i]=a[i][j];

}

}

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

a[i][j]=b[i][j];

}

}

}

void InputMatrix(int a[][N], int n)

{

int i,j;

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

scanf("%d",&a[i][j]);

}

}

}

void PrintMatrix(int a[][N], int n)

{

int i,j;

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

printf("%5d",a[i][j]);

}

putchar('\n');

}

}

找出一个2\*6的二维数组中的鞍点，即该位置上的元素是该行上的最大值，是该列上的最小值。二维数组也可能没有鞍点。

\*\*输入提示信息：无

\*\*输入格式要求：%d

\*\*输出格式要求：若存在鞍点，则输出格式用"%d\n"，若没有鞍点，则输出"no saddle point\n"

样例输入：

12 45 33  3 5 54

23 49 58 22 7 55

样例输出：

54

#include<stdio.h>

int main()

{

int a[2][6];

int i,j;

for(i=0;i<2;i++)

{

for(j=0;j<6;j++)

{

scanf("%d",&a[i][j]);

}

}

int flag1=0,flag2=0,flag=1;

for(i=0;i<6;i++)

{

if(a[0][flag1]<a[0][i])

flag1=i;

if(a[1][flag2]<a[1][i])

flag2=i;

}

if(a[0][flag1]<a[1][flag1])

{

printf("%d\n",a[0][flag1]);

flag=0;

}

if(a[1][flag2]<a[0][flag2])

{

printf("%d\n",a[0][flag2]);

flag=0;

}

if(flag)

{

printf("no saddle point\n");

}

return 0;

}

按如下函数原型用函数编程解决如下的日期转换问题（要求考虑闰年的问题）：

输入某年某月某日，计算并输出它是这一年的第几天。

/\* 函数功能： 对给定的某年某月某日，计算它是这一年的第几天

函数参数： 整型变量year、month、day，分别代表年、月、日

函数返回值：这一年的第几天 \*/

int DayofYear(int year, int month, int day);

下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

|  |
| --- |
| #include <stdio.h>  **int** DayofYear(**int** year, **int** month, **int** day);  **int** dayTab[2][13] =  {      {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},      {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}  };    main()  {  **int** year, month, day;  **printf**("Please enter year, month, day:");  **scanf**("%d,%d,%d", year, month, day);  **printf**("yearDay = %d\n" DayofYear(year, month, day));  }  **int** DayofYear(**int** year, **int** month, **int** day);  {  **int** i, leap;      leap = year / 4 = 0 & year / 100 != 0 | year / 400 = 0;  **for** (i = 0, i < month, i++);      {          day = day - dayTab[leap][i];      }  } |

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

（4）改错时不能改变程序原有的意图，也不要改变代码的输入输出格式。

#include <stdio.h>

int DayofYear(int year, int month, int day);

int dayTab[2][13] =

{

{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},

{0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}

};

main()

{

int year, month, day;

printf("Please enter year, month, day:\n");

scanf("%d,%d,%d", &year,&month,&day);

printf("yearDay = %d\n" ,DayofYear(year, month, day));

}

int DayofYear(int year, int month, int day)

{

int i,leap;

leap = ((year%4==0 && year%100!=0) || year%400==0);

for (i = 0; i < month; i++)

{

day = day + dayTab[leap][i];

}

return day;

}

用二维字符数组方法编程实现从键盘任意输入10个字符串，将这10个字符串按字典顺序排序后打印输出，字符串最大长度80。

\*\*输入提示信息格式: 无

\*\*输入数据格式要求: 要求读入的字符串可以包含有空格

\*\*输出数据格式要求: 一行输出一个字符串

注:不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include<stdio.h>

#include<string.h>

int main()

{

char str[10][81];

int i;

for(i=0;i<10;i++)

{

gets(str[i]);

}

int j;

for(i=0;i<10;i++)

{

for(j=i+1;j<10;j++)

{

if(strcmp(str[i],str[j])>0)

{

char s[81];

strcpy(s,str[i]);

strcpy(str[i],str[j]);

strcpy(str[j],s);

}

}

}

for(i=0;i<10;i++)

{

printf("%s\n",str[i]);

}

return 0;

}

编程将下列矩阵中的元素向右移动一列，最右一列移至第一列。

1 4 6

8 10 12

注意：移动过程中对同一个二维数组操作。

输入格式：无

输出格式："%6d"（换行使用"\n"）

运行结果：

6 1 4

12 8 10

#include<stdio.h>

void swap(int \*x,int \*y)

{

int temp;

temp=\*x;

\*x=\*y;

\*y=temp;

}

int main()

{

int a[2][3]={{1,4,6},{8,10,12}};

swap(&a[0][0],&a[0][1]);

swap(&a[1][0],&a[1][1]);

swap(&a[0][0],&a[0][2]);

swap(&a[0][1],&a[1][2]);

int i,j;

for(i=0;i<2;i++)

{

for(j=0;j<3;j++)

{

printf("%6d",a[i][j]);

}

putchar('\n');

}

return 0;

}

1.#include <stdio.h>

struct date\_rec

{

int day;

int month;

int year;

} ;

struct date\_rec current\_date;

int days\_of\_month[][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}, {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}

} ;

int is\_leap(int year)

{

return ((year % 400 == 0) || ((year % 4 == 0) && (year % 100 != 0)));

}

void input\_date(struct date\_rec \*current\_date)

{

printf("请输入当前日期（年 月 日）：");

scanf("%d%d%d", &current\_date->year, &current\_date->month, &current\_date->day);

}

void increment\_date(struct date\_rec \*current\_date)

{

current\_date->day++;

if (current\_date->day > days\_of\_month[is\_leap(current\_date->year)][current\_date->month])

{

current\_date->day = current\_date->day - days\_of\_month[is\_leap(current\_date->year)][current\_date->month]

;

current\_date->month++;

if (current\_date->month > 12)

{

current\_date->year++;

current\_date->month = current\_date->month - 12;

}

}

}

void output\_date(struct date\_rec \*current\_date)

{

printf("当前日期：%d年%d月%d日！", current\_date->year,

current\_date->month, current\_date->day);

}

int main()

{

input\_date(&current\_date);

increment\_date(&current\_date);

output\_date(&current\_date);

return 0;

}

给定如下定义：

struct date\_rec

{

int day ;

int month ;

int year ;

} ;

struct date\_rec current\_date ;

写一个程序包含如下的函数，完成：

(a) 输入current\_date的值：

void input\_date(struct date\_rec \*current\_date)

(b) 将current\_date增加1天：

void increment\_date(struct date\_rec \*current\_date)

(c) 显示current\_date的值：

void output\_date(struct date\_rec \*current\_date)

考虑每个月的实际天数，同时也考虑闰年的情况。

\*\*输入格式要求："%d%d%d" 提示信息："请输入当前日期（年 月 日）："

\*\*输出格式要求："当前日期：%d年%d月%d日！" （加1天后的新日期）

2.求矩阵A(2\*3)的转置B(3\*2)。如

A=[

1

4

2

5

3

6

]

B=

?

?

1

2

3

4

5

6

?

?

\*\*输出格式要求："%d "

程序运行示例如下：

1 4

2 5

3 6

#include <stdio.h>

int a[2][3]={{1,2,3},{4,5,6}};

main()

{

int i,j,b[3][2];

for(i=0;i<=1;i++)

for(j=0;j<=2;j++)

b[j][i]=a[i][j];

for(i=0;i<=2;i++,printf("\n"))

for(j=0;j<=1;j++)

printf("%d ",b[i][j]);

}

3.任意输入英文的星期几，通过查找星期表，输出其对应的数字，

若查到表尾，仍未找到，则输出错误提示信息。

下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

改错时不能改变程序原有的意图，不能改变函数原型

#include <stdio.h>

#define WEEKDAYS 7

#define MAX\_STR\_LEN 10

main()

{

int i, pos;

int findFlag;

char x[MAX\_STR\_LEN];

char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",

"Wednesday", "Thursday", "Friday", "Saturday"

};

printf("Please enter a string:\n");

scanf("%s", &x[i]);

for (i = 0; i < WEEKDAYS || !findFlag; i++)

{

if (x[i] == weekDay[i])

{

pos = i;

findFlag = 1;

}

}

if (findFlag)

{

printf("%s is %d\n", x, pos);

}

else

{

printf("Not found!\n");

}

}

#include <stdio.h>（疑似为参考答案。是个修改题）

#define WEEKDAYS 7

#define MAX\_STR\_LEN 10

main()

{

int i, pos;

int findFlag;

char x[MAX\_STR\_LEN];

char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",

"Wednesday", "Thursday", "Friday", "Saturday"

};

printf("Please enter a string:\n");

scanf("%s", &x[i]);

for (i = 0; i < WEEKDAYS || !findFlag; i++)

{

if (x[i] == weekDay[i])

{

pos = i;

findFlag = 1;

}

}

if (findFlag)

{

printf("%s is %d\n", x, pos);

}

else

{

printf("Not found!\n");

}

}

4.矩阵转置

某二维数组存放的数据构成一个n\*n的方阵，其中n<=5。写程序，从键盘输入n的值（n<=6），该n\*n矩阵中各元素的值按下面的公式计算：

a[i][j] = i \* n + j + 1

其中，a[i][j]表示第i行第j列的元素。要求分别输出该矩阵和它的转置矩阵。

输入提示信息：无

输入n的输入格式：%d

输出提示："The original matrix is:\n"

输出提示："The changed matrix is:\n"

输出格式：对矩阵的输出要求：列与列之间为%3d，行与行之间为\n

输入样例1：

5↙

输出样例1：

The\_original\_matrix\_is:

\_\_1\_\_2\_\_3\_\_4\_\_5

\_\_6\_\_7\_\_8\_\_9\_10

\_11\_12\_13\_14\_15

\_16\_17\_18\_19\_20

\_21\_22\_23\_24\_25

The\_changed\_matrix\_is:

\_\_1\_\_6\_11\_16\_21

\_\_2\_\_7\_12\_17\_22

\_\_3\_\_8\_13\_18\_23

\_\_4\_\_9\_14\_18\_24

\_\_5\_10\_15\_20\_25

(输出样例中下划线“\_”代表空格)

输入样例2：

4↙

输出样例2：

The original matrix is:

\_\_1\_\_2\_\_3\_\_4

\_\_5\_\_6\_\_7\_\_8

\_\_9\_10\_11\_12

\_13\_14\_15\_16

The changed matrix is:

\_\_1\_\_5\_\_9\_13

\_\_2\_\_6\_10\_14

\_\_3\_\_7\_11\_15

\_\_4\_\_8\_12\_16

(输出样例中下划线“\_”代表空格)

#include <stdio.h>

#include <stdlib.h>

int main()

{

int a[6][6];

int i, j, n, tep;

scanf("%d", &n);

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

a[i][j] = i \* n + j + 1;

}

}

printf("The original matrix is:\n");

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

printf("%3d", a[i][j]);

}

printf("\n");

}

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

if (i < j)

{

tep = a[i][j];

a[i][j] = a[j][i];

a[j][i] = tep;

}

}

}

printf("The changed matrix is:\n");

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

printf("%3d", a[i][j]);

}

printf("\n");

}

return 0;

}

5.星期查找

任意输入英文的星期几，通过查找如图所示的星期表，输出其对应的数字，若查到表尾，仍未找到，则输出错误提示信息。

提示：用一个二维字符数组weekDay来存放如图所示的星期表的内容（字符串）。输入待查找的字符串，然后在星期表中顺序查找与输入字符串相匹配的字符串。找到的字符串在星期表数组中的第一维下标（行号）即为题目所求。

程序运行结果示例1：

Please enter a string:

Friday↙

Friday is 5

程序运行结果示例2：

Please enter a string:

Fruday↙

Not found!

输入格式: 字符串输入采用gets()函数

输出格式：

输入提示信息："Please enter a string:\n"

找到了，输出："%s is %d\n"

没找到，输出："Not found!\n"

#include <stdio.h>

#include <string.h>

#define WEEKDAYS 7 /\*每星期天数\*/

#define MAX\_STR\_LEN 10 /\*字符串最大长度\*/

int main()

{

int i, pos;

int findFlag = 0; /\* 置找到标志为假 \*/

char x[MAX\_STR\_LEN];

char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",

"Wednesday", "Thursday", "Friday", "Saturday"

};

printf("Please enter a string:\n");

gets(x);

/\*scanf("%s", x); /\* 输入待查找的字符串 \*/

for (i = 0; i < WEEKDAYS && !findFlag; i++)

{

if (strcmp(x, weekDay[i]) == 0)

{

pos = i; /\*记录找到的位置\*/

findFlag = 1; /\*若找到，则置找到标志为真，退出循环\*/

}

}

if (findFlag) /\*找到标志为真，说明找到\*/

printf("%s is %d\n", x, pos);

else /\*找到标志为假，说明未找到\*/

printf("Not found!\n");

return 0;

}

6.编程计算2×3阶矩阵A和3×2阶矩阵B之积C。

要求：

(1)从键盘分别输入矩阵A和B，

输出乘积矩阵C

(2)

\*\*输入提示信息为：

输入矩阵A之前提示："Input 2\*3 matrix a:\n"

输入矩阵B之前提示："Input 3\*2 matrix b:\n"

\*\*输入矩阵中每个值的格式为："%d"

\*\*输出格式为：

输出矩阵C之前提示："Results:\n"

输出矩阵C中每个值的格式："%6d"

输出矩阵C各行结束时换行

例如：

Input 2\*3 matrix a:

1 2 3

0 1 2

Input 3\*2 matrix b:

1 -2

2 3

1 3

Results:

8 13

4 9

#include <stdio.h>

#define ROW 2

#define COL 3

main()

{

int a[ROW][COL], b[COL][ROW], c[ROW][ROW], i, j,k;

printf("Input 2\*3 matrix a:\n");

for (i=0; i<ROW ;i++)//1

{

for (j=0; j<COL; j++)//1

{

scanf("%d", &a[i][j]);//1

}

}

printf("Input 3\*2 matrix b:\n");

for (i=0; i<COL; i++)//1

{

for (j=0; j<ROW; j++)//1

{

scanf("%d", &b[i][j]);//1

}

}

for (i=0; i<ROW; i++)//1

{

for (j=0; j<ROW; j++)//1

{

c[i][j] = 0 ;//1

for (k=0; k<COL; k++)//1

{

c[i][j] = c[i][j]+a[i][k]\*b[k][j] ;//2

}

}

}

printf("Results:\n");

for (i=0; i<ROW; i++)//1

{

for (j=0; j<ROW; j++) //1

{

printf("%6d", c[i][j]);//1

}

printf("\n") ;//1

}

}

7.编程将下列矩阵中的元素向右移动一列，最右一列移至第一列。

1 4 6

8 10 12

注意：移动过程中对同一个二维数组操作。

输入格式：无

输出格式："%6d"（换行使用"\n"）

运行结果：

6 1 4

12 8 10

#include <stdio.h>

#define ROW 2

#define COL 3

int main(void)

{

int a[ROW][COL] = {1, 4, 6, 8, 10, 12}; //1

int i, j, temp; //1

for (i = 0; i < ROW; i++) //1

{

temp = a[i][COL - 1]; //将当前行最后一列暂存

for (j = COL - 2; j >= 0; j--) //1

{

a[i][j + 1] = a[i][j]; //将当前行其他列后移

}

a[i][0] = temp; //将暂存数据赋予当前行0列

}

for (i = 0; i < ROW; i++) //1

{

for (j = 0; j < COL; j++) //1

{

printf("%6d", a[i][j]); //1

}

printf("\n"); //1

}

return 0;

}

8.在魔方阵中，所有的行、列和对角线都拥有相同的和。例如：

17 24 1 8 15

23 5 7 14 16 4 9 2

4 6 13 20 22 和 3 5 7

10 12 19 21 3 8 1 6

11 18 25 2 9

写一个程序读入一个二维整型数组并判断它是否为魔方矩阵。

\*\*输入格式要求："%d"

提示信息："请输入矩阵的阶数（<=10）：" "请输入矩阵：\n" "a[%d][%d]："

\*\*输出格式要求："该矩阵不是魔方阵！\n" "该矩阵为魔方矩阵！"

#include <stdio.h>

#include <stdlib.h>

#define N 10

int main()

{

int a[N][N];

int n, i, j, sum, oldsum;

printf("请输入矩阵的阶数（<=10）：");

scanf("%d", &n);

printf("请输入矩阵：\n");

for (i = 0; i < n; i++)

{

for (j = 0; j < n; j++)

{

printf("a[%d][%d]：", i, j);

scanf("%d", &a[i][j]);

}

}

//计算第一行的和

oldsum = 0;

for (i = 0; i < n; i++)

{

oldsum = oldsum + a[0][i];

}

//判断各行的和是否和第一行的和相等，不等则不是魔方矩阵

for (i = 0; i < n; i++)

{

sum = 0;

for (j = 0; j < n; j++)

{

sum = sum + a[i][j];

}

if (sum != oldsum)

{

printf("该矩阵不是魔方阵！\n");

exit(0);

}

}

//判断各列的和是否和第一行的和相等，不等则不是魔方矩阵

for (i = 0; i < n; i++)

{

sum = 0;

for (j = 0; j < n; j++)

{

sum = sum + a[j][i];

}

if (sum != oldsum)

{

printf("该矩阵不是魔方阵！\n");

exit(0);

}

}

//判断对角线的和是否和第一行的和相等，不等则不是魔方矩阵

sum = 0;

for (i = 0; i < n; i++)

{

sum = sum + a[i][i];

}

if (sum != oldsum)

{

printf("该矩阵不是魔方阵！\n");

exit(0);

}

sum = 0;

for (i = 0; i < n; i++)

{

sum = sum + a[i][n - i - 1];

}

if (sum != oldsum)

{

printf("该矩阵不是魔方阵！\n");

exit(0);

}

printf("该矩阵为魔方矩阵！");

return 0;

}

9.在给定的一组书名中，从键盘任意输入一个书名（书名可以有空格）。若找到，则打印该书名，否则打印"没找到"。找出其中错误并改正之。

#include <string.h>

#include <stdio.h>

int main()

{

int i, findFlag;

char x;

char str[][13]={ "Pascal","Basic","Fortran", "Java", "Visual C", "Visual Basic" };

printf("请输入一个字符串:");

gets(x);

while (i<6 & !findFlag)

{

if (x = str[i])

{

findFlag = 1;

}

i++;

}

if (findFlag)

{

printf("%s\n", x);

}

else

{

printf("没找到!\n");

}

return 0;

}

注意：对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

改错时不能改变程序原有的意图，但是可以改变变量的类型

#include <string.h>

#include <stdio.h>

main()

{

int i = 0, findFlag = 0;

char x[13];

char str[][13] = { "Pascal", "Basic", "Fortran", "Java", "Visual C", "Visual Basic" };

printf("请输入一个字符串:");

gets( x );

while (i < 6 && !findFlag)

{

if ( strcmp(x, str[i]) == 0 )

{

findFlag = 1;

}

i++;

}

if ( findFlag )

{

printf("%s\n", x);

}

else

{

printf("没找到!\n");

}

}

10.编程实现从键盘输入整型数组a[5][5]的元素的值，

然后计算并输出主对角线和副对角线之和。

即：a[0][0]+a[1][1]+a[2][2]+a[3][3]+a[4][4]

+a[0][4]+a[1][3]+a[3][1]+a[4][0]

要求:

(1)程序通过循环实现各项的累加。

(2)

\*\*输入提示信息为："Input a 5\*5 matrix\n"

\*\*输入矩阵中各个元素的格式为："%d"

\*\*输出格式为："sum=%5d\n"

#include <stdio.h>

main()

{

int a[5][5];

int i,j,sum;

printf("Input a 5\*5 matrix\n");

for(i=0;i<=4;i++) //1

{

for(j=0;j<=4;j++) //1

scanf("%d",&a[i][j]);//2

}

sum=0; //2

for(i=0;i<=4;i++) //1

{

for(j=0;j<=4;j++) //1

if(i==j || i+j==4) //2

sum=sum+a[i][j]; //2

}

printf("sum=%5d\n",sum); //2

}

11.1~9组成三个3位的平方数

讲1、2、3、4、5、6、7、8、9九个数字分成三组，每个数字只能用一次，即每组三个

数不许有重复数字，要求每组中的三位数都组成一个平方数。

\*\*输入格式要求：提示信息："The 3 squares with 3 different digits each are:\n"

\*\*输出格式要求："%d,%d,%d\n"

程序运行示例如下：

The 3 squares with 3 different digits each are:

169,256,784

169,529,784

196,256,784

196,529,784

324,784,961

361,529,784

#include <stdio.h>

main()

{

int a[20],num[20][3],b[10];

int i,j,k,m,n,t,flag;

printf("The 3 squares with 3 different digits each are:\n");

for(j=0,i=11;i<=31;i++)

if(i%10!=0)

{

k=i\*i;

num[j+1][0]=k/100;

num[j+1][1]=k/10%10;

num[j+1][2]=k%10;

if(!(num[j+1][0]==num[j+1][1]||num[j+1][0]==num[j+1][2]||num[j+1][1]==num[j+1][2]))

{

a[++j]=k;

}

}

for(i=1;i<=j-2;++i)

{

b[1]=num[i][0];

b[2]=num[i][1];

b[3]=num[i][2];

for(t=i+1;t<=j-1;++t)

{

b[4]=num[t][0];

b[5]=num[t][1];

b[6]=num[t][2];

for(flag=0,m=1;!flag&&m<=3;m++)

{

for(n=4;!flag&&n<6;n++)

if(b[m]==b[n]) flag=1;

}

if(!flag)

for(k=t+1;k<=j;++k)

{

b[7]=num[k][0];

b[8]=num[k][1];

b[9]=num[k][2];

for(flag=0,m=1;!flag&&m<=6;m++)

for(n=7;!flag&&n<=9;n++)

if(b[m]==b[n]) flag=1;

if(!flag)

printf("%d,%d,%d\n",a[i],a[t],a[k]);

}

}

}

}

12.打印魔方阵，所谓魔方阵是指这样的方阵，它的每一行、每一列和对角线之和均相等。例如，三阶魔方阵为：

8 1 6

3 5 7

4 9 2

要求打印出由1到n2的自然数构成的魔方阵（n为奇数）。

魔方阵中各数的排列规律如下：

（1）将1放在第一行中间一列；

（2）从2开始直到n×n，各数依次按下列规律存放：每一个数存放的行比前一个数的行数减1，列数加1；

（3）如果上一个数的行数为1，则下一个数的行数为n（指最下一行）；

（4）当上一个数的列数为n时，下一个数的列数应为1，行数减1；

（5）如果按上面规则确定的位置上已有数，或上一个数是第1行第n列时，则把下一个数放在上一个数的下面。

\*\*输入数据格式："%d"

\*\*输出格式要求："%3d"

程序的运行示例如下：

请输入n（0<n<=15，n是奇数）：5

矩阵阶数是：5

17 24 1 8 15

23 5 7 14 16

4 6 13 20 22

10 12 19 21 3

11 18 25 2 9

#include <stdio.h>

int main()

{

int a[16][16], i, j, k, p, m, n;

/\* 初始化 \*/

p = 1;

while (p == 1)

{

printf("请输入n（0<n<=15，n是奇数）：\n");

scanf("%d", &n);

if ((n != 0) && (n <= 15) && (n % 2 != 0))

{

printf("矩阵阶数是：%d\n", n);

p = 0;

}

}

for (i = 1; i <= n; i++)

{

for (j = 1; j <= n; j++)

{

a[i][j] = 0;

}

}

/\* 建立魔方 \*/

j = n / 2 + 1;

a[1][j] = 1;

for (k = 2; k <= n \* n; k++)

{

i = i - 1;

j = j + 1;

if ((i < 1) && (j > n))

{

i = i + 2;

j = j - 1;

}

else

{

if (i < 1)

{

i = n;

}

if (j > n)

{

j = 1;

}

}

if (a[i][j] == 0)

{

a[i][j] = k;

}

else

{

i = i + 2;

j = j - 1;

a[i][j] = k;

}

}

/\* 输出 \*/

for (i = 1; i <= n; i++)

{

for (j = 1; j <= n; j++)

{

printf("%3d", a[i][j]);

}

printf("\n");

}

return 0;

}

13.谁家孩子跑最慢

张、王、李三家各有三个小孩。一天，三家的九个孩子在一起比赛短

跑，规定不分年龄大小，跑第一得9分，跑第二得8分，一次类推。比赛结果

各家的总分相同，且这些孩子没有同时到达终点，也没有一家的两个或三

个孩子获得相连的名次。已知获第一名的是李家的孩子，获得第二的是王

家的孩子。求各家各个孩子的得分，获得最后一名的是谁家的孩子？

\*\*输入格式要求：提示信息："The last one arrived to end is a child from family Zhang.\n"

\*\*输出格式要求："%d "

程序运行示例如下：

7 5 3

8 6 1

9 4 2

The last one arrived to end is a child from family Wang.

#include<stdio.h>

int score[4][4];

main()

{

int i, j, k, who;

score[1][1] = 7;

score[2][1] = 8;

score[3][1] = 9;

for (i = 4; i < 6; i++)

for (j = 4; j < 7; j++)

for (k = 4; i != j && k < 7; k++)

if (k != 1 && k != j

&& 15 - i - score[1][1] != 15 - j - score[2][1]

&& 15 - i - score[1][1] != 15 - k - score[3][1]

&& 15 - j - score[2][1] != 15 - k - score[3][1])

{

score[1][2] = i;

score[1][3] = 15 - i - 7;

score[2][2] = j;

score[2][3] = 15 - j - 8;

score[3][2] = k;

score[3][3] = 15 - k - 9;

}

for (who = 0, i = 1; i <= 3; i++, printf("\n"))

for (j = 1; j <= 3; j++)

{

printf("%d ", score[i][j]);

if (score[i][j] == 1)

who = i;

}

if (who == 1)

printf("The last one arrived to end is a child from family Zhang.\n");

else if (who == 2)

printf("The last one arrived to end is a child from family Wang.\n");

else

printf("The last one arrived to end is a child from family Wang.\n");

}

字符串排序。调用Input函数输入10个字符串，调用Sort函数对这10个字符串按字典顺序排序，调用Print函数打印输出排序后的字符串。

Input函数原型：

void Input(char str[][80],int n);

Sort函数原型：

void Sort(char str[][80],int n);

Print函数原型：

void Print(char str[][80],int n);

\*\*\*输入数据格式：读入可包含有空格的字符串（字符串最大长度80）

\*\*\*输出数据格式: 一行输出一个字符串

#include<stdio.h>

#include<string.h>

void Input(char str[][80],int n);

void Sort(char str[][80],int n);

void Print(char str[][80],int n);

int main()

{

char str[10][80];

Input(str,10);

Sort(str,10);

Print(str,10);

return 0;

}

void Input(char str[][80],int n)

{

int i;

for(i=0;i<10;i++)

{

gets(str[i]);

}

}

void Sort(char str[][80],int n)

{

int i,j;

for(i=0;i<10;i++)

{

for(j=i+1;j<10;j++)

{

if(strcmp(str[i],str[j])>0)

{

char s[81];

strcpy(s,str[i]);

strcpy(str[i],str[j]);

strcpy(str[j],s);

}

}

}

}

void Print(char str[][80],int n)

{

int i;

for(i=0;i<10;i++)

{

printf("%s\n",str[i]);

}

}

用二维数组编程计算并输出n(n<20)行杨辉三角形。

\*\*输入提示信息要求："Input n(n<20):\n"

\*\*输入格式："%d"

\*\*输出要求："%4d"

每输出一行后就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include<stdio.h>

int main()

{

int a[20][20];

int n,i,j;

printf("Input n(n<20):\n");

scanf("%d",&n);

for(i=0;i<n;i++)

{

for(j=0;j<=i;j++)

{

if(j==0||j==i)

a[i][j]=1;

else

{

a[i][j]=a[i-1][j-1]+a[i-1][j];

}

}

}

for(i=0;i<n;i++)

{

for(j=0;j<=i;j++)

{

printf("%4d",a[i][j]);

}

putchar('\n');

}

return 0;

}

编程打印如下形式的杨辉三角形。打印的杨辉三角形的行数n（不超过15行）。要求由用户在主函数中通过键盘输入。请按照如下给定的函数原型进行编程。函数CaculateYH用于生成杨辉三角形，函数PrintYH用于打印杨辉三角形。给定的函数原型如下:

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

\*\*输入提示信息\*\*："Input n(n<15):\n"

\*\*输入数据格式\*\*："%d"

\*\*输出提示信息\*\*：无

\*\*输出数据格式\*\*："%5d"

杨辉三角形形式如下：

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

#include<stdio.h>

#define N 15

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

int main()

{

int a[15][15];

int n;

printf("Input n(n<15):\n");

scanf("%d",&n);

CaculateYH(a,n);

PrintYH(a,n);

return 0;

}

void CaculateYH(int a[][N], int n)

{

int i,j;

for(i=0;i<n;i++)

{

for(j=0;j<=i;j++)

{

if(j==0||j==i)

a[i][j]=1;

else

{

a[i][j]=a[i-1][j-1]+a[i-1][j];

}

}

}

}

void PrintYH(int a[][N], int n)

{

int i,j;

for(i=0;i<n;i++)

{

for(j=0;j<=i;j++)

{

printf("%5d",a[i][j]);

}

putchar('\n');

}

}

检验并打印幻方矩阵

幻方矩阵是指该矩阵中每一行、每一列、每一对角线上的元素之和都是相等的。从键盘输入一个5×5的矩阵并将其存入一个二维整型数组中，检验其是否为幻方矩阵，并将其按指定格式显示到屏幕上。

输入格式: "%d"

输出格式：

如果是幻方矩阵，输出提示信息: "It is a magic square!\n"

矩阵元素的输出： "%4d"（换行使用"\n"）

如果不是幻方矩阵，输出提示信息: "It is not a magic square!\n"

输入样例1：

17\_24\_1\_8\_15

23\_5\_7\_14\_16

4\_6\_13\_20\_22

10\_12\_19\_21\_3

11\_18\_25\_2\_9

输出样例1：

It is a magic square!

\*\*17\*\*24\*\*\*1\*\*\*8\*\*15

\*\*23\*\*\*5\*\*\*7\*\*14\*\*16

\*\*\*4\*\*\*6\*\*13\*\*20\*\*22

\*\*10\*\*12\*\*19\*\*21\*\*\*3

\*\*11\*\*18\*\*25\*\*\*2\*\*\*9

输入样例2：

1\_0\_1\_6\_1

3\_1\_1\_1\_1

1\_1\_1\_1\_2

1\_1\_1\_1\_1

9\_1\_7\_1\_1

输出样例2：

It is not a magic square!

(输人样例中“\_”代表空格，输出样例中“\*”代表空格)

#include  <stdio.h>

#define   N  5

int main(void)

{

    int  i, j;

    int  x[N][N];

    for (i = 0; i <= 4; i++)

        for (j = 0; j <= 4; j++)

        {

            scanf("%d", &x[i][j]);

        }

    int  rowSum[N], colSum[N], diagSum1, diagSum2;

    int  flag = 1;

    for (i = 0; i < N; i++)

    {

        rowSum[i] = 0;

        for (j = 0; j < N; j++)

        {

            rowSum[i] = rowSum[i] + x[i][j];

        }

    }

    for (j = 0; j < N; j++)

    {

        colSum[j] = 0;

        for (i = 0; i < N; i++)

        {

            colSum[j] = colSum[j] + x[i][j];

        }

    }

    diagSum1 = 0;

    for (j = 0; j < N; j++)

    {

        diagSum1 = diagSum1 + x[j][j];

    }

    diagSum2 = 0;

    for (j = 0; j < N; j++)

    {

        diagSum2 = diagSum2 + x[j][N - 1 - j];

    }

    if (diagSum1 != diagSum2)

    {

        flag = 0;

    }

    else

    {

        for (i = 0; i < N; i++)

        {

            if ((rowSum[i] != diagSum1) || (colSum[i] != diagSum1))

                flag = 0;

        }

    }

    if (flag)

    {

        printf("It is a magic square!\n");

        for (i = 0; i < N; i++)

        {

            for (j = 0; j < N; j++)

            {

                printf("%4d", x[i][j]);

            }

            printf("\n");

        }

    }

    else

    {

        printf("It is not a magic square!\n");

    }

    return 0;

}

输入n×n阶矩阵(n最大为10)，按如下函数原型

/\* 函数功能： 计算n×n矩阵中两条对角线上的元素之和 \*/

int AddDiagonal(int a[N][N], int n);

用函数编程计算并输出其两条对角线上的各元素之和。

\*\*输入提示信息："Input n:" "Input %d\*%d matrix:\n"

\*\*输入格式要求："%d"

\*\*输出格式要求："sum = %d\n"

程序运行示例如下：

Input n:5

Input 5\*5 matrix:

1 2 3 4 5

2 3 4 5 6

3 4 5 6 7

4 5 6 7 8

5 6 7 8 9

sum = 45

注：不允许使用goto语句

#include<stdio.h>

#define N 10

int AddDiagonal(int a[N][N], int n)

{

int i,sum=0;

for(i=0;i<n;i++)

{

sum+=a[i][i];

}

for(i=n-1;i>=0;i--)

{

sum+=a[i][n-i-1];

}

if(n%2)

{

sum-=a[n/2][n/2];

}

return sum;

}

int main()

{

int a[10][10],n;

printf("Input n:");

scanf("%d",&n);

int i,j;

printf("Input %d\*%d matrix:\n",n,n);

for(i=0;i<n;i++)

{

for(j=0;j<n;j++)

{

scanf("%d",&a[i][j]);

}

}

int sum;

sum=AddDiagonal(a,n);

printf("sum = %d\n" ,sum);

return 0;

}

输入m×n阶矩阵A和B，用函数编程计算并输出A与B之和。

\*\*输入格式要求："%d,%d" 提示信息："Input m, n:" "Input %d\*%d matrix a:\n" "Input %d\*%d matrix b:\n"

\*\*输出格式要求："Results:\n" "%6d"

程序运行示例如下：

Input m, n:2,3

Input 2\*3 matrix a:

1 2 3

4 5 6

Input 2\*3 matrix b:

11 12 13

14 15 16

Results:

12 14 16

18 20 22

#include<stdio.h>

int main()

{

int m,n;

int a[20][20],b[20][20];

printf("Input m, n:");

scanf("%d,%d",&m,&n);

printf("Input %d\*%d matrix a:\n",m,n);

int i,j;

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

scanf("%d",&a[i][j]);

}

}

printf("Input %d\*%d matrix b:\n",m,n);

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

scanf("%d",&b[i][j]);

}

}

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

a[i][j]+=b[i][j];

}

}

printf("Results:\n");

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

printf("%6d",a[i][j]);

}

putchar('\n');

}

return 0;

}

下面程序用于从键盘输入3\*4矩阵的元素，通过调用函数FindMax，求出这3\*4矩阵元素中的最大值及其所在行列位置，然后输出这个最大值及其所在行列位置。（找出其中的4处错误，并改正之）

#include <stdio.h>

**int** FindMax(**int** x[3][4],**int** \*pRow,**int** \*pCol);

main()

{

**int** a[3][4], max, i, j, row, col;

**for** (i=0; i<3; i++)

    {

**for** (j=0; j<4; j++)

        {

**scanf**("%d",a[i][j]);

        }

    }

    max = FindMax(a,&row,&col);

**printf**("max=%d,row=%d,col=%d\n",max,row,col);

}

**int** FindMax(**int** x[3][4],**int** \*pRow,**int** \*pCol)

{

**int** max, i, j;

    max = x[0][0];

**for** (i=0; i<3; i++)

    {

**for** (j=0; j<4; j++)

        {

**if** (x[i][j] > max);

            {

                max = x[i][j];

                \*pRow = i;

                \*pCol = j;

            }

        }

    }

**return** max;

}

#include <stdio.h>

int FindMax(int x[3][4],int \*pRow,int \*pCol);

main()

{

int a[3][4], max, i, j, row=0, col=0;

for (i=0; i<3; i++)

{

for (j=0; j<4; j++)

{

scanf("%d",&a[i][j]);

}

}

max = FindMax(a,&row,&col);

printf("max=%d,row=%d,col=%d\n",max,row,col);

}

int FindMax(int x[3][4],int \*pRow,int \*pCol)

{

int max, i, j;

max = x[0][0];

for (i=0; i<3; i++)

{

for (j=0; j<4; j++)

{

if (x[i][j] > max)

{

max = x[i][j];

\*pRow = i;

\*pCol = j;

}

}

}

return max;

}

用数组实现7阶魔方矩阵。所谓的N阶魔方矩阵是指把1～N\*N的自然数按一定方法排列成N\*N的矩阵，使得：任意行、任意列以及两个对角线上的数之和都相等（N为奇数）。例如下面的5阶魔方矩阵，任意行、任意列以及两个对角线上的数之和都为65。

5阶魔方如下：

17 24 1 8 15

23 5 7 14 16

4 6 13 20 22

10 12 19 21 3

11 18 25 2 9

奇数阶魔方矩阵的算法如下：

第1步：将1放入第一行的正中处。

第2步：按如下的方法依次将第i个数（i从2到N\*N）放到合适的位置上。

λ 如果第i-1个数的右上位置没有放数，则将第i个数放到前一个数的右上位置。

λ 如果第i-1个数的右上位置已经有数，则将第i个数放到第i-1个数的下一行，列数相同的位置。

\*\*输出格式要求："\n%d阶魔方矩阵如下：\n" 输出7个"===="后换行 "%4d" 输出7个"===="后换行

程序运行结果如下：

7阶魔方矩阵如下：

============================

30 39 48 1 10 19 28

38 47 7 9 18 27 29

46 6 8 17 26 35 37

5 14 16 25 34 36 45

13 15 24 33 42 44 4

21 23 32 41 43 3 12

22 31 40 49 2 11 20

============================

#include <stdio.h>

#define N 7

**int** main()

{

**int** matrix[N][N];

**int** row, col;

**int** i, m, n;

    //0. 初始化数组元素全为0

**for** (row = 0; row < N; row++)

    {

**for** (col = 0; col < N; col++)

        {

            matrix[row][col] = 0;

        }

    }

    //1. 定位1的初始位置

    row = 0;

    col = (N - 1) / 2;

    matrix[row][col] = 1;

    //2. 将2..N\*N每个数插入到相应位置

**for** (i = 2; i <= N \* N; i++)

    {

**int** r, c;

        r = row;

        c = col;

        row = (row + N - 1) % N;   //行列坐标的计算，算法难点

        col = (col + N + 1) % N;

**if** (0 == matrix[row][col])  //该处有数则冲突，处理到前一个数的下一列

            matrix[row][col] = i;   //无冲突则插入该数。

**else**

        {

            r = (r + 1) % N;

            matrix[r][c] = i;

            row = r;

            col = c;

        }

    }

    //3. 输出魔方矩阵

**printf**("\n%d阶魔方矩阵如下：\n", N);

**for** (m = 0; m <= N; m++)

    {

**printf**("====");

    }

**printf**("\n");

**for** (m = 0; m < N; m++)

    {

**for** (n = 0; n < N; n++)

**printf**("%4d", matrix[m][n]);

**printf**("\n");

    }

**for** (m = 0; m <= N; m++)

    {

**printf**("====");

    }

**printf**("\n");

**return** 0;

}

杨辉三角形

编程打印具有如下形式的杨辉三角形（它的特点是左右两边全是1，从第二行起，中间的每一个数是上一行里相邻两个数之和），其中输出数据的行数n从键盘输入，并且n<=10。

程序运行结果示例1：

Input n (n<=10):

5↙

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

程序运行结果示例2：

Input n (n<=10):

7↙

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

输入格式: "%d"

输出格式：

输入提示信息："input n (n<=10):\n"

输出数据格式："%4d"

数据换行： "\n"

#include<stdio.h>

int main()

{

int a[20][20];

int n,i,j;

printf("input n (n<=10):\n");

scanf("%d",&n);

for(i=0;i<n;i++)

{

for(j=0;j<=i;j++)

{

if(j==0||j==i)

a[i][j]=1;

else

{

a[i][j]=a[i-1][j-1]+a[i-1][j];

}

}

}

for(i=0;i<n;i++)

{

for(j=0;j<=i;j++)

{

printf("%4d",a[i][j]);

}

putchar('\n');

}

return 0;

}

写一个n×n(n<10)矩阵的转置程序，输出其原矩阵的值和转置以后的结果。

（注：不输入数组元素，用a[i][j] = n\* i + j + 1求得）

\*\*输入提示："please input n:"

\*\*输入格式要求："%d"

\*\*输出提示信息："转置前：\n"

\*\*输出格式要求："%-4d"

\*\*输出提示信息："转置后：\n"

\*\*输出格式要求："%-4d""

程序运行结果如下：

please input n:5

转置前：

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

16 17 18 19 20

21 22 23 24 25

转置后：

1 6 11 16 21

2 7 12 17 22

3 8 13 18 23

4 9 14 19 24

5 10 15 20 25

#include <stdio.h>

#define ROW 10

#define COLUMN 10

**int** main()

{

**int** a[ROW][COLUMN];

**int** i, j, k,n;

**printf**("please input n:");

**scanf**("%d",&n);

**for** (i = 0; i < n ; i++)

**for** (j = 0; j < n ; j++ )

            a[i][j] = n \* i + j + 1;  /\*对矩阵赋值\*/

**printf**("转置前：\n");

**for** (i = 0; i < n ; i++)

    {

**for** (j = 0; j < n ; j++ )

**printf**("%-4d", a[i][j]);         /\*输出转置前的矩阵\*/

**printf**("\n");

    }

**for** (i = 0; i < n ; i++)

    {

**for** (j = 0; j < i ; j++ )

        {

            k = a[i][j];

            a[i][j] = a[j][i];

            a[j][i] = k;

        }

    }

**printf**("转置后：\n");

**for** (i = 0; i < n ; i++ )

    {

**for** (j = 0; j < n ; j++ )

**printf**("%-4d", a[i][j]);      /\*输出后转置后的矩阵\*/

**printf**("\n");

    }

}

编写程序将如下的一个数组a向右旋转90度，构成新的数组b，并打印出来。

a =

1 2

3 4

5 6

7 8

旋转后

b =

7 5 3 1

8 6 4 2

\*\*输出提示信息："Array a:\n" "Array b:\n"

\*\*输出格式要求："%4d"

程序运行示例如下：

Array a:

1 2

3 4

5 6

7 8

Array b:

7 5 3 1

8 6 4 2

#include <stdio.h>

#define M 4

#define N 2

main()

{

**int** a[M][N] = {{1, 2}, {3, 4}, {5, 6}, {7, 8}};

**int** b[N][M], i, j;

**printf**("Array a:\n");

**for** (i = 0; i < M; i++)

    {

**for** (j = 0; j < N; j++)

        {

**printf**("%4d", a[i][j]);

            b[j][M - 1 - i] = a[i][j];

        }

**printf**("\n");

    }

**printf**("Array b:\n");

**for** (i = 0; i < N; i++)

    {

**for** (j = 0; j < M; j++)

        {

**printf**("%4d", b[i][j]);

        }

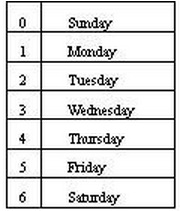
**printf**("\n");

    }

}

星期查找

任意输入英文的星期几，通过查找如图所示的星期表，输出其对应的数字，若查到表尾，仍未找到，则输出错误提示信息。



提示：用一个二维字符数组weekDay来存放如图所示的星期表的内容（字符串）。输入待查找的字符串，然后在星期表中顺序查找与输入字符串相匹配的字符串。找到的字符串在星期表数组中的第一维下标（行号）即为题目所求。

程序运行结果示例1：

Please enter a string:

Friday↙

Friday is 5

程序运行结果示例2：

Please enter a string:

Fruday↙

Not found!

**输入格式:**字符串输入采用gets()函数

**输出格式：**

输入提示信息："Please enter a string:\n"

找到了，输出："%s is %d\n"

没找到，输出："Not found!\n"

#include  <stdio.h>

#include  <string.h>

#define   WEEKDAYS  7             /\*每星期天数\*/

#define   MAX\_STR\_LEN  10         /\*字符串最大长度\*/

**int** main()

{

**int**    i, pos;

**int**    findFlag = 0;          /\* 置找到标志为假 \*/

**char**   x[MAX\_STR\_LEN];

**char**   weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",

                                     "Wednesday", "Thursday", "Friday",                                  "Saturday"

                                    };

**printf**("Please enter a string:\n");

**gets**(x);

    /\*scanf("%s", x);                /\* 输入待查找的字符串 \*/

**for** (i = 0; i < WEEKDAYS && !findFlag; i++)

    {

**if** (**strcmp**(x, weekDay[i]) == 0)

        {

            pos = i;          /\*记录找到的位置\*/

            findFlag = 1;    /\*若找到，则置找到标志为真，退出循环\*/

        }

    }

**if** (findFlag)        /\*找到标志为真，说明找到\*/

**printf**("%s is %d\n", x, pos);

**else**                  /\*找到标志为假，说明未找到\*/

**printf**("Not found!\n");

**return** 0;

}

将5\*5的矩阵A转置（即行变列，列变行），并将行列互换即转置

前、后的矩阵分别打印出来。

下面程序中存在比较隐蔽的错误，

请通过分析和调试程序，发现并改正程序中的错误。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才得满分。

|  |
| --- |
| #include (stdio.h)  main()  {  **int**  a[5][5]={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21}, i, j;    **printf**(“before transposition\n”);  **for** (i=0; i<=4; i++)  **for** (j=0; j<=4; j++)  **printf**("%4d%s", a[i][j],j-4?' ':'\n' );  **for** (i=0; i<=4; i++)  **for** (j=0; j<=4; j++)         {              t=a[i][j];              a[i][j]=a[j][i];              a[j][i]=t;         }  **printf**(“after transposition\n”);  **for** (i=0; i<=4; i++);  **for** (j=0; j<=4; j++)  **printf**("%4d%s", a[i][j],j-4?' ':'\n' );  } |

|  |
| --- |
| #include <stdio.h>  main()  {  **int**  a[5][5]={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21}, i, j,t;  **printf**("before transposition\n");  **for** (i=0; i<=4; i++)    {  **for** (j=0; j<=4; j++)  **printf**("%4d%c", a[i][j],j-4?' ':'\n' );    }    **for** (i=1; i<=4; i++)    {  **for** (j=0; j<=i-1; j++)       {        t=a[i][j];            a[i][j]=a[j][i];            a[j][i]=t;       }    }  **printf**("after transposition\n");  **for** (i=0; i<=4; i++)    {  **for** (j=0; j<=4; j++)       {  **printf**("%4d%c", a[i][j],j-4?' ':'\n' );        }     }  } |

使用二维字符数组编程，任意输入英文的星期几，通过查找星期表，输出其对应的数字，若查到表尾，仍未找到，则输出"Not found!"的错误提示信息。

下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

|  |
| --- |
| #include  <stdio.h>  #define   WEEKDAYS  7  #define   MAX\_STR\_LEN  10  main()  {  **int** i, pos;  **int** findFlag;  **char** x[MAX\_STR\_LEN];  **char** weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",                                     "Wednesday", "Thursday", "Friday", "Saturday"                                    };  **printf**("Please enter a string:\n");  **scanf**("%s", &x);  **for** (i = 0; i <= WEEKDAYS || findFlag; i++)      {  **if** (x[i] = weekDay[i])          {              pos = i;              findFlag = 1;          }      }  **if** (findFlag)      {  **printf**("%s is %d\n", x, pos);      }  **else**      {  **printf**("Not found!\n");      }  } |

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

允许修改和增加语句，不允许删除语句。

当且仅当全部错误都修改正确才给分，部分错误修改正确不给分。

改错时不能改变程序原有的意图，不能改变函数原型。

#include  <stdio.h>

#include  <string.h>

#define   WEEKDAYS  7             /\*每星期天数\*/

#define   MAX\_STR\_LEN  10         /\*字符串最大长度\*/

**int** main()

{

**int**    i, pos;

**int**    findFlag = 0;

**char**   x[MAX\_STR\_LEN];

**char**   weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};

**printf**("Please enter a string:\n");

**scanf**("%s", x);

**for** (i = 0; i < WEEKDAYS && !findFlag; i++)

    {

**if** (**strcmp**(x, weekDay[i]) == 0)

        {

            pos = i;

            findFlag = 1;

        }

    }

**if** (findFlag)

**printf**("%s is %d\n", x, pos);

**else**

**printf**("Not found!\n");

**return** 0;

}

写一个程序输入一个4乘5的数组，查找所有小于0的数组元素及其所在的行列下标。

\*\*输入格式要求："%lf" 提示信息："请输入一个4乘5的矩阵：\n" "请输入a[%d][%d]："

\*\*输出格式要求："第%d行%d列的元素%.2f小于0\n"

#include <stdio.h>

**int** main()

{

**double** a[4][5];

**int** i, j, row, col;

**printf**("请输入一个4乘5的矩阵：\n");

**for** (i = 0; i < 4; i++)

    {

**for** (j = 0; j < 5; j++)

        {

**printf**("请输入a[%d][%d]：", i, j);

**scanf**("%lf", &a[i][j]);

        }

    }

**for** (i = 0; i < 4; i++)

    {

**for** (j = 0; j < 5; j++)

        {

**if** (a[i][j] < 0.0)

            {

**printf**("第%d行%d列的元素%.2f小于0\n", i, j, a[i][j]);

            }

        }

    }

**return** 0;

}

有一篇文章，共有三行文字，每行有80个字符。要求分别统计出其中英文大写字母、小写字母、数字、空格、以及其它字符的个数。

程序的运行示例如下：

请输入第0行：

helloworld

请输入第1行：

12345gogogo

请输入第2行：

end bye bye.

helloworld

12345gogogo

end bye bye.

大写字母数：0

小写字母数：25

数字个数 ：5

空格个数 ：2

其它字符 ：1

\*\*\*输入提示信息\*\*\*："\n请输入第%d行：\n"

\*\*\*输入数据格式\*\*\*：使用gets()

\*\*\*输出数据格式\*\*\*："%s\n"

"大写字母数：%d\n"

"小写字母数：%d\n"

"数字个数 ：%d\n"

"空格个数 ：%d\n"

"其它字符 ：%d\n"

    #include <stdio.h>

**int** main()

{

**int** i, j, uppn, lown, dign, span, othn;

**char** text[3][80];

    uppn = lown = dign = span = othn = 0;

**for** (i = 0; i < 3; i++)

    {

**printf**("\n请输入第%d行：\n", i);

**gets**(text[i]);

**for** (j = 0; j < 80 && text[i][j] != '\0'; j++)

        {

**if** (text[i][j] >= 'A' && text[i][j] <= 'Z')

            {

                uppn += 1;

            }

**else** **if** (text[i][j] >= 'a' && text[i][j] <= 'z')

            {

                lown += 1;

            }

**else** **if** (text[i][j] >= '0' && text[i][j] <= '9')

            {

                dign += 1;

            }

**else** **if** (text[i][j] == ' ')

            {

                span += 1;

            }

**else**

            {

                othn += 1;

            }

        }

    }

**for** (i = 0; i < 3; i++)

    {

**printf**("%s\n", text[i]);

    }

**printf**("大写字母数：%d\n", uppn);

**printf**("小写字母数：%d\n", lown);

**printf**("数字个数  ：%d\n", dign);

**printf**("空格个数  ：%d\n", span);

**printf**("其它字符  ：%d\n", othn);

**return** 0;

}

给定某年某月某日，将其转换成这一年的第几天并输出。

\*\*输入格式要求："%d%d%d"

\*\*输出格式要求："%d"

程序运行示例如下：

2012 11 8

313

|  |
| --- |
| #include <stdio.h>  main()  {  **static** **int** day\_tab[2][13]={          {0,31,28,31,30,31,30,31,31,30,31,30,31},          {0,31,29,31,30,31,30,31,31,30,31,30,31}};  **int** y,m,d;  **scanf**("%d%d%d",&y,&m,&d);  **printf**("%d\n",day\_of\_year(day\_tab,y,m,d));  }  day\_of\_year(day\_tab,year,month,day)  **int** day\_tab[][13];  **int** year,month,day;  {  **int** i,j;      i=(year%4==0&&year%100!=0)||year%400==0;  **for**(j=1;j<month;j++)          day+=day\_tab[i][j];  **return** day;  } |

输入一个正整数n (1＜n≤6)，根据下式生成1个n\*n的方阵（二维数组存放）：

a[i][j] = i \* n + j + 1（0≤i≤n-1，0≤j≤n-1）

然后将该方阵转置（行列互换）后输出。

注意：原方阵和转置方阵用同一个二维数组存放。

\*\*输入格式要求："%d" 提示信息："Enter n: "

\*\*输出格式要求："%4d"

对矩阵的输出要求：列与列之间为%4d，行与行之间为\n

程序运行示例如下：

Enter n: 3

1 4 7

2 5 8

3 6 9

|  |
| --- |
| #include <stdio.h>    **int** main(**void**)  {  **int** i, j, n, temp;  **int** a[6][6];        /\* 给二维数组赋值 \*/  **printf**("Enter n: ");  **scanf**("%d", &n);  **for** (i = 0; i < n; i++)             /\* 行下标是外循环的循环变量 \*/  **for** (j = 0; j < n; j++)      /\* 列下标是内循环的循环变量 \*/              a[i][j] = i \* n + j + 1;    /\* 给数组元素赋值 \*/        /\* 行列互换\*/  **for** (i = 0; i < n; i++)  **for** (j = 0; j < n; j++)  **if** (i <= j)              /\* 只遍历上三角阵 \*/              {                  temp = a[i][j];     /\* 以下3句交换 a[i][j] 和 a[j][i] \*/                  a[i][j] = a[j][i];                  a[j][i] = temp;              }        /\* 按矩阵的形式输出a \*/  **for** (i = 0; i < n; i++)          /\* 针对所有行的循环 \*/      {  **for** (j = 0; j < n; j++)      /\* 输出第i行的所有元素 \*/  **printf**("%4d", a[i][j]);  **printf**("\n");           /\* 换行 \*/      }    **return** 0;  } |

输入n×n阶的矩阵(用二维数组存放，n<=10)，编程计算：（1）两条对角线上的各元素之和。（2）两条对角线上行、列下标均为偶数的各元素之积。

(注意：如果n为奇数，则中心点的元素只计算一次）

输入提示信息：

"Input n:"

"Input %d\*%d matrix:\n"

输入格式："%d"

输出格式："sum = %d\nproduct = %ld\n"

运行示例：

Input n:3↙

Input 3\*3 matrix:

1 2 3

2 3 4

3 4 5

输出：

sum = 15

product = 45

|  |
| --- |
| #include  <stdio.h>  #define ARR\_SIZE 10 //1  **int** main(**void**)  {  **int**         a[ARR\_SIZE][ARR\_SIZE], i, j, n, sum = 0; //1  **long**    product = 1;  **printf**("Input n:");  **scanf**("%d", &n) ;  **printf**("Input %d\*%d matrix:\n", n, n); //1  **for** (i = 0; i < n; i++)      {  **for** (j = 0; j < n; j++)          {  **scanf**("%d", &a[i][j]);          }      }  **for** (i = 0; i < n; i++) //1      {  **for** (j = 0; j < n; j++) //1          {  **if** (i == j || i + j == n - 1) //1                  sum += a[i][j]; //1  **if** ((i == j || i + j == n - 1) && i % 2 == 0 && j % 2 == 0) //1                  product \*= a[i][j]; //1          }      }  **printf**("sum = %d\nproduct = %ld\n", sum, product);//1  **return** 0;  } |

#include <stdio.h>

void Convert(int a[][4], int b[][3]);

main()

{

int i, j;

int a[3][4], b[4][3];//2

for (i = 0; i < 3; i++)//1

{

for (j = 0; j < 4; j++)//1

{

scanf("%d", &a[i][j]);

}

}

Convert(a, b);//1

for (i = 0; i < 4; i++)//1

{

for (j = 0; j < 3; j++)//1

printf(" %4d", b[i][j]);

printf("\n");//1

}

}

void Convert(int a[][4], int b[][3])

{

int i, j;

for (i = 0; i < 4; i++)//1

{

for (j = 0; j < 3; j++)//1

{

b[i][j] = a[j][i];//1

}

}

}

在主函数中输入一个3X4的整型矩阵，求其转置矩阵并存放在另一个二维数组中 在主函数中将转置后的新数组输出。（注：转置矩阵 把矩阵的行换成相应的列，得到的新矩阵称为其转置矩阵）

利用以下函数完成转置功能。

void Convert（int a[][4], int b[][3]);

\*\*输入格式要求： "%d"

\*\*输出格式要求：" %4d"

#include <stdio.h>

#define ROW 6

#define COL 6

/\* 函数功能：计算矩阵相乘之积，结果存于二维数组c中 \*/

void MultiplyMatrix(int a[ROW][COL], int b[COL][ROW], int c[ROW][ROW], int m, int n)//1

{

int i, j, k;

for (i = 0; i < m; i++)//1

{

for (j = 0; j < m; j++)//1

{

c[i][j] = 0;//1

for (k = 0; k < n; k++)//1

{

c[i][j] = c[i][j] + a[i][k] \* b[k][j];//1

}

}

}

}

/\* 函数功能：输出矩阵a中的元素 \*/

void PrintMatrix(int a[ROW][ROW], int m)//1

{

int i , j ;

for (i = 0; i < m; i++)

{

for (j = 0; j < m; j++)

{

printf("%6d", a[i][j]);

}

printf("\n");

}

}

int main()

{

int a[ROW][COL], b[COL][ROW], c[ROW][ROW], i, j, m, n, ret;

do

{

printf("Input m, n:");

ret = scanf("%d,%d", &m, &n);//1

if (ret != 2) while (getchar() != '\n');//1

}while (m > 6 || n > 6 || ret != 2);//e1

printf("Input %d\*%d matrix a:\n", m, n);

for (i = 0; i < m ; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &a[i][j]);//1

}

}

printf("Input %d\*%d matrix b:\n", m, n);

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

scanf("%d", &b[i][j]);//1

}

}

MultiplyMatrix(a, b, c, m, n);//1

printf("Results:\n");

PrintMatrix(c, m);//1

return 0;

}

利用矩阵相乘公式cij= aik\*bkj，编程计算并输出m×n阶矩阵A和n×m阶矩阵B之积。其中，m和n从键盘输入，m和n的值不超过6，否则提示用户重新输入，如果输入非法字符也提示用户重新输入。要求按如下函数原型编写程序：

/\* 函数功能：计算m×n阶矩阵A和n×m阶矩阵B之积，结果存于二维数组c中 \*/

void MultiplyMatrix(int a[ROW][COL], int b[COL][ROW], int c[ROW][ROW], int m, int n);

/\* 函数功能：输出m×m阶矩阵a中的元素 \*/

void PrintMatrix(int a[ROW][ROW], int m);

\*\*输入m和n的输入提示信息："Input m, n:"

\*\*输入m和n的输入格式："%d,%d"

\*\*输入矩阵元素的输入提示信息："Input %d\*%d matrix a:\n"

\*\*输入矩阵元素的输入格式： "%d"

\*\*输出提示信息："Results:\n"

\*\*输出格式："%6d"

程序运行示例：

Input m, n:7,a↙

Input m, n:7,8↙

Input m, n:2,3↙

Input 2\*3 matrix a:

1 2 3↙

4 5 6↙

Input 2\*3 matrix b:

1 2↙

3 4↙

5 6↙

Results:

22 28

49 64

注意：

（1）不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

（2）用标准C语言编程，所有变量必须在第一条可执行语句前定义。

#include <stdio.h>

#define M 10

#define N 10

void Transpose(int (\*a)[N], int (\*at)[M], int m, int n);

void InputMatrix(int (\*a)[N], int m, int n);

void PrintMatrix(int (\*at)[M], int n, int m);

int main()

{

int s[M][N], st[N][M], m, n;

printf("Input m, n:");

scanf("%d,%d", &m, &n);

InputMatrix(s, m, n);

Transpose(s, st, m, n);

printf("The transposed matrix is:\n");

PrintMatrix(st, n, m);

return 0;

}

/\* 函数功能：计算m\*n矩阵a的转置矩阵at \*/

void Transpose(int (\*a)[N], int (\*at)[M], int m, int n)

{

int i, j;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

\*(\*(at + j) + i) = \*(\*(a + i) + j);

}

}

}

/\* 函数功能：输入m\*n矩阵a的值 \*/

void InputMatrix(int (\*a)[N], int m, int n)

{

int i, j;

printf("Input %d\*%d matrix:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", \*(a + i) + j);

}

}

}

/\* 函数功能：输出n\*m矩阵at的值 \*/

void PrintMatrix(int (\*at)[M], int n, int m)

{

int i, j;

for (i = 0; i < n; i++)

{

for (j = 0; j < m; j++)

{

printf("%d\t", \*(\*(at + i) + j));

}

printf("\n");

}

}

用指向一维数组的指针变量即二维数组的行指针作为函数参数，实现矩阵转置。按如下函数原型编程计算并输出m×n阶矩阵的转置矩阵。其中，m和n的值由用户从键盘输入。已知m和n的值都不超过10。

void Transpose(int (\*a)[N], int (\*at)[M], int m, int n);

void InputMatrix(int (\*a)[N], int m, int n);

void PrintMatrix(int (\*at)[M], int n, int m);

输入提示信息："Input m, n:"

输入格式："%d,%d"

输入提示信息："Input %d\*%d matrix:\n"

输出提示信息和格式："The transposed matrix is:\n"

输出格式："%d\t"

程序测试用例：

Input m, n:3,4

Input 3\*4 matrix:

1 2 3 4

5 6 7 8

9 10 11 12

The transposed matrix is:

1 5 9

2 6 10

3 7 11

4 8 12

#include <stdio.h>

#define M 3

#define N 4

/\* 函数功能：计算矩阵相乘之积，结果存于二维数组c中 \*/

void MultiplyMatrix(int a[M][N], int b[N][M], int c[M][M])

{

int i, j, k;

for (i = 0; i < M; i++)

{

for (j = 0; j < M; j++)

{

c[i][j] = 0;

for (k = 0; k < N; k++)

{

c[i][j] = c[i][j] + a[i][k] \* b[k][j];

}

}

}

}

/\* 函数功能：输出矩阵a中的元素 \*/

void InputMatrixA(int a[M][N])

{

int i, j;

printf("Input %d\*%d matrix a:\n", M, N);

for (i = 0; i < M ; i++)

{

for (j = 0; j < N; j++)

{

printf("a[%d,%d]:", i, j);

scanf("%d", &a[i][j]);

}

}

}

void InputMatrixB(int b[N][M])

{

int i, j;

printf("Input %d\*%d matrix b:\n", N, M);

for (i = 0; i < N ; i++)

{

for (j = 0; j < M; j++)

{

printf("b[%d,%d]:", i, j);

scanf("%d", &b[i][j]);

}

}

}

void OutputMatrix(int a[M][M])

{

int i , j ;

printf("Results:\n");

for (i = 0; i < M; i++)

{

for (j = 0; j < M; j++)

{

printf("%6d", a[i][j]);

}

printf("\n");

}

}

int main()

{

int a[M][N], b[N][M], c[M][M];

InputMatrixA(a);

InputMatrixB(b);

MultiplyMatrix(a, b, c);

OutputMatrix(c);

return 0;

}

实验五（2016春刘秉权C语言课）：利用矩阵相乘公式，编程计算M\*N阶矩阵和N\*M阶矩阵B之积C(M\*M阶矩阵)。

要求：

1.输入输出格式形如：

Input 3\*4 matrix a:(输出)

a[0,0]:(输出)11（输入）

a[0,1]:2

...

a[2,2]:7

a[2,3]:9

Input 4\*3 matrix b:(输出)

b[0,0]:23(输入)

b[0,1]:21

...

b[3,1]:7

b[3,2]:54

Results:(输出)

1643 2063 1063(整数输出6位)

4144 5205 5394

1014 1106 1923

2.用宏定义形式分别定义M、N的值为3、4。

3.函数原型分别定义成：

void MultiplyMatrix(int a[M][N], int b[N][M], int c[M][M]);/\* 函数功能：计算矩阵相乘之积，结果存于二维数组c中 \*/

void InputMatrixA(int a[M][N]);/\* 函数功能：输入矩阵a中的元素 \*/

void InputMatrixB(int b[N][M]);/\* 函数功能：输入矩阵b中的元素 \*/

void OutputMatrix(int a[M][M]);/\* 函数功能：输出矩阵a中的元素 \*/

4.主函数定义成：

int main()

{

int a[M][N], b[N][M], c[M][M];

InputMatrixA(a);

InputMatrixB(b);

MultiplyMatrix(a,b,c);

OutputMatrix(c);

return 0;

}

#include <stdio.h>

#define N 10

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

int main()

{

int s[N][N], n;

printf("Input n:\n");

scanf("%d", &n);//1

InputMatrix(s, n);//1

Transpose(s, n);//1

printf("The transposed matrix is:\n");

PrintMatrix(s, n);//1

return 0;

}

void Transpose(int a[][N], int n)

{

int i, j, temp;

for (i = 0; i < n; i++) //1

{

for (j = i; j < n; j++)//1

{

temp = a[i][j];//1

a[i][j] = a[j][i];//1

a[j][i] = temp;//1

}

}

}

void InputMatrix(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++)//1

{

for (j = 0; j < n; j++)//1

{

scanf("%d", &a[i][j]); //1

}

}

}

void PrintMatrix(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j < n; j++) //1

{

printf("%5d", a[i][j]); //1

}

printf("\n"); //1

}

}

按如下函数原型，用二维数组作为函数参数，编程计算并输出n×n阶矩阵的转置矩阵。其中，n由用户从键盘输入。已知n值不超过10。

void Transpose(int a[][N], int n);

void InputMatrix(int a[][N], int n);

void PrintMatrix(int a[][N], int n);

\*\*输入提示信息："Input n:\n"

\*\*输入格式："%d"

\*\*输出提示信息："The transposed matrix is:\n"

\*\*输出要求："%5d"

输出矩阵时，每输出一行就换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

#include <stdio.h>

#define M 3

#define N 4

void Input(int (\*p)[4], int m, int n); //1

int FindMax(int (\*p)[4], int m, int n, int \*pRow, int \*pCol); //1

main()

{

int score[M][N], maxScore, row, col;

printf("Please input your data:\n");

Input(score, M, N); //1

maxScore = FindMax(score, M, N, &row, &col); //2

printf("The maximum is %d, which is in row %d, colum %d\n", maxScore, row, col);

}

void Input(int (\*p)[4], int m, int n)

{

int i, j;

for (i = 0; i < m; i++) //1

{

for (j = 0; j < n; j++) //1

scanf("%d", (p[i] + j));

}

}

int FindMax( int (\*p)[4], int m, int n, int \*pRow, int \*pCol )

{

int i, j, max;

max = \*(p[0]); //1

\*pRow = 0;

\*pCol = 0;

for (i = 0; i < m; i++) //1

{

for (j = 0; j < n; j++) //1

{

if ( \*(p[i] + j) > max ) //1

{

max = \*(p[i] + j) ; //1

\*pRow = i;

\*pCol = j;

}

}

}

return max;

}

用指针编程实现3X4的二维数组的元素读入以及求此二维数组的最大值及最大值下标

请用以下函数实现：

void Input(int (\*p)[4],int m, int n); /\*数组元素读入函数\*/

int FindMax(int \*p[4], int m, int n, int \*pRow, int \*pCol); /\*求最大值及下标函数\*/

\*\*\*输入提示信息："Please input your data:\n"

\*\*\*输入格式要求：无格式要求

\*\*\*输出格式要求："The maximum is %d, which is in row %d, colum %d\n"

样例：

Please input your data:

\*\*输入样例：

3 5 2 7 1 6 12 11 4 10 8 9

\*\*输出样例：

The maximum is 12, which is in row 1, colum 2

#include <stdio.h>

#define M 10

#define N 10

void InputMatrix(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);

int main()

{

int a[M][N], m, n, row, col, max;

printf("Input m,n:\n");

scanf("%d,%d", &m, &n);

InputMatrix(\*a, m, n);

max = FindMax(\*a, m, n, &row, &col);

printf("max=%d,row=%d,col=%d\n", max, row, col);

return 0;

}

/\* 函数功能：输入m\*n矩阵的值 \*/

void InputMatrix(int \*p, int m, int n)

{

int i, j;

printf("Input %d\*%d array:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &p[i \* n + j]);

}

}

}

/\* 函数功能：在m\*n矩阵中查找最大值及其所在的行列号 \*/

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol)

{

int i, j, max = p[0];

\*pRow = 0;

\*pCol = 0;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

if (p[i \* n + j] > max)

{

max = p[i \* n + j];

\*pRow = i; /\*记录行下标\*/

\*pCol = j; /\*记录列下标\*/

}

}

}

return max;

}

找数组最值

按如下函数原型编程从键盘输入一个m行n列的二维数组，然后计算数组中元素的最大值及其所在的行列下标值。其中，m和n的值由用户键盘输入。已知m和n的值都不超过10。

void InputArray(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);//函数返回最大值，pRow和pCol分别返回最大值所在的行列下标

例如，程序的1次运行结果如下：

Input n:

3,4↙

Input 3\*4 array:

1 2 3 4↙

5 6 7 8↙

9 0 -1 -2↙

max=9,row=2,col=0

数组维数输入提示信息: "Input m,n:\n"

数组维数输入格式: "%d,%d"

数组元素输入提示信息: "Input %d\*%d array:\n"

数组元素输入格式:："%d"

输出格式： "max=%d,row=%d,col=%d\n"

#include<stdio.h>

#define N 15

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

int main()

{

int a[N][N] = {0}, n;//1

printf("Input n(n<15):\n");

scanf("%d", &n);//1

CaculateYH(a, n);//1

PrintYH(a, n); //1

return 0;

}

void CaculateYH(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

a[i][0] = 1;//1

a[i][i] = 1;//1

}

for (i = 2; i < n; i++)//1

{

for (j = 1; j <= i - 1; j++) //1

{

a[i][j] = a[i - 1][j - 1] + a[i - 1][j];//1

}

}

}

void PrintYH(int a[][N], int n)

{

int i, j;

for (i = 0; i < n; i++) //1

{

for (j = 0; j <= i; j++) //1

{

printf("%5d", a[i][j]);//1

}

printf("\n");//1

}

}

编程打印如下形式的杨辉三角形。打印的杨辉三角形的行数n（不超过15行）。要求由用户在主函数中通过键盘输入。请按照如下给定的函数原型进行编程。函数CaculateYH用于生成杨辉三角形，函数PrintYH用于打印杨辉三角形。给定的函数原型如下:

void CaculateYH(int a[][N], int n);

void PrintYH(int a[][N], int n);

\*\*输入提示信息\*\*："Input n(n<15):\n"

\*\*输入数据格式\*\*："%d"

\*\*输出提示信息\*\*：无

\*\*输出数据格式\*\*："%5d"

杨辉三角形形式如下：

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

1 10 45 120 210 252 210 120 45 10 1

#include <stdio.h>

#define M 10

#define N 10

void InputMatrix(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);

int main()

{

int a[M][N], m, n, row, col, max;

printf("Input m,n:\n");

scanf("%d,%d", &m, &n);

InputMatrix(\*a, m, n);

max = FindMax(\*a, m, n, &row, &col);

printf("max=%d,row=%d,col=%d\n", max, row, col);

return 0;

}

/\* 函数功能：输入m\*n矩阵的值 \*/

void InputMatrix(int \*p, int m, int n)

{

int i, j;

printf("Input %d\*%d array:\n", m, n);

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

scanf("%d", &p[i \* n + j]);

}

}

}

/\* 函数功能：在m\*n矩阵中查找最大值及其所在的行列号 \*/

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol)

{

int i, j, max = p[0];

\*pRow = 0;

\*pCol = 0;

for (i = 0; i < m; i++)

{

for (j = 0; j < n; j++)

{

if (p[i \* n + j] > max)

{

max = p[i \* n + j];

\*pRow = i; /\*记录行下标\*/

\*pCol = j; /\*记录列下标\*/

}

}

}

return max;

}

找数组最值

按如下函数原型编程从键盘输入一个m行n列的二维数组，然后计算数组中元素的最大值及其所在的行列下标值。其中，m和n的值由用户键盘输入。已知m和n的值都不超过10。

void InputArray(int \*p, int m, int n);

int FindMax(int \*p, int m, int n, int \*pRow, int \*pCol);//函数返回最大值，pRow和pCol分别返回最大值所在的行列下标

例如，程序的1次运行结果如下：

Input n:

3,4↙

Input 3\*4 array:

1 2 3 4↙

5 6 7 8↙

9 0 -1 -2↙

max=9,row=2,col=0

数组维数输入提示信息: "Input m,n:\n"

数组维数输入格式: "%d,%d"

数组元素输入提示信息: "Input %d\*%d array:\n"

数组元素输入格式:："%d"

输出格式： "max=%d,row=%d,col=%d\n"

给定如下定义：

struct date\_rec

{

int day ;

int month ;

int year ;

} ;

struct date\_rec current\_date ;

写一个程序包含如下的函数，完成：

(a) 输入current\_date的值：

void input\_date(struct date\_rec \*current\_date)

(b) 将current\_date增加1天：

void increment\_date(struct date\_rec \*current\_date)

(c) 显示current\_date的值：

void output\_date(struct date\_rec \*current\_date)

考虑每个月的实际天数，同时也考虑闰年的情况。

\*\*输入格式要求："%d%d%d" 提示信息："请输入当前日期（年 月 日）："

\*\*输出格式要求："当前日期：%d年%d月%d日！" （加1天后的新日期）

1.#include <stdio.h>

struct date\_rec

{

int day;

int month;

int year;

} ;

struct date\_rec current\_date;

int days\_of\_month[][13] = {{0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}, {0, 31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}

} ;

int is\_leap(int year)

{

return ((year % 400 == 0) || ((year % 4 == 0) && (year % 100 != 0)));

}

void input\_date(struct date\_rec \*current\_date)

{

printf("请输入当前日期（年 月 日）：");

scanf("%d%d%d", &current\_date->year, &current\_date->month, &current\_date->day);

}

void increment\_date(struct date\_rec \*current\_date)

{

current\_date->day++;

if (current\_date->day > days\_of\_month[is\_leap(current\_date->year)][current\_date->month])

{

current\_date->day = current\_date->day - days\_of\_month[is\_leap(current\_date->year)][current\_date->month]

;

current\_date->month++;

if (current\_date->month > 12)

{

current\_date->year++;

current\_date->month = current\_date->month - 12;

}

}

}

void output\_date(struct date\_rec \*current\_date)

{

printf("当前日期：%d年%d月%d日！", current\_date->year,

current\_date->month, current\_date->day);

}

int main()

{

input\_date(&current\_date);

increment\_date(&current\_date);

output\_date(&current\_date);

return 0;

}

编写函数，通过指针连接两个字符串。

\*\*输入格式要求："%s" 提示信息："Enter string 1:" "Enter string 2:"

\*\*输出格式要求："a+b=%s\n"

程序运行示例如下：

Enter string 1:happy

Enter string 2:hollween

a+b=happyhollween

#include <stdio.h>

**char** \***strcat**(**char** \*str1,**char** \*str2)

{

**char** \*p=str1;

**while**(\*p!='\0') p++;

**while**(\*p++=\*str2++);

**return** str1;

}

main()

{

**char** a[50],b[30];

**printf**("Enter string 1:");

**scanf**("%s",a);

**printf**("Enter string 2:");

**scanf**("%s",b);

**printf**("a+b=%s\n",**strcat**(a,b));

}

请按如下函数原型编程实现将字符数组中的字符串逆序存放。要求在主函数读入字符串，并在主函数中输出逆序存放后的字符串。

void inverse(char \*p);

要求:

1）输入的字符串，应包含字母，数字以及空格字符。

2）字符串最大长度为80。

3）没有输入输出提示信息。

请按如下框架编程，在“......”的位置添加缺失的语句，使程序完整。

|  |
| --- |
| #include<stdio.h>  **void** inverse(**char** \*p);  **int** main()  {  **char** a[80];  **gets**(a);        ......    **puts**(a);  **return** 0;  }  **void** inverse(**char** \*p)  {  **char** \*q;  **char** t;      q = p;  **while** (\*q != '\0')      {          q++;      }        ......  } |

#include<stdio.h>

**void** inverse(**char** \*p);

**int** main()

{

**char** a[80];

**gets**(a);

    inverse(a);  //2

**puts**(a);

}

**void** inverse(**char** \*p)

{

**char** \*q;

**char** t;

    q = p;

**while** (\*q != '\0')

    {

        q++;

    }

**for** (q--; p < q; p++, q--)//2

    {

        t = \*p;//2

        \*p = \*q;//2

        \*q = t;//2

    }

}

从键盘输入一个字符串a(可以包含：字母、数字、标点符号，以及空格字符)，将字符串a复制到字符串b中，再输出字符串，即编写实现字符串处理函数的功能，但要求不能使用字符串处理函数strcpy()。

程序运行示例：

Input a string:hello boy↙

The copy is:hello boy

程序如下，横线处代表有缺失的源代码，请补充缺少的部分，并将完整的程序代码填写在答题区。

#include <stdio.h>

#define N 80

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 函数声明 \*/

int main()

{

char a[N], b[N];

printf("Input a string:");

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 输入字符串 \*/

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 调用函数将字符数组a中的字符串拷贝到b中 \*/

printf("The copy is:");

puts(b); /\* 输出复制后的字符串 \*/

return 0;

}

/\* 函数功能：用字符指针作为函数参数，实现字符串拷贝 \*/

void MyStrcpy(char \*dstStr, char \*srcStr)

{

while ( \_\_\_\_\_\_\_\_\_\_\_\_\_) /\* 若当前srcStr所指字符不是字符串结束标志 \*/

{

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 复制字符 \*/

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 使srcStr指向下一个字符 \*/

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 使dstStr指向下一个存储单元 \*/

}

\_\_\_\_\_\_\_\_\_\_\_\_\_ /\* 在字符串dstStr的末尾添加字符串结束标志 \*/

}

#include  <stdio.h>

#define N 80

**void**  MyStrcpy(**char** \*dstStr, **char** \*srcStr);

**int** main()

{

**char**  a[N], b[N];

**printf**("Input a string:");

**gets**(a);                      //1

    MyStrcpy(b, a);               //1

**printf**("The copy is:");

**puts**(b);

**return** 0;

}

/\* 函数功能：用字符指针作为函数参数，实现字符串拷贝 \*/

**void**  MyStrcpy(**char** \*dstStr, **char** \*srcStr)

{

**while** (\*srcStr != '\0')    //1

    {

        \*dstStr = \*srcStr;  //1

        srcStr++;               //1

        dstStr++;               //1

    }

    \*dstStr = '\0';             //1

}

用字符指针作函数参数编程实现如下功能：在字符串中删除与某字符相同的字符。

\*\*输入格式要求："%s"

输入提示信息：

"Input a string:"

"Input a character:"

\*\*输出格式要求："Results:%s\n"

程序运行示例1如下：

Input a string:hello,world!

Input a character:o

Results:hell,wrld!

请按如下框架编程：

|  |
| --- |
| #include <stdio.h>  #include <string.h>  #define N 100  **void**  Squeeze(**char** \*s, **char** c);  **int** main()  {  **char**  str[20], ch;  **printf**("Input a string:");  **gets**(str);  **printf**("Input a character:");      ch = **getchar**();        ......    **printf**("Results:%s\n", str);  **return** 0;  }  **void**  Squeeze(**char** \*s, **char** c)  {      ......  } |
| #include <stdio.h>  #include <string.h>  #define N 100  void  Squeeze(char \*s, char c);  int main()  {      char  str[20], ch;      printf("Input a string:");      gets(str);      printf("Input a character:");      ch = getchar();      Squeeze(str, ch);      printf("Results:%s\n", str);      return 0;  }  void  Squeeze(char \*s, char c)  {      char str[N];      char \*t = str;      strcpy(t, s);      for (; \*t != '\0'; t++)      {          if (\*t != c)          {              \*s = \*t;              s++;          }      }      \*s = '\0';  /\* 在字符串t2的末尾添加字符串结束标志 \*/  } |

请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：

1、在主函数读入字符串；

2、判断是否是回文在主函数调用Judger函数实现，若是回文该函数返回值为1，否则返回值为0；

3、打印信息在主函数进行。

请按题目要求及给定函数原型编写程序，否则即使程序结果运行正确，人工批阅后成绩记为0分。

Judger()函数函数原型为：

int Judger( char \*p);

返回值：返回值是1表示是回文，返回值0表示不是回文。

|  |
| --- |
|  |

 #include <stdio.h>

**int** Judger(**char** \*p);

**int** main()

{

**char** a[80];

**int** flag;

**gets**(a);

    flag = Judger(a);   //2

**if** (flag)

    {

**puts**(a);

    }

**else**

    {

**printf**("No!");

    }

**return** 0;

}

**int** Judger(**char** \*p)   //2

{

**int** flag = 1;

**char** \*q;

    q = p;           //1

**while** (\*q != '\0')   //1

    {

        q++;            //1

    }

    q--;               //1

**while** (p < q)      //1

    {

**if** (\*p != \*q)  //1

        {

            flag = 0;

        }

        p++;           //1

        q--;           //1

    }

**return** flag;

}

编写函数，通过指针将一个字符串反向。

\*\*输入格式要求："%s" 提示信息："Enter string:"

\*\*输出格式要求："a=%s\n"

程序运行示例如下：

Enter string:love

a=evol

#include <stdio.h>

revstr(**char** \*s)

{

**char** \*p=s,c;

**while**(\*p) p++;

    p--;

**while**(s<p)

    {

        c=\*s;

        \*s++=\*p;

        \*p--=c;

    }

}

main()

{

**char** a[50];

**printf**("Enter string:");

**scanf**("%s",a);

    revstr(a);

**printf**("a=%s\n",a);

}

请按给定的程序框架，将is\_within()函数补充完整。is\_within()函数有两个参数，一个是字符，另一个是字符指针。其功能是如果字符在字符串中，返回值为1；如果字符不在字符串中，返回值为0。

|  |
| --- |
| #include <stdio.h>  **int** is\_within(**char** \*p, **char** s);  **int** main()  {  **char** a[80], s;  **int** flag;    **gets**(a);      s = **getchar**();        flag = is\_within(a, s);    **if** (flag)  **printf**("Read the characters in a string.");  **else**  **printf**("Read the characters not in the string.");    **return** 0;  }    **int** is\_within(**char** \*p, **char** s)  {      ... ...  } |
| #include <stdio.h>  int is\_within(char \*p, char s);  int main()  {      char a[80], s;      int flag;        gets(a);      s = getchar();        flag = is\_within(a, s);        if (flag)          printf("Read the characters in a string.");      else          printf("Read the characters not in the string.");        return 0;  }  int is\_within(char \*p, char s)  {      int flag = 0;        for (; \*p != '\0'; p++)   //2      {          if (s == \*p)         //2          {              flag = 1;        //1              break;           //1          }      }      return flag;             //1  } |

    用指针做函数参数编程实现字符串连接函数strcat()的功能。

下面程序中存在比较隐蔽的错误，

请通过分析和调试程序，发现并改正程序中的错误。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

经教师手工核对后，如果未用指针做函数参数编程，那么即使做对也不给分。

#include <stdio.h>

#include <string.h>

#define N 80

void MyStrcat(char \*dstStr, char \*srcStr);

main() {

char s[N], t[N];

printf("Input a string:\n");

gets(s);

printf("Input another string:\n");

gets(t);

MyStrcat(s, t);

printf("Concatenate results:%s\n", s);

}

void MyStrcat(char \*dstStr, char \*srcStr) {

while (\*srcStr != '\0');

{

srcStr++;

}

while (\*srcStr != '\0');

{

\*dstStr = \*srcStr;

srcStr++;

dstStr++;

}

}

#include <stdio.h>

#include <string.h>

#define N 80

**void** MyStrcat(**char** \*dstStr, **char** \*srcStr);

main() {

**char** s[N], t[N];

**printf**("Input a string:\n");

**gets**(s);

**printf**("Input another string:\n");

**gets**(t);

    MyStrcat(s, t);

**printf**("Concatenate results:%s\n", s);

}

**void** MyStrcat(**char** \*dstStr, **char** \* srcStr) {

**while** (\*dstStr != '\0')

    {

        dstStr++;

    }

**while** (\*srcStr != '\0')

    {

        \*dstStr = \*srcStr;

        srcStr++;

        dstStr++;

    }

    \*dstStr = '\0';

}

**参考答案**

字符数组困难

将一个字符串插入至另一个源字符串的某个位置：

将一个字符串2插入到源字符串1中 第一次出现某字符的位置，并打印出形成的新串。

如果 字符串1中找不到输入的字符， 则显示“Not found!”并结束程序。

注：源字符串长度及待插入字符串长度不超过50

提示信息：

printf("Input source string 1:\n")

printf("Input inserted string 2:\n")

printf("Input a letter to locate the index:\n")

输出信息格式：

printf("The new string is:%s")

printf(“Not found!”)

测试样例1：

输入信息：

Input source string 1:

abcdecfg

Input inserted string 2:

\*-\*-\*-\*

Input the a letter to locate the index:

c

输出结果：

The new string is:ab\*-\*-\*-\*cdecfg

测试样例2：

输入信息：

Input source string 1:

abcdecfg

Input inserted string 2:

\*\*\*\*

Input the a letter to locate the index:

h

输出结果：

Not found!

#include <stdio.h>

#include <string.h> //1

#include <stdlib.h>

int main()

{

int i = 0, j = 0, n, len;

char str1[200] = {'\0'};

char str2[100];

char strtem[100];

char ch;

printf("Input source string 1:\n");

gets(str1);

printf("Input inserted string 2:\n");

gets(str2);

printf("Input a letter to locate the index:\n");

scanf(" %c", &ch);

len = strlen(str1);

while (str1[i] != '\0')

{

if (str1[i] == ch)

{

n = i;

break;

}

i++; } if (i != len) //1

{

for (i = n; str1[i] != '\0'; j++, i++)

{

strtem[j] = str1[i]; //1

str1[i] = '\0'; //1

}

strtem[j] = '\0'; //1

strcat(str1, str2); //1

strcat(str1, strtem); //1

printf("The new string is:%s", str1);

}

else

{

printf("Not found!");

}

return 0;

}

**一维数组的参数**

下面函数MyStrcat实现strcat的功能，即：将两个字符串连接的功能，

将源字符串srcStr连接到目的字符串dstStr的尾部。

注意：

（1）请将修改正确后的完整源程序拷贝粘贴到答题区内。

（2）对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

（3）当且仅当错误全部改正，且程序运行结果调试正确，才能得满分。

|  |
| --- |
| #include <stdio.h>  #define ARR\_SIZE 80;  void MyStrcat(char dstStr[], char srcStr[]);  main()  {      char  s[], t[];      printf("Please enter source string:\n");      gets(s);      printf("Please enter destination string:\n");      gets(t);        MyStrcat(s,t);      printf("The concatenate string is:\n");      puts(t);  }  void MyStrcat(char dstStr[], char srcStr[])  {      int i = 0, j = 0;        while (dstStr[i]!='\0')      {          i++;      }      for(;srcStr[i]!='\0';i++,j++)      {          dstStr[j] = srcStr[i];      }      dstStr[j]='\0';  } |

|  |
| --- |
| #include <stdio.h>  #define ARR\_SIZE 80  void MyStrcat(char dstStr[], char srcStr[]);  main()  {      char  s[ARR\_SIZE], t[ARR\_SIZE];      printf("Please enter source string:\n");      gets(s);      printf("Please enter destination string:\n");      gets(t);        MyStrcat(t,s);      printf("The concatenate string is:\n");      puts(t);  }  void MyStrcat(char dstStr[], char srcStr[])  {      int i = 0, j = 0;        while (dstStr[i]!='\0')      {          i++;      }      for(;srcStr[j]!='\0';i++,j++)      {          dstStr[i] = srcStr[j];      }      dstStr[i]='\0';  } |

从键盘输入某班学生某门课的成绩（每班人数最多不超过40人），当输入为负值时，表示输入结束，试编程从键盘任意输入一个学号，查找该学号学生的成绩。

\*\*输入格式要求："%ld"(学号) "%ld%d" 提示信息："Total students are %d\n" "Input the searching ID:" "Input student’s ID and score:"

\*\*输出格式要求："score = %d\n" "Not found!\n"

程序的两次运行示例如下：

① Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

Input the searching ID:070310123

score = 83

② Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

Input the searching ID:070310128

Not found!

完

|  |
| --- |
| #include <stdio.h>  #define N 40  int ReadScore(int score[], long num[]);          /\*ReadScore()函数原型\*/  int LinSearch(long num[], long x, int n);        /\* LinSearch()原函数型\*/  int main()  {      int score[N], n, pos;      long num[N], x;      n = ReadScore(score, num);  /\* 输入成绩和学号，返回学生总数 \*/      printf("Total students are %d\n", n);      printf("Input the searching ID:");      scanf("%ld", &x);            /\* 以长整型格式从键盘输入待查找的学号x \*/      pos = LinSearch(num, x, n); /\* 查找学号为num的学生 \*/      if (pos != -1)                   /\* 若找到，则打印其分数 \*/      {          printf("score = %d\n", score[pos]);      }      else                            /\* 若未找到，则打印"未找到"提示信息\*/      {          printf("Not found!\n");      }      return 0;  }  /\* 函数功能：输入学生的学号及其某门课成绩，当输入负值时，结束输入，返回学生人数 \*/  int ReadScore(int score[], long num[]) /\* ReadScore()函数定义 \*/  {      int i = -1;         /\*i初始化为-1，循环体内增1后可保证数组下标从0开始\*/      do      {          i++;          printf("Input student’s ID and score:");          scanf("%ld%d", &num[i], &score[i]);      }      while (num[i] > 0 && score[i] >= 0);  /\* 输入负值时结束成绩输入 \*/      return i;                                   /\* 返回学生总数 \*/  }  /\*按线性查找法查找值为x的数组元素，若找到则返回x在数组中的下标位置，否则返回-1\*/  int LinSearch(long num[], long x, int n) /\* LinSearch()函数定义 \*/  {      int  i;      for (i = 0; i < n; i++)      {          if (num[i] == x)    return (i); /\* 若找到则返回x在数组中的下标 \*/      }      return(-1);                           /\* 若循环结束仍未找到，则返回-1 \*/  } |

计算平均数、中位数和众数

在调查数据分析（Survey data analysis）中经常需要计算平均数、中位数和众数。用函数编程计算40个输入数据（是取值1—10之间的任意整数）的平均数（Mean）、中位数（Median）和众数（Mode）。中位数指的是排列在数组中间的数。众数是数组中出现次数最多的那个数（不考虑两个或两个以上的输入数据出现次数相同的情况）。

提示：计算中位数时，首先要调用排序函数对数组按升序进行排序，然后取出排序后数组中间位置的元素answer[n/2] ，就得到了中位数。如果数组元素的个数是偶数，那么中位数就等于数组中间那两个元素的算术平均值。众数就是40个输入数据中出现次数最多的那个数。计算众数时，首先要统计不同取值的输入数据出现的次数，然后找出出现次数最多的那个数据，这个数据就是众数（这里没有考虑两个或者两个以上的输入数据出现次数相同的情况）。

程序运行结果示例：

Input the feedbacks of 40 students:

10 9 10 8 7 6 5 10 9 8↙

8 9 7 6 10 9 8 8 7 7↙

6 6 8 8 9 9 10 8 7 7↙

9 8 7 9 7 6 5 9 8 7↙

Mean value=7

Median value=8

Mode value=8

输入格式: "%d"

输出格式：

输入数据的提示信息："Input the feedbacks of 40 students:\n"

平均数输出："Mean value=%d\n"

中位数输出："Median value=%d\n"

众数输出： "Mode value=%d\n"

|  |
| --- |
| #include <stdio.h>  #define  M   40  #define  N   11  int Mean(int answer[], int n);  int Median(int answer[], int n);  int Mode(int answer[], int n);  void DataSort(int a[], int n);  int main()  {      int  i, feedback[M];      printf("Input the feedbacks of 40 students:\n");      for (i = 0; i < M; i++)      {          scanf("%d", &feedback[i]);      }      printf("Mean value=%d\n", Mean(feedback, M));      printf("Median value=%d\n", Median(feedback, M));      printf("Mode value=%d\n", Mode(feedback, M));      return 0;  }  /\* 函数功能：若n>0则计算并返回n个数的平均数，否则返回-1 \*/  int Mean(int answer[], int n)  {      int i, sum = 0;      for (i = 0; i < n; i++)      {          sum += answer[i];      }      return  n > 0 ? sum / n : -1;  }  /\* 函数功能：计算n个数的中位数 \*/  int Median(int answer[], int n)  {      DataSort(answer, n);      if (n % 2 == 0)          return  (answer[n / 2] + answer[n / 2 - 1]) / 2;      else          return  answer[n / 2];  }  /\* 函数功能：计算n个数的众数 \*/  int Mode(int answer[], int n)  {      int  i, grade, max = 0, modeValue = 0, count[N] = {0};      for (i = 0; i < n; i++)      {          count[answer[i]]++;      }      for (grade = 1; grade <= N - 1; grade++)      {          if (count[grade] > max)          {              max = count[grade];              modeValue = grade;          }      }      return modeValue;  }  /\*  函数功能：按选择法对数组a中的n个元素进行排序 \*/  void DataSort(int a[], int n)  {      int i, j, k, temp;      for (i = 0; i < n - 1; i++)      {          k = i;          for (j = i + 1; j < n; j++)          {              if (a[j] > a[k]) k = j;          }          if (k != i)          {              temp = a[k];              a[k] = a[i];              a[i] = temp;          }      }  } |

|  |  |
| --- | --- |
| **总分** | 0 |

从键盘输入某班学生某门课的成绩和学号（最多不超过40人），当输入为负值时，表示输入结束，用函数编程通过返回数组中最大元素的下标，查找并输出成绩的最高分及其所在的学生学号。

\*\*输入格式要求："%ld%d" 提示信息："Input student’s ID and score:" "input error!\n"

\*\*输出格式要求："Total students are %d\n" "The highest is:%ld, %d\n"

程序运行示例如下：

Input student’s ID and score:070310122 84

Input student’s ID and score:070310123 83

Input student’s ID and score:070310124 88

Input student’s ID and score:070310125 87

Input student’s ID and score:070310126 61

Input student’s ID and score:-1 -1

Total students are 5

The highest is:70310124, 88

|  |
| --- |
| #include <stdio.h>  #define N 40  int ReadScore(int score[], long num[]);  int FindMax(int score[], int n);  int main()  {      int score[N], maxNum, n;      long num[N];      n = ReadScore(score, num);      /\* 输入成绩，返回学生人数 \*/      printf("Total students are %d\n", n);      maxNum = FindMax(score, n);        /\* 计算并返回最高分所在数组的下标 \*/      printf("The highest is:%ld, %d\n", num[maxNum], score[maxNum]);      return 0;  }  /\* 函数功能：输入学生某门课的成绩，当输入负值时，结束输入，返回学生人数 \*/  int ReadScore(int score[], long num[])  {      int i = -1;      do      {          i++;          printf("Input student’s ID and score:");          scanf("%ld%d", &num[i], &score[i]);      }      while (score[i] >= 0 && num[i] >= 0);      return i;  }  /\* 函数功能：计算并返回最高分所在数组的下标 \*/  int FindMax(int score[], int n)  {      int max, i, maxNum = 0;      max = score[0];      for (i = 1; i < n; i++)      {          if (score[i] > max)          {              max = score[i];              maxNum = i;          }      }      return maxNum;  } |

插入排序（Insertion Sort）。用函数编程实现在一个按升序排序的数组中查找x应插入的位置，将x插入数组中，使数组元素仍按升序排列。

\*\*输入格式要求："%d" 提示信息："Input array size:" "Input array:" "Input x:"

\*\*输出格式要求："After insert %d:\n" "%4d"

程序运行示例如下：

Input array size:5

Input array:1 3 5 7 9

Input x:4

After insert 4:

1 3 4 5 7 9

#include<stdio.h>

#define N 20 /\* 插入前数组最大元素个数 \*/

void Insert(int a[], int n, int x);

main()

{

int a[N+1]; /\* 定义数组长度为插入前的数组元素个数加1 \*/

int x, i, n;

printf("Input array size:");

scanf("%d", &n); /\* 输入插入前数组元素个数 \*/

printf("Input array:");

for (i=0; i<n; i++)

{

scanf("%d", &a[i]); /\* 输入插入前已按升序排序的数组元素 \*/

}

printf("Input x:");

scanf("%d", &x); /\* 输入待插入的元素x \*/

Insert(a, n, x); /\* 插入元素x到已排序数组中 \*/

printf("After insert %d:\n", x);

for (i=0; i<n+1; i++)

{

printf("%4d", a[i]); /\* 输出插入x后的数组元素 \*/

}

printf("\n");} /\* 函数功能：将x插入到一个已按升序排序的数组中 \*/

void Insert(int a[], int n, int x)

{

int i = 0, pos;

while (i < n && x > a[i]) /\* 查找待插入位置 \*/

{

i++;

}

pos = i; /\* 记录元素x应插入的数组下标位置pos \*/

for (i = n-1; i>= pos; i--)/\* 从尾部开始移动pos及其后所有的元素 \*/

{

a[i+1] = a[i]; /\* 向后复制数组元素 \*/

}

a[pos] = x; /\* 插入元素x到位置pos \*/

}

从键盘输入一行字符（最长不超过80字符），

用函数编程统计其中单词（以空格作为间隔符的字符串）的个数。

例如How are you中单词个数为3。

要求：

(1)按如下函数原型进行编程：

int CountWords(char str[])

(2)在主函数中：

输入一行字符;

然后调用函数CountWords计算单词的个数;

最后打印计算结果。

\*\*输入提示信息为："Input a string:\n"

\*\*输出格式为:"Number of words=%d\n"

#include <stdio.h>

int CountWords(char str[]);

void main()

{

char str[80];

int num;

printf("Input a string:\n");

gets(str);//2

num=CountWords(str);//2

printf("Number of words=%d\n", num);//2

}

int CountWords(char str[])

{

int i, num;

if(str[0]!=' ') //2

{

num=1;

}//1

else

{

num=0;

}//1

for (i=1; str[i]!='\0'; i++)//4

{

if (str[i]!=' ' && str[i-1] == ' ') //4

{

num++;//1

}

}

return num;//1

}

请按如下函数原型编程实现从键盘输入一组无序的10个整数，按从小到大的顺序插入到数组中，然后输出。

函数原型：insert（int px[]）

要求：

在主函数中输入一组无序的整数，调用函数insert（）实现按从小到大的顺序插入到数组中，然后在主函数中输出。

\*\*要求输入提示信息为:无

\*\*要求输入格式为: "%d"

\*\*要求输出格式为："%d "

"%d"

程序运行示例如下：

9 23 78 99 34 66 1 20 60 16 （此处输入十个无序整数）

1 9 16 20 23 34 60 66 78 99

#include <stdio.h>

void insert(int px[]);

int main()

{

int a[10], i;

for (i = 0; i < 10; i++)

{

scanf("%d", &a[i]);

}

insert(a);

for (i = 0; i < 9; i++)

{

printf("%d ", a[i]);

}

printf("%d", a[9]);

return 0;

}

void insert(int px[])

{

int k, i, j;

for (j = 0; j < 10; j++)

{

for (i = 0; i < 9 - j; i++)

{

if (px[i] > px[i + 1])

{

k = px[i];

px[i] = px[i + 1];

px[i + 1] = k;

}

}

}

}

|  |
| --- |
|  |

编写一个程序，将十进制短整型数n转换成二进制数。

输入提示信息："n="

\*\*输入格式："%d"

输出信息："the binary number is "

\*\*输出格式要求："%d"

程序运行示例如下：

n=37

the binary number is 0000000000100101

#include <stdio.h>

void trans(int n, int b[]);

int main()

{

int b[16], n, i;

for (i = 0; i <= 15; i++)

{

b[i] = 0;

}

printf("n=");

scanf("%d", &n);

trans(n, b);

printf("the binary number is ");

for (i = 15; i >= 0; i--)

{

printf("%d", b[i]);

}

printf("\n");

return 0;

}

void trans(int n, int b[])

{

int i = 0;

while (n != 0)

{

b[i] = n % 2;

i++;

n /= 2;

}

}

从键盘输入n个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。（提示：记住下标再互换位置。）

要求按如下函数原型编写程序

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

\*\*输入提示信息要求：

"Input n(n<=10):\n"

"Input %d numbers:\n"

\*\*要求输入格式为："%d"

\*\*输出提示信息："Exchange results:"

\*\*要求输出格式为："%5d"

输出数组中所有数据后换行

程序运行结果示例：

Input n(n<=10):

5

Input 5 numbers:

11 35 42 67 21

Exchange results: 67 35 42 11 21

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

int main()

{

int a[10], n;

printf("Input n(n<=10):\n");

scanf("%d", &n);

printf("Input %d numbers:\n", n);

ReadData(a, n);//1

MaxMinExchang(a, n);//1

printf("Exchange results:");

PrintData(a, n);//1

return 0;

}

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++) //1

{

scanf("%d", &a[i]);

}

}

void PrintData(int a[], int n) {

int i;

for (i = 0; i < n; i++)//1

{

printf("%5d", a[i]);//1

}

printf("\n");//1

}

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;//1

int i, temp;

for (i = 1; i < n; i++) //1

{

if (a[i] > maxValue) //1

{

maxValue = a[i];//1

maxPos = i;//1

}

if (a[i] < minValue) //1

{

minValue = a[i];

minPos = i;

}

}

temp = a[maxPos];//1

a[maxPos] = a[minPos];//1

a[minPos] = temp;//1

}

验证卡布列克运算。即：任意一个四位数，只要它们各个位上的数字是不全相同的，就有这样的规律：

（1）将组成该四位数的四个数字由大到小排列，形成由这四个数字构成的最大的四位数；

（2）将组成该四位数的四个数字由小到大排列，形成由这四个数字构成的最小的四位数（如果四个数字中含有0，则得到的数不足四位）；

（3）求两个数的差，得到一个新的四位数（高位零保留）。

重复以上过程，最后得到的结果总是6174。这个数被称为卡布列克常数。请编写一个递归函数，完成以上的卡布列克运算。

\*\*输入格式要求："%d" 提示信息："Enter number:"

\*\*输出格式要求：" [%d]:%d-%d=%d\n"

程序运行示例如下：

Enter number: 1234

[1]:4321-1234=3087

[2]:8730-378=8352

[3]:8532-2358=6174

#include <stdio.h>

int count=0;

main()

{

int n;

printf("Enter number:");

scanf("%d",&n);

vr6174(n);

}

vr6174(int num)

{

int each[4],max,min;

if(num!=6174&&num)

{

parse\_sort(num,each);

max\_min(each,&max,&min);

num=max-min;

printf(" [%d]:%d-%d=%d\n",++count,max,min,num);

vr6174(num);

}

}

parse\_sort(int num,int \*each)

{

int i,\*j,\*k,temp;

for(i=0;i<=4;i++)

{

j=each+3-i;

\*j=num%10;

num/=10;

}

for(i=0;i<3;i++)

for(j=each,k=each+1;j<each+3-i;j++,k++)

if(\*j>\*k)

{ temp=\*j; \*j=\*k; \*k=temp;}

return;

}

max\_min(int \*each,int \*max,int \*min)

{

int \*i;

\*min=0;

for(i=each;i<each+4;i++)

\*min=\*min\*10+\*i;

\*max=0;

for(i=each+3;i>=each;i--)

\*max=\*max\*10+\*i;

return;

}

从键盘输入一行字符（最长不超过80字符），

用函数编程统计其中单词（以空格作为间隔符的字符串）的个数。

例如How are you中单词个数为3。

要求：

(1)按如下函数原型进行编程：

int CountWords(char str[])

(2)在主函数中：

输入一行字符;

然后调用函数CountWords计算单词的个数;

最后打印计算结果。

\*\*输入提示信息为："Input a string:\n"

\*\*输出格式为:"Number of words=%d\n"

#include <stdio.h>

int CountWords(char str[]);

void main()

{

char str[80];

int num;

printf("Input a string:\n");

gets(str);//2

num=CountWords(str);//2

printf("Number of words=%d\n", num);//2

}

int CountWords(char str[])

{

int i, num;

if(str[0]!=' ') //2

num=1;//1

else

num=0;//1

for (i=1; str[i]!='\0'; i++)//4

{

if (str[i]!=' ' && str[i-1] == ' ') //4

{

num++;//1

}

}

return num;//1

}

\*将1到9这九个数字分成三个3位数，要求第一个3位数，正好是第二个3位数的二倍，是第三个3位数的三倍，问应当怎样分。

\*\*输出格式要求："No.%d: %d %d %d\n"

程序运行示例如下：

No.1: 192 384 576

No.2: 219 438 657

No.3: 273 546 819

No.4: 327 654 981

#include <stdio.h>

int a[9];

main()

{

int m,count=0;

for(m=123;m<=333;m++)

if(ok(m,a)&&ok(2\*m,a+3)&&ok(3\*m,a+6))

printf("No.%d: %d %d %d\n",++count,m,2\*m,3\*m);

}

int ok(t,z)

int t,\*z;

{

int \*p1,\*p2;

for(p1=z;p1<z+3;p1++)

{

\*p1=t%10;

t/=10;

for(p2=a;p2<p1;p2++)

if(\*p1==0||\*p2==\*p1)

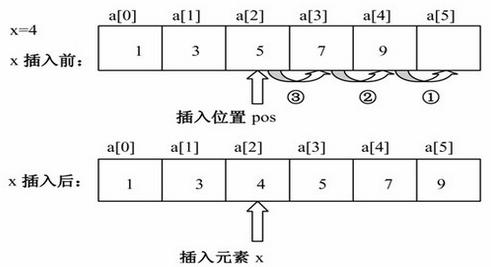
return 0;

}

return 1;

}

在升序排序的数组中插入一个元素 用函数编程实现在一个按升序排序的数组中查找x应插入的位置，将x插入数组中，使数组元素仍按升序排列。 提示：插入（Insertion）是数组的基本操作之一。插入法排序算法的关键在于要找到正确的插入位置，然后依次移动插入位置及其后的所有元素，腾出这个位置放入待插入的元素。插入排序的原理如图所示：



程序运行结果示例：

Input array size:

5↙

Input array:

1 3 5 7 9↙

Input x:

4↙

After insert 4:

1 3 4 5 7 9

**输入格式:**

插入前数组元素个数、数组元素、待插入的元素x的输入格式都是："%d"

**输出格式：**

输入插入前数组元素个数提示信息："Input array size:\n"

输入插入前已按升序排序的数组元素提示信息："Input array:\n"

输入待插入的元素x提示信息："Input x:\n"

输出插入x后的数组元素提示信息："After insert %d:\n"

数组元素输出格式："%4d"

#include<stdio.h>

#define N 20 /\* 插入前数组最大元素个数 \*/

void Insert(int a[], int n, int x);

int main()

{

int a[N + 1]; /\* 定义数组长度为插入前的数组元素个数加1 \*/

int x, i, n;

printf("Input array size:\n");

scanf("%d", &n); /\* 输入插入前数组元素个数 \*/

printf("Input array:\n");

for (i = 0; i < n; i++)

{

scanf("%d", &a[i]); /\* 输入插入前已按升序排序的数组元素 \*/

}

printf("Input x:\n");

scanf("%d", &x); /\* 输入待插入的元素x \*/

Insert(a, n, x); /\* 插入元素x到已排序数组中 \*/

printf("After insert %d:\n", x);

for (i = 0; i < n + 1; i++)

{

printf("%4d", a[i]); /\* 输出插入x后的数组元素 \*/

}

return 0;

}

/\* 函数功能：将x插入到一个已按升序排序的数组中 \*/

void Insert(int a[], int n, int x)

{

int i = 0, pos;

while (i < n && x > a[i]) /\* 查找待插入位置 \*/

{

i++;

}

pos = i; /\* 记录元素x应插入的数组下标位置pos \*/

for (i = n - 1; i >= pos; i--) /\* 从尾部开始移动pos及其后所有的元素 \*/

{

a[i + 1] = a[i]; /\* 向后复制数组元素 \*/

}

a[pos] = x; /\* 插入元素x到位置pos \*/

}

从键盘输入10个整数，用函数编程实现计算其最大值和最小值，并互换它们所在数组中的位置。

要求按如下函数原型编写程序

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

\*\*输入提示信息要求：

"Input n(n<=10):\n"

"Input %d numbers:\n"

\*\*要求输入格式为："%d"

\*\*输出提示信息："Exchange results:"

\*\*要求输出格式为："%5d"

输出数组中所有数据后换行

注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程，主函数不能使用int main和return 0。

#include <stdio.h>

void ReadData(int a[], int n);

void PrintData(int a[], int n);

void MaxMinExchang(int a[], int n);

int main()

{

int a[10], n;

printf("Input n(n<=10):\n");

scanf("%d", &n);

printf("Input %d numbers:\n", n);

ReadData(a, n);//1

MaxMinExchang(a, n);//1

printf("Exchange results:");

PrintData(a, n);//1

return 0;

}

void ReadData(int a[], int n)

{

int i;

for (i = 0; i < n; i++) //1

{

scanf("%d", &a[i]);

}

}

void PrintData(int a[], int n) {

int i;

for (i = 0; i < n; i++)//1

{

printf("%5d", a[i]);//1

}

printf("\n");//1

}

void MaxMinExchang(int a[], int n)

{

int maxValue = a[0], minValue = a[0], maxPos = 0, minPos = 0;//1

int i, temp;

for (i = 1; i < n; i++) //1

{

if (a[i] > maxValue) //1

{

maxValue = a[i];//1

maxPos = i;//1

}

if (a[i] < minValue) //1

{

minValue = a[i];

minPos = i;

}

}

temp = a[maxPos];//1

a[maxPos] = a[minPos];//1

a[minPos] = temp;//1

}

输入某班学生某门课的成绩（最多不超过40人），

当输入为负值时，表示输入结束，

用函数编程统计成绩不低于平均分的学生人数。

要求：

(1)按如下函数原型进行编程：

int GetAboveAver(int score[], int n);

(2)在主函数中：

输入学生成绩，

然后调用函数GetAboveAver计算成绩不低于平均分的学生人数，

最后输出该人数。

要求:

(1)学生成绩和平均分均定义为int类型

(2)\*\*无输入提示信息

\*\*输入格式为："%d"

\*\*输出格式为："Students of above average is %d\n"

#include <stdio.h>

#define N 40

int GetAboveAver(int score[], int n);

void main()

{

int score[N], m, n;

n = -1; //1

do{

n++;

scanf("%d", &score[n]);//1

}while (score[n] >= 0); //e2

m = GetAboveAver(score, n); //2

printf("Students of above average is %d\n", m);

}

int GetAboveAver(int score[], int n)

{

int i, sum, aver,count;

count = 0;//1

sum = 0;//1

for (i=0; i<n; i++)//1

{

sum += score[i];//2

}

aver=sum/n;//1

for (i=0; i<n; i++)//1

{

if (score[i] >= aver)//2

count++;//1

}

return count;//1

}

在数组元素中找最大值及其所在下标位置。

#include <stdio.h>

#define n=10;

int FindMax(int num[], int n, int \*pMaxPos);

main()

{

int num[n], maxValue, maxPos, minValue, minPos, i;

printf("Input %d numbers:\n",n);

for (i=0; i<n; i++)

{

scanf("%d", &num[i]);

}

maxValue = FindMax(num, n, &maxPos);

printf("Max=%d, Position=%d\n", maxValue, maxPos);

}

int FindMax(int num[], int n, int \*pMaxPos)

{

int i, max = num[0];

for (i=1; i<n; i++)

{

if (num[i] > max)

{

max = num[i];

\*pMaxPos = i;

}

}

return max;

}

#include <stdio.h>

#define N 10

int FindMax(int num[], int n, int \*pMaxPos);

main()

{

int num[N], maxValue, maxPos, minValue, minPos, i;

printf("Input %d numbers:\n",N);

for (i=0; i<N; i++)

{

scanf("%d", &num[i]);

}

maxValue = FindMax(num, N, &maxPos);

printf("Max=%d, Position=%d\n", maxValue, maxPos);

}

int FindMax(int num[], int n, int \*pMaxPos)

{

int i, max = num[0];

\*pMaxPos=0;

for (i=0;i<n;i++)

{

if (num[i] > max)

{

max = num[i];

\*pMaxPos = i;

}

}

return max;

}

编写人得票统计程序。设有3个候选人，每次输入一个得票候选人的名字，不考虑弃权情况，要求最后输出各个候选人的得票结果（参加投票人数由程序运行时输入）。

程序的运行示例如下：

输入3个候选人的基本信息：姓名

a

b

c

输入参加投票的人数:5

输入5个参加投票人的投票情况

a

a

b

b

c

输出3个候选人的基本信息：姓名，票数

a 2

b 2

c 1

#include <stdio.h>

#define N 3

struct Person

{

char name[16];

int count;

} ;

void Input(struct Person ld[], int n); //输入n个候选人的基本信息

void Vote(struct Person ld[], int n); //对n个候选人进行投票

void Output(struct Person ld[], int n); //输出n个候选人的基本信息

int main(int argc, char \*argv[])

{

struct Person leader[N];

Input(leader, N);

Vote(leader, N);

Output(leader, N);

return 0;

}

void Input(struct Person ld[], int n)

{

int i;

printf("输入%d个候选人的基本信息：姓名\n", n);

for (i = 0; i < n; i++)

scanf("%s", ld[i].name);

}

void Vote(struct Person ld[], int n)

{

int m, i, j; //m表示参加投票的人数

char ename[16];

printf("输入参加投票的人数:");

scanf("%d", &m);

printf("输入%d个参加投票人的投票情况\n", m);

for (i = 0; i < m; i++)

{

ld[i].count = 0;

}

for (i = 0; i < m; i++)

{

scanf("%s", ename);

for (j = 0; j < n; j++)

if (strcmp(ld[j].name, ename) == 0)

ld[j].count++;

}

}

void Output(struct Person ld[], int n)

{

int i;

printf("输出%d个候选人的基本信息：姓名，票数\n", n);

for (i = 0; i < n; i++)

printf("%s %d\n", ld[i].name, ld[i].count);

}

有n个整数，使其前面各数顺序向后移m个位置，最后m个数变成最前面m个数。

程序的运行示例如下：

共有多少个数？5

后移多少个？2

请以,号为间隔输入5个数。

1,2,3,4,5

移动后顺序为：

4,5,1,2,3

#include <stdio.h>

move(int array[], int n, int m)

{

int \*p, array\_end;

array\_end = \*(array + n - 1);

for (p = array + n - 1; p > array; p--)

{

\*p = \*(p - 1);

}

\*array = array\_end;

m--;

if (m > 0)

{

move(array, n, m);

}

}

int main(void)

{

int number[20], n, m, i;

printf("the total numbers is:");

scanf("%d", &n);

printf("back m:");

scanf("%d", &m);

for (i = 0; i < n - 1; i++)

{

scanf("%d,", &number[i]);

}

scanf("%d", &number[n - 1]);

move(number, n, m);

for (i = 0; i < n - 1; i++)

{

printf("%d,", number[i]);

}

printf("%d", number[n - 1]);

return 0;

}

要求：从键盘输入10个整数存入一个数组中，用函数编程将其中的最大数与最小数位置互换，然后输出互换后的数组。函数原型：void MaxMinExchang(int a[], int n);

输入提示信息："Input 10 numbers:"

输入格式："%d"

输出提示信息："Exchang results:"

输出格式："%4d"

程序运行结果示例：

Input 10 numbers:8 2 7 4 9 11 5 47 6 97↙

Exchang results: 8 97 7 4 9 11 5 47 6 2

#include <stdio.h>

void MaxMinExchang(int a[], int n);

int FindMaxPos(int s[], int n);

int FindMinPos(int s[], int n);

int main()

{

int i, a[10];

printf("Input 10 numbers:");

for (i = 0; i < 10; i++)

{

scanf("%d", &a[i]);

}

MaxMinExchang(a, 10);//1

printf("Exchang results:");

for (i = 0; i < 10; i++)

{

printf("%4d", a[i]);//1

}

printf("\n");

return 0;

}

/\* 函数功能：将数组中的最大数与最小数位置互换 \*/

void MaxMinExchang(int a[], int n)

{

int maxPos, minPos, temp;

maxPos = FindMaxPos(a, n);//1

minPos = FindMinPos(a, n);//1

temp = a[maxPos];

a[maxPos] = a[minPos];//1

a[minPos] = temp;

}

/\* 函数功能：计算数组中的最大值在数组中的下标位置 \*/

int FindMaxPos(int s[], int n)//1

{

int maxPos = 0, max = s[0], i;

for (i = 1; i < n; i++)

{

if (s[i] > max) //1

{

max = s[i];

maxPos = i;

}

}

return maxPos;

}

/\* 函数功能：计算数组中的最小值在数组中的下标位置 \*/

int FindMinPos(int s[], int n)//1

{

int minPos = 0, min = s[0], i;

for (i = 1; i < n; i++)

{

if (s[i] < min)//1

{

min = s[i];

minPos = i;

}

}

return minPos;

}

下面程序的功能是从键盘任意输入n个数，然后找出其中的最大数与最小数，并将其位置对换。

该程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

#include <stdio.h>

#define n = 10;

void MaxMinExchang(int \*a, int n)

{

int maxValue = \*a, minValue = \*a, maxPos, minPos;

int i, temp;

for (i=0; i<n; i++)

{

if (\*a+i > maxValue)

{

maxValue = \*a+i;

maxPos = i;

}

if (\*a+i < minValue)

{

minValue = \*a+i;

minPos = i;

}

}

temp = \*a + maxPos;

\*a + maxPos = \*a + minPos;

\*a + minPos = temp;

}

main()

{

int a[n], i, n;

printf("Input n(n<=10):\n");

scanf("%d", n) ;

printf("Input %d Numbers:\n", n);

for (i=0; i<n; i++)

{

scanf("%d", a[i]);

}

MaxMinExchang(a, n);

printf("After MaxMinExchange:\n");

for (i=0; i<n; i++)

{

printf("%d ", a[i]);

}

printf("\n");

}

把如下两个无序数组a和b从小到大排序后，在按照从小到大的顺序一次存放到新的数组中。

int a[5]={9,78,33,12,23};

int b[8]={1,34,63,10,5,94,39,27};

\*\*输出格式要求："%4d"

程序运行示例为：

9

12

23

33

78

1

5

10

27

34

39

63

94

1 5 9 10 12 23 27 33 34 39 63 78 94

编程输入10个数，找出其中的最大值及其所在的数组下标位置。找出其中错误并改正之。

#include <stdio.h>

int FindMax(int num[], int n, int \*pMaxPos);

main()

{

int num[10], maxValue, maxPos, minValue, minPos, i;

printf("Input 10 numbers:\n ");

for (i=0; i<10; i++)

{

scanf("%d", num[i]);

}

maxValue = FindMax(num, 10, \*maxPos);

printf("Max=%d, Position=%d\n",maxValue, maxPos);

}

int FindMax(int num[], int n, int \*pMaxPos)

{

int i, max;

max = num[0];

for (i = 1, i < n, i++)

{

if (num[i] > max)

{

max = num[i];

\*pMaxPos = i;

}

}

return max;

}

#include <stdio.h>

int FindMax(int num[], int n, int \*pMaxPos);

main()

{

int num[10], maxValue, maxPos, minValue, minPos, i;

printf("Input 10 numbers:\n ");

for (i=0; i<10; i++)

{

scanf("%d", &num[i]);

}

maxValue = FindMax(num, 10, &maxPos);

printf("Max=%d, Position=%d\n",maxValue, maxPos);

}

int FindMax(int num[], int n, int \*pMaxPos)

{

int i, max;

max = num[0];

\*pMaxPos = 0;

for (i = 1; i < n; i++)

{

if (num[i] > max)

{

max = num[i];

\*pMaxPos = i;

}

}

return max;

}

用函数编程统计平均成绩。要求在主函数中输入学生成绩，输入负数时,结束输入，调用子函数Average()计算平均成绩，并输出结果。

函数原型：int Average(int score[],int n)

\*\*\*\*要求输入提示信息为："Input score: (输入负数结束)\n" (只提示一次）

\*\*\*\*输出格式要求为：

"Total students are %d\n"

"Average score is %d\n"

程序运行示例：

Input score: (输入负数结束)

50 60 70 -1

Total students are 3

Average score is 60

#include <stdio.h>

#define N 40

int Average(int score[],int n)

{

int i,sum=0;

for(i=0;i<n;i++)

{

sum=sum+score[i]; //1

}

return sum/n;//1

}

main()

{

int score[N],m;

int i=-1;

printf("Input score: (输入负数结束)\n");

do{

i++;

scanf("%d",&score[i]);

}while(score[i]>=0);//e2

printf("Total students are %d\n",i);//1

m=Average(score,i);//2

printf("Average score is %d\n",m);//1

}

#include <stdio.h>

void sort(int a[], int n)

{

int i, j, t, \*p;

for (i = 0; i < n - 1; i++)

{

p = a + i;

for (j = i + 1; j < n; j++)

{

if (\*p > \*(a + j))

p = a + j;

}

if (p != (a + i))

{

t = \*(a + i);

\*(a + i) = \*p;

\*p = t;

}

}

}

void addins(int a[], int b[], int c[], int n, int m)

{

int i, j, k, l;

int \*p1 = a, \*p2 = b , \*p3 = c;

i = j = k = 0;

while (i < n && j < m)

{

if (\*(p1 + i) <= \*(p2 + j))

{

\*(p3 + k) = \*(p1 + i);

i++;

}

else

{

\*(p3 + k) = \*(p2 + j);

j++;

}

k++;

}

if (i == n)

{

for (l = j; l < m; l++)

{

\*(p3 + k) = \*(p2 + l);

k++;

}

}

if (j == m)

{

for (l = i; l < n; l++)

{

\*(p3 + k) = \*(p1 + l);

k++;

}

}

}

main()

{

int a[5] = {9, 78, 33, 12, 23};

int b[8] = {1, 34, 63, 10, 5, 94, 39, 27};

int c[13];

int i, n, m;

int \*p1 = a, \*p2 = b, \*p3 = c;

sort(a, 5);

for (i = 0; i < 5; i++)

{

printf("%4d\n", \*(p1 + i));

}

sort(b, 8);

for (i = 0; i < 8; i++)

{

printf("%4d\n", \*(p2 + i));

}

addins(b, a, c, 8, 5);

for (i = 0; i < 13; i++)

{

printf("%4d", \*(p3 + i));

}

}

按如下函数原型编写程序，用一个整型数组feedback保存调查的40个反馈意见。用函数编程计算反馈意见的平均数（Mean）、中位数（Median）和众数（Mode）。中位数指的是排列在数组中间的数。如果原始数据的个数是偶数，那么中位数等于中间那两个元素的算术平均值。众数是数组中出现次数最多的那个数（不考虑两个或两个以上的反馈意见出现次数相同的情况）。

int Mean(int answer[], int n);

int Median(int answer[], int n);

int Mode(int answer[], int n);

void DataSort(int a[], int n);

输入提示信息："Input the feedbacks of 40 students:\n"

输入格式："%d"

输出提示信息和输出格式：

"Mean value = %d\n"

"Median value = %d\n"

"Mode value = %d\n"

#include <stdio.h>

#define M 40

#define N 11

int Mean(int answer[], int n);

int Median(int answer[], int n);

int Mode(int answer[], int n);

void DataSort(int a[], int n);

int main()

{

int i, feedback[M]; //1

printf("Input the feedbacks of 40 students:\n"); //1

for (i = 0; i < M; i++) //1

{

scanf("%d", &feedback[i]); //1

}

printf("Mean value = %d\n", Mean(feedback, M)); //1

printf("Median value = %d\n", Median(feedback, M)); //1

printf("Mode value = %d\n", Mode(feedback, M)); //1

return 0;

}

/\* 函数功能：若n>0则计算并返回n个数的平均数，否则返回-1 \*/

int Mean(int answer[], int n)

{

int i, sum = 0; //1

for (i = 0; i < n; i++) //1

{

sum += answer[i];//1

}

return n > 0 ? sum / n : -1;

}

/\* 函数功能：计算n个数的中位数 \*/

int Median(int answer[], int n)

{

DataSort(answer, n); //1

if (n % 2 == 0) //1

return (answer[n / 2] + answer[n / 2 - 1]) / 2;

else

return answer[n / 2];

}

/\* 函数功能：计算n个数的众数 \*/

int Mode(int answer[], int n)

{

int i, grade, max = 0, modeValue = 0, count[N] = {0};//1

for (i = 0; i < n; i++)

{

count[answer[i]]++;//1

}

for (grade = 1; grade <= N - 1; grade++)//1

{

if (count[grade] > max)//1

{

max = count[grade];//1

modeValue = grade;

}

}

return modeValue;

}

/\* 函数功能：按选择法对数组a中的n个元素进行排序 \*/

void DataSort(int a[], int n)

{

int i, j, k, temp;//1

for (i = 0; i < n - 1; i++)//1

{

k = i;//1

for (j = i + 1; j < n; j++)//1

{

if (a[j] > a[k]) k = j;//1

}

if (k != i)//1

{

temp = a[k];//1

a[k] = a[i];//1

a[i] = temp;//1

}

}

}

一维数组中等

13个人围成一圈，从第1个人开始顺序报号1、2、3，凡报到3的人退出圈子。找出最后留在圈子里的人原来的序号

#include <stdio.h>

#define N 13

struct person

{

int number;

int nextp;

} link[N + 1];

int main(void)

{

int i, count, h;

/\* 建立队列 \*/

for (i = 1; i <= N; i++)

{

if (i == N)

{

link[i].nextp = 1;

}

else

{

link[i].nextp = i + 1;

}

link[i].number = i;

}

printf("\n");

count = 0;

h = N;

printf("出圈成员及顺序：");

while (count < N - 1)

{

i = 0;

while (i != 3)

{

h = link[h].nextp;

if (link[h].number)

{

i++;

}

}

printf("%3d", link[h].number);

link[h].number = 0;

count++;

}

printf("\n最后的成员是：");

for (i = 1; i <= N; i++)

{

if (link[i].number)

{

printf("%3d", link[i].number);

}

}

return 0;

}

按照筛法求素数的思想求出100以内的所有素数。每行10个。

#include <stdio.h>

#include <stdlib.h>

int main()

{

int i,j,k=1,a[100];

for(i=2;i<=100;i++)

{

for(j=i/2;j>1;j--)

if(i%j==0) break;

if(j==1)

{

a[k]=i;

k++;

}

}

for(i=1;i<k;i++)

{

printf("%5d",a[i]);

if(i%10==0) printf("\n");

}

return 0;

}

编写程序，输入一个以回车符结束的字符串（少于80个字符），过滤去所有的非十六进制字符后，组成一个新字符串（十六进制形式），然后将其转换为十进制数后输出。

#include <stdio.h>

int main(void)

{

char t[80], ch;

int i, len;

double x;

i = 0;

printf("请输入十六进制字符串：");

while ((ch = getchar()) != '\n')

{

if ((ch >= '0' && ch <= '9')

|| (ch >= 'a' && ch <= 'f')

|| (ch >= 'A' && ch <= 'F'))

{

t[i] = ch;

i++;

}

}

len = i;

t[i] = '\0';

printf("十六进制串%s的值=十进制", t);

x = 0.0;

for (i = 0; i < len; i++)

{

if (t[i] >= '0' && t[i] <= '9')

{

x = x \* 16 + (t[i] - '0');

}

else if (t[i] >= 'a' && t[i] <= 'f')

{

x = x \* 16 + (t[i] - 'a' + 10);

}

else if (t[i] >= 'A' && t[i] <= 'F')

{

x = x \* 16 + (t[i] - 'A' + 10);

}

}

printf("%.0f\n", x);

return 0;

}

从键盘任意输入某班30个学生的成绩(成绩类型为整型)，保存到数组中，并输出成绩按由高到低的排序结果。

#include <stdio.h>

main()

{

int score[30], temp;

int i, j;

printf("Please enter scores:\n");

for (i = 0; i < 30; i++)

{

scanf("%d", &score[i]);

}

for (i = 0; i < 29; i++)

{

for (j = i + 1; j < 30; j++)

{

if (score[j] > score[i])

{

temp = score[j];

score[j] = score[i];

score[i] = temp;

}

}

}

printf("Sorted results:\n");

for (i = 0; i < 30; i++)

{

printf("%d\n", score[i]);

}

}

从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

#include "stdio.h"

#include "string.h"

main()

{

int i=0, findFlag=1;

char x[13];

char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str[i])==0)

{

findFlag = 0;

}

i++;

}

if ( !findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

从键盘输入10个不同的整数存入一个数组中，用函数编程将其中的最大数与最小数位置互换，分别输出互换之前最大值和最小值及其在数组中的位置,以及互换后的数组

#include <stdio.h>

#include <stdlib.h>

int main()

{

int i, a[10], max, min, maxPos = 0, minPos = 0, temp;

printf("Input 10 numbers:");

for (i = 0; i < 10; i++)

{

scanf("%d", &a[i]);

}

max = a[0];

min = a[0];

for (i = 1; i < 10; i++)

{

if (a[i] > max) //1

{

max = a[i];

maxPos = i;

}

if (a[i] < min) //1

{

min = a[i];

minPos = i;

}

}

printf("Max element position:%d,Max element:%d\n", maxPos, max); //1

printf("Min element position:%d,Min element:%d\n", minPos, min); //1

temp = a[maxPos];

a[maxPos] = a[minPos];

a[minPos] = temp;

printf("Exchange results:"); //1

for (i = 0; i < 10; i++)

{

printf("%4d", a[i]);//1

}

printf("\n");

return 0;

}

从键盘输入10个整数存入一个数组中，用函数编程将其中的最大数与最小数位置互换，然后输出互换后的数组。函数原型：void MaxMinExchang(int a[], int n);

#include <stdio.h>

void MaxMinExchang(int a[], int n);

int FindMaxPos(int s[], int n);

int FindMinPos(int s[], int n);

int main()

{

int i, a[10];

printf("Input 10 numbers:");

for (i = 0; i < 10; i++)

{

scanf("%d", &a[i]);

}

MaxMinExchang(a, 10);//1

printf("Exchang results:");

for (i = 0; i < 10; i++)

{

printf("%4d", a[i]);//1

}

printf("\n");

return 0;

}

/\* 函数功能：将数组中的最大数与最小数位置互换 \*/

void MaxMinExchang(int a[], int n)

{

int maxPos, minPos, temp;

maxPos = FindMaxPos(a, n);//1

minPos = FindMinPos(a, n);//1

temp = a[maxPos];

a[maxPos] = a[minPos];//1

a[minPos] = temp;

}

/\* 函数功能：计算数组中的最大值在数组中的下标位置 \*/

int FindMaxPos(int s[], int n)//1

{

int maxPos = 0, max = s[0], i;

for (i = 1; i < n; i++)

{

if (s[i] > max) //1

{

max = s[i];

maxPos = i;

}

}

return maxPos;

}

/\* 函数功能：计算数组中的最小值在数组中的下标位置 \*/

int FindMinPos(int s[], int n)//1

{

int minPos = 0, min = s[0], i;

for (i = 1; i < n; i++)

{

if (s[i] < min)//1

{

min = s[i];

minPos = i;

}

}

return minPos;

}

改错题：输入一行字符，统计其中的英文字符、数字字符、

#include <stdio.h>

#define ARR\_SIZE 80

main()

{

char str[ARR\_SIZE];

int len=0, i=0, letter=0, digit=0, space=0, others=0;

scanf("%s",&str);

len = strlen(str);

for (i=0; i<len; i++)

{

if ('a' <= str[i] &&str[i]<= 'z' || 'A' <= str[i]&& str[i]<= 'Z')

letter ++;

else if ('0' <= str[i]&& str[i]<= '9')

digit ++;

else if (str[i] == ' ' )

space ++;

else

others ++;

}

printf("English character: %d\n", letter);

printf("digit character: %d\n", digit);

printf("space: %d\n", space);

printf("other character: %d\n", others);

}

数字排列 在较中的九个顶点上，空出中间的点，其余的点上任意填入数字1至8；1的位置固定不动，然后移动其余数字，使1到8顺时针从小到大排列。移动的规则是：只能将数字沿线移向空白的点。请编程显示数字移动过程。

int a[9]={0,1,2,5,8,7,6,3};

int b[9];

int c[9];

int count=0;

main()

{int i,j,k,t;

void print();

printf("Please enter original order of digits 1~8:");

for (i=0;i<8;i++)

scanf("%d",&b[a[i]]);

printf("The sorting process is as felow:\n");

print();

for(t=-1,j=0;j<8 && t==-1;j++)

if(b[a[j]==1]) t=j;

for(j=0;j<8;j++)

c[j]=a[(j+t)%8];

for(i=2;i<9;i++)

for(j=i-1;j<8;j++)

if(b[c[j]]==i&&j!=i-1)

{ b[4]=i;

b[c[j]]=0;

print();

for (k=j;k!=i-1;k--)

{

b[c[k]]=b[c[k-1]];

b[c[k-1]]=0;

print();

}

b[c[k]]=i;

b[4] =0;

print();

break;

}

else if(b[c[j]==i])

break;

}

void print(void)

{ int c;

for(c=0;c<9;c++)

if(c%3==2)

printf("%2d\n",b[c]);

else

printf("%2d",b[c]);

printf("---%2d---\n",count++);

}

下面程序实现折半查找算法，当找到输入元素后显示其在数组中的下标。找出其中的错误，并改正之

#include <stdio.h>

main()

{

int up=9, low=0, mid, found=0, find;

int a[10]={1, 5, 6, 9, 11, 17, 25, 34, 38, 41};

scanf("%d", &find);

printf("\n");

while (up>low && !found)

{

mid=(up+low)/2;

if( a[mid] == find )

{

found=1;

break;

}

else if(a[mid]>find)

up=mid;

else

low=mid;

}

if(found) printf("found number is %dth", mid);

else printf("no found");

}

下面的两个数组代表生产8个产品时固定和可变的成本：写一个程序输入编号为1到8的产品号以及生产的数量，输出生产这种产品的总成本。

#include <stdio.h>

int main()

{

float fixed[] = { 11.31, 12.12, 13.67, 11.91, 12.30, 11.8, 11.00, 12.00 } ;

float variable[] = { 1.12, 1.13, 3.14, 1.35, 2.20, 1.28, 1.00, 2.10 } ;

int i, n;

printf("请输入产品号（1-8）：");

scanf("%d", &i);

printf("请输入该种产品的生产数量：");

scanf("%d", &n);

printf("生产编号为%d，数量为%d的产品的总成本为%.2f\n", i, n, fixed[i - 1] + n \* variable[i - 1]);

return 0;

}

用结构体实现输入4个学生的学号（无符号整数）、姓名（字符串）和考试成绩（整数），然后按从高分到低分的顺序排列后输出。

#include <stdio.h>

#define N 4

struct Student1

{

unsigned int sno;

char sname[16];

int score;

} ;

int main( )

{

struct Student1 stu[N]; //1

int i, j;

struct Student1 temp;

for (i = 0; i < N; i++) //1

{

printf("输入第%d个学生的信息：学号、姓名、成绩\n", i + 1); //1

scanf("%d %s %d", &stu[i].sno, stu[i].sname, &stu[i].score);//1

}

for (i = 0; i < N - 1; i++) //1

{

for (j = 0; j < N - i - 1; j++)//1

{

if (stu[j].score < stu[j + 1].score)//1

{

temp = stu[j];

stu[j] = stu[j + 1]; //1

stu[j + 1] = temp;

}

}

}

printf("排序后学生的信息为:\n");

for (i = 0; i < N; i++) //1

{

printf("%d %s %d\n", stu[i].sno, stu[i].sname, stu[i].score);//1

}

return 0;

}

有n个人围成一圈，顺序排号。从第一人开始报数（从1到3报数），凡报到3的人退出圈子，问最后留下的是原来第几号的那位。

#include <stdio.h>

#define NMAX 50 /\* 定义最多人数为50人 \*/

int main(void)

{

int i, k, m, n, num[NMAX], \*p;

printf("请输入人数：");

scanf("%d", &n);

p = num;

for (i = 0; i < n; i++)

{

\*(p + i) = i + 1; /\* 以1至n为序给每个人编号 \*/

}

i = 0;

k = 0; /\* k为按1,2,3报数时的计数变量 \*/

m = 0; /\* m为退出人数 \*/

while (m < n - 1)

{

if (\*(p + i) != 0)

{

k++;

}

if (k == 3)

{

\*(p + i) = 0; /\* 对退出人员的编号置为0 \*/

k = 0;

m++;

}

i++;

if (i == n)

{

i = 0; /\* 报数到尾后i恢复为0 \*/

}

}

while (\*p == 0)

{

p++;

}

printf("%d个人中最后留下的是第%d号。", n, \*p);

return 0;

}

编写程序，把一维数组a 输入任意6个整数，假设a 为： 7 4 5 9 1 2

#include <stdio.h>

#include <stdlib.h>

int main()

{

int a[6],i,j,temp;

for(i=0;i<6;i++)

scanf("%d",&a[i]);

printf("\n");

for(j=0;j<6;j++)

{

for(i=0;i<6;i++)

{

printf("%d ",a[i]);

}

printf("\n");

temp=a[5];

for(i=5;i>0;i--)

{

a[i]=a[i-1];

}

a[0]=temp;

}

return 0;

}

Two Bags of Potatoes

#include <stdio.h>

#include <string.h>

#define M 100005

int ans[M];

int main()

{

int n, y, k;

(scanf("%d%d%d", &y, &k, &n));

{

int i;

int cnt = 0;

for (i = 1; i \* k <= n; i++)

{

if (i \* k > y)

ans[cnt++] = i \* k - y;

}

if (cnt == 0)

{

printf("-1\n");

return 0;

//continue;

}

printf("%d", ans[0]);

for (i = 1; i < cnt; i++)

printf(" %d", ans[i]);

printf("\n");

}

return 0;

}

从键盘任意输入一个字符串（字符串可以有空格），在给定的一组字符串中寻找该输入字符串，

#include "stdio.h"

#include "string.h"

main()

{

int i=0, findFlag=1;

char x[13];

char \*str[13]= { "Pascal","Basic","Fortran", "Java","Visual C", "Visual Basic" };

printf("Input string:\n");

gets(x);

while( i<6 && findFlag)

{

if(strcmp(x,str[i])==0)

{

findFlag = 0;

}

i++;

}

if ( !findFlag )

{

printf("%s\n",x);

}

else

{

printf("Not find!\n");

}

}

不用标准库函数strcmp，自己编写一个函数MyStrcmp实现将两个字符串s和t进行比较，然后将s和t中第一个不相同字符的ASCII码值之差作为函数值返回。要求如下： (1)函数MyStrcmp函数原型为： int MyStrcmp(char s[],char t[]) (2)在主函数中从键盘分别输入两个字符串s和t， (每个字符串的最大长度为80个字符) 然后调用MyStrcmp函数，接下来，判定该函数的返回值：如果大于0，输出"string s>string t.\n" 如果小于0，输出"string s<string t.\n" 如果等于0，输出"string s=string t.\n" (3)\*\*用gets输入字符串。 \*\*输入提示信息：输入s之前提示："Input s\n" 输入t之前提示："Input t\n"

#include <stdio.h>

#define SIZE 80

**int**MyStrcmp(**char**s[],**char**t[]);

main()

{

**char**s[SIZE],t[SIZE],i;

**printf**("Input s\n");

**gets**(s);//1

**printf**("Input t\n");

**gets**(t);//1

    i=MyStrcmp(s,t);//2

**if**(i>0)//1

**printf**("string s>string t.\n");

**elseif**(i<0)//1

**printf**("string s<string t.\n");

**else**//1

**printf**("string s=string t.\n");

}

**int**MyStrcmp(**char**s[], **char**t[])

{

**int**i;

**for**(i=0; s[i] == t[i]; i++ )//6

    {

**if**(s[i] ==  '\0')  //2

**return**0 ;//2

    }

**return**(s[i]-t[i]); //2

}

从键盘输入一个字符串（最长不超过80字符），然后从键盘输入一个字符，删除字符串中与该字符相同的字符。其中，在字符串中删除与某字符相同的字符，要求用字符数组作函数参数编程实现。已知该函数的原型为： void Squeeze(char s[], char c); 程序运行结果示例： Input a string: hello, my friend!↙ Input a character: !↙Results:hello, my friend 输入格式: 输入字符串用 gets()函数输入单个字符用getchar()函数输出格式：输入字符串的提示信息： "Input a string:\n" 输入单个字符的提示信息： "Input a character:\n" 输出格式： "Results:%s\n" 注意：不能使用指针、结构体和goto语句编程。

#include <stdio.h>

**void**  Squeeze(**char**s[], **char**c);// 2

**int**main()

{

**char**  str[20], ch;

**printf**("Input a string:\n");  //1

**gets**(str);                    //1

**printf**("Input a character:\n");//1

    ch = **getchar**();

    Squeeze(str, ch);      //2

**printf**("Results:%s\n", str);//1

**return**0;

}

**void**  Squeeze(**char**s[], **char**c)//2

{

**int**i, j;

**for**(i = j = 0; s[i] != '\0'; i++)//1

    {

**if**(s[i] != c)//1

        {

            s[j] = s[i];//1

            j++;//1

        }

    }

    s[j] = '\0'; //1

}

下面程序的功能是删除字符串中第一次出现的a字符。找出其中错误，并改正之。

#include <stdio.h>

#include <string.h>

**void**fun(**char**\*x,**int**n,**int**\*t)

{  **int**i,k=0;

    x[n]='a';x[n+1]='\0';

**while**(x[k]!='a') k++;

**if**(k==n){\*t=0;}

**else**

    {  **for**(i=k;i<n;i++)

            x[i]=x[i+1];

        x[i-1]='\0';

    }

}

main()

{

**char**x[20];

**int**len,t;

**gets**(x);

**puts**(x);

    len=**strlen**(x);

    fun(x,len,&t);

**if**(t==0) **printf**("Not exist!\n");

**else**    **puts**(x);

}

下面的函数MyStrcmp()用于实现函数strcmp()的功能，将两个字符串s和t进行比较，要求将两个字符串中第一个不相同字符的ASCII码值之差作为MyStrcmp()函数的返回值。找出其中错误并改正之。

#include <stdio.h>

**int**MyStrcmp(**char**s[], **char**t[]);

main()

{

**char**  str1[20],str2[20];

**printf**("Input string:");

**gets**(str1);

**printf**("Input another string:");

**gets**(str2);

**if**(MyStrcmp(str1,str2) > 0)

    {

**printf**("str1 大于 str2\n");

    }

**elseif**(MyStrcmp(str1,str2) < 0)

    {

**printf**("str1 小于 str2\n");

    }

**else**

    {

**printf**("str1 等于 str2\n");

    }

}

**int**MyStrcmp(**char**s[], **char**t[])

{

**int**i;

**for**(i=0; s[i] == t[i]; i++)

    {

**if**(s[i] == '\0')  **return**0 ;

    }

**return**(s[i] - t[i]);

}

编程判断输入的一个字符串是否是“回文”。所谓“回文”字符串就是从左读和从右读都一样的字符串。例如: "abcba"就是一个回文字符串。要求：先输出("Input a string:\n")，输入字符串，然后判断是否回文，最后输出 ("This string is a plalindrome.\n")或者("This string is not a plalindrome.\n") 下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。

#include <stdio.h>

#include <string.h>

**int**main()

{

**char**str[100];

**int**i, j;

**printf**("Input a string:\n");

**gets**(str);

    i = 0;

    j = **strlen**(str) - 1;

**while**((i< j) && (str[i] == str[j]))

    {

        i++;

        j--;

    }

**if**(i< j)

    {

**printf**("This string is not a plalindrome.\n");

    }

**else**

    {

**printf**("This string is a plalindrome.\n");

    }

**return**0;

}

该程序的作用是将一个字符串中的元音字母复制到另一个字符串，然后输出。改正错误。

#include<stdio.h>

**void**Cpy(**char**s[], **char**c[]);

main()

{

**char**str1[80], str2[80];

**printf**("输入一个字符串\n");

**gets**(str1);

    Cpy(str1, str2); //改为：Cpy(str1, str2)

**printf**("输出一个字符串\n");

**puts**(str2);

}

**void**Cpy(**char**s[], **char**c[])

{

**int**i, j = 0; //改为：j=0;

**for**(i = 0; s[i] != '\0'; i++)

    {

**if**(s[i] == 'a'|| s[i] == 'A'|| s[i] == 'e'|| s[i] == 'E'||

            s[i] == 'i'|| s[i] == 'I'|| s[i] == 'o'|| s[i] == 'O'||

            s[i] == 'u'|| s[i] == 'U')

            //将=改为：==

        {

            c[j] = s[i]; //改为:c[j] = s[i];

            j++;

        }

    }

    c[j] = '\0';

}

用指针做函数参数自己编程实现字符串连接函数strcat()的功能。下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。经教师手工核对后，如果未用指针做函数参数编程，那么即使做对也不给分。改错时不能改变程序原有的意图，不能改变函数原型。

#include <stdio.h>

#define N  80

**void**MyStrcat(**char**\*dstStr, **char**\*srcStr);

**int**main()

{

**char**s[N], t[N];

**printf**("Input a string:\n");

**gets**(s);

**printf**("Input another string:\n");

**gets**(t);

    MyStrcat(s, t);

**printf**("Concatenate results:%s\n", s);

**return**0;

}

**void**MyStrcat(**char**\*dstStr, **char**\*srcStr)

{

**while**(\*dstStr != '\0')

    {

        dstStr++;

    }

**while**(\*srcStr != '\0')

    {

        \*dstStr = \*srcStr;

        srcStr++;

        dstStr++;

    }

    \*dstStr = '\0';

}

下面程序的Squeeze函数的功能是删除字符串s中所出现的与变量c相同的字符。请改正程序中的错误，使它能得出正确的结果。

#include <stdio.h>

**void**Squeeze(**char**s[], **char**c);

main()

{

**char**a[80], c;

**gets**(a);

**scanf**("%c", &c);

    Squeeze(a, c);

**printf**("%s\n", a);

}

**void**Squeeze(**char**s[], **char**c)

{

**int**i, j;

**for**(i = 0, j = 0; s[i] != '\0'; i++)

    {

**if**(s[i] != c)

        {

            s[j] = s[i];

            j++;

        }

    }

    s[j] = '\0';

}

输入一行字符（第一个字符有可能是空格），用函数编程统计其中有多少单词。假设单词之间以空格分开。要求如下： （1）在主函数中从键盘输入字符串，字符串的最大长度为80个字符 调用CountWords函数，并统计字符串中的单词个数。 （2）在子函数CountWords中统计字符串中的单词个数。函数原型为： int CountWords(char str[]); (3)\*\*输入提示信息："Input a string:" \*\*输出提示信息和格式："Numbers of words = %d\n" \*\*用gets()输入字符串 注：不能使用指针、结构体、共用体、文件、goto、枚举类型进行编程。

#include <stdio.h>

**int**CountWords(**char**str[]);

**int**main()

{

**char**  str[80];

**printf**("Input a string:");

**gets**(str);//1

**printf**("Numbers of words = %d\n", CountWords(str));//1

**return**0;

}

**int**CountWords(**char**str[])//1

{

**int**    i, num;

    num = (str[0] != ' ') ? 1 : 0;//1

**for**(i = 1; str[i] != '\0'; i++)//1

    {

**if**(str[i] != ' '&& str[i - 1] == ' ')//1

        {

            num++;//1

        }

    }

**return**num;//1

}

有一个字符串，包含n个字符。写一个函数，将此字符串中从第m个字符开始的全部字符复制成为另一个字符串。 程序的运行示例如下： 请输入一个字符串： 1234567890 从第几个字符开始复制？3 复制结果为：34567890

#include <stdio.h>

copystr(**char**\*p1, **char**\*p2, **int**m)

{

**int**n;

    n = 0;

**while**(n < m - 1)

    {

        n++;

        p1++;

    }

**while**(\*p1 != '\0')

    {

        \*p2 = \*p1;

        p1++;

        p2++;

    }

    \*p2 = '\0';

}

**int**main(**void**)

{

**int**m;

**char**str1[20], str2[20];

**printf**("请输入一个字符串：\n");

**scanf**("%s", str1);

**printf**("从第几个字符开始复制？");

**scanf**("%d", &m);

**if**(**strlen**(str1) < m)

    {

**printf**("输入错误。");

    }

**else**

    {

        copystr(str1, str2, m);

**printf**("复制结果为：%s", str2);

    }

**return**0;

}

调用GetInput()函数从键盘任意输入一个字符串，然后分别在GetInput()和main()中输出该字符串。 注意： （1）请将修改正确后的完整源程序拷贝粘贴到答题区内。 （2）对于没有错误的语句，请不要修改， 修改原本正确的语句也要扣分。 （3）当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

#include  <stdio.h>

**void**GetInput(**char**\* s);

**int**main(**void**)

{

**char**str[80];

**char**\*ptr = str;

   GetInput(ptr);

**puts**(ptr);

**return**0;

}

**void**GetInput(**char**\* s)

{

**scanf**("%s", s);

**puts**(s);

}

自己定义一个函数，用字符数组作为函数参数的类型， 以实现和函数strlen相同的功能。 要求： （1）按如下函数原型进行编程： int Mystrlen(char str[]); （2）在主函数中： 任意输入一个字符串（长度不超过80个字符）; 调用函数Mystrlen()计算输入字符串的实际长度; 然后打印计算结果。 \*\*要求输入提示信息为："Please input the string:\n" \*\*要求输出格式为:"%d\n" 注意： （1）字符串中的空格计入字符串实际长度， 但是字符串结束标志不计入字符串实际长度。 （2）必须使用函数编程。否则没有成绩。

#include <stdio.h>

**int**Mystrlen(**char**str[]);  //2

main()

{

**char**str[80];

**int**len;

**printf**("Please input the string:\n");  //2

**gets**(str);  //2

    len = Mystrlen(str);  //2

**printf**("%d\n", len);  //2

}

**int**Mystrlen(**char**str[])

{

**int**len = 0, i;

**for**(i=0; str[i]!=0; i++)  //2

    {

        len++;  //2

    }

**return**len;  //2

}

纯数字字符串检验 按给定函数原型编程检查一个字符串是否全由数字组成。 int IsAllDigit(char p[]);/\*若全由数字组成，则函数返回1，否则返回0\*/ 在主函数中，从键盘输入一个字符串（假设字符串的最大长度为20个字符），调用函数IsAllDigit()，检查该字符串是否全由数字组成，然后在主函数中根据函数IsAllDigit()的返回值输出相应的提示信息。 程序运行结果示例1： Please input a string: help456↙ The string is not a digit string. 程序运行结果示例2： Please input a string: 20150216↙ The string is a digit string. 输入格式: 字符串输入采用 gets()函数 字符串输入提示信息："Please input a string:\n" 输出格式： 判断是纯数字字符串："The string is a digit string." 判断不是纯数字字符串："The string is not a digit string."

#include<stdio.h>

**int**IsAllDigit(**char**p[]);

**int**main()

{

**char**a[80];

**printf**("Please input a string:\n");

**gets**(a);

**if**(IsAllDigit(a))

**printf**("The string is a digit string.");

**else**

**printf**("The string is not a digit string.");

}

**int**IsAllDigit(**char**p[])

{

**int**flag = 1, i;

**for**(i = 0; p[i] != '\0'; i++)

    {

**if**(p[i] <'0'|| p[i] >'9')

        {

            flag = 0;

**break**;

        }

    }

**return**flag;

}

程序改错。编写一个函数Inverse()，实现将字符数组中的字符串逆序存放的功能。（4个错误）

#include<string.h>

#include<stdio.h>

#define ARR\_SIZE 80

**void**Inverse(**char**str[]);

main()

{

**char**a[ARR\_SIZE] ;

**printf**("Please enter a string: ");

**gets**(a);

    Inverse(a);

**printf**("The inversed string is: ");

**puts**(a);

}

**void**Inverse(**char**str[])

{

**int**   len, i, j;

**char**  temp;

    len = **strlen**(str);

**for**(i=0, j=len-1; i<j; i++, j--)

    {

    temp = str[i];

        str[i] = str[j];

        str[j] = temp;

    }

}

下面的程序判断是不是回文，是回文返回1，否则返回0。回文是顺读和倒读都一样。请改正程序中的错误，使它能得出正确的结果。

#include <string.h>

#include <stdio.h>

hui(**char**\*a)

{

**int**i, k, j = 0;

**for**(i = 0; i< 15; i++)

**if**(a[i] != '\0')

            j++;

**else**

**break**;

**if**(j % 2 == 0)

**return**0;

**for**(i = 0, k = j - 1; i<= (j - 1) / 2; i++, k--)

**if**(a[i] != a[k])

**return**0;

**return**1;

}

main()

{

**int**x;

**char**a[15];

**gets**(a);

**puts**(a);

    x = hui(a);

**if**(x == 1)

**printf**("this is huiwen!!!");

**else**

**printf**("this is not huiwen!!!");

}

输入一个字符，如果此字符在已知的字符串中出现，则删除此字符(只删除第一个) ，如删除成功就返回被删字符，否则返回'\0'。请改正程序中的错误，使它能得出正确的结果。 程序运行结果示例1： The Original string is:seven↙ Enter x: e The deleted charactor is :e↙ The Original string changes to :sven↙ ↙ 程序运行结果示例2： The Original string is:seven↙ Enter x: a not found!!!↙ The Original string changes to :seven↙ ↙

#include <string.h>

#include <stdio.h>

**char**sh(**char**\*a, **char**x)

{

**int**i = 0, k, j;

**char**q;

**while**(a[i] != '\0') i++;

**for**(j = 0; j <i; j++)

    {

**if**(a[j] == x)

        {

            q = a[j];

**for**(k = j; k <i; k++)

                a[k] = a[k + 1];

**return**q;

        }

    }

**return**'\0';

}

main()

{

**char**a[6] = "seven", x, ch;

**printf**("The Original string is:");

**puts**(a);

**printf**("Enter x: ");

**scanf**("%c", &x);

    ch = sh(a, x);

**if**(ch == '\0')

**printf**("%c not found!!!", x);

**else**

    {

**printf**("The deleted charactor is :");

**putchar**(x);

    }

**printf**("\nThe Original string changes to :");

**puts**(a);

}

下列给定程序中，函数fun()的功能是：从s所指字符串中，找出t所指字符串的个数作为函数值返回。如，当s所指字符串中的内容为abcdabfab，t所指字符串的内容为ab，则函数返回整数3。请改正程序中的错误，使它得出正确的结果：

#include <stdio.h>

int fun(char \*s, char \*t)

{

int n;

char \*p, \*r;

n = 0;

while (\*s)

{

p = s;

r = t;

while (\*r)

if (\*r == \*p)

{

r++;

p++

}

else break;

if (r == '\0')

n++;

s++;

}

return n;

}

main()

{

char s[100], t[100];

int m;

printf("\nPlease enter string s: ");

scanf("%s", s);

printf("\nPlease enter substring t:");

scanf("%s", t);

m = fun(s, t);

printf("\nThe result is: m=%d\n", m);

}

代码：

#include <stdio.h>

**int** fun(**char** \*s, **char** \*t)

{

**int** n;

**char** \*p, \*r;

    n = 0;

**while** (\*s)

    {

        p = s;

        r = t;

**while** (\*r)

**if** (\*r == \*p)

            {

                r++;

                p++;

            }

**else**  **break**;

**if** (\*r == '\0')

            n++;

        s++;

    }

**return** n;

}

main()

{

**char** s[100], t[100];

**int** m;

**printf**("\nPlease enter string s: ");

**scanf**("%s", s);

**printf**("\nPlease enter substring t:");

**scanf**("%s", t);

    m = fun(s, t);

**printf**("\nThe result is: m=%d\

请编写一个查找子字符串的程序，并统计子字符串出现的次数。

\*\*输入格式要求："%s" 提示信息："请输入主串:" "请输入要查找的串:"

\*\*输出格式要求："%s,%s:" "子串出现的次数：%d\n" "子串不在主串中\n"

程序运行示例1如下：

请输入主串:Hello,world!

请输入要查找的串:l

Hello,world!,l:子串出现的次数：3

程序运行示例2如下：

请输入主串:Hello,world!

请输入要查找的串:abc

Hello,world!,abc:子串不在主串中

代码：

#include <stdio.h>

#define Max 100

**int** count(**char** mstr[], **char** sstr[])

{

**int** i, j, k, n = 0;

**for** (i = 0; mstr[i] != '\0'; i++)

    {

        j = i;

        k = 0;

**while** (sstr[k] == mstr[j])

        {

            j++;

            k++;

        }

**if** (sstr[k] == '\0')

            n++;

    }

**return** n;

}

**int** main()

{

**int** n;

**char** mstr[Max], sstr[Max];

**printf**("请输入主串:");

**scanf**("%s", mstr);

**printf**("请输入要查找的串:");

**scanf**("%s", sstr);

**printf**("%s,%s:", mstr, sstr);

    n = count(mstr, sstr);

**if** (n > 0)

    {

**printf**("子串出现的次数：%d\n", n);

    }

**else**

**printf**("子串不在主串中\n");

}

以下程序实现从键盘输入一个字符串，将其字符顺序颠倒后重新存放，并输出这个字符串。（用字符数组实现）。找出其中错误并改正之。

|  |
| --- |
| #include <stdio.h>  #include <string.h>  **void** Inverse(**char** rstr[])    main()  {  **char**  str[80];    **printf**("Input a string:\n");  **gets**(str);      Inverse(str);  **printf**("The inversed string is:\n");  **puts**(str);  }    **void** Inverse(**char** rstr[])  {  **int** i,n;  **char** temp;    **for**(i=0, n=(**strlen**(rstr)); i<n; i++, n--)      {          temp = rstr[i];          rstr[i] = rstr[n];          rstr[n] = temp;      }  } |

代码：

#include <stdio.h>

#include <string.h>

**void** Inverse(**char** rstr[]);

main()

{

**char**  str[80];

**printf**("Input a string:\n");

**gets**(str);

    Inverse(str);

**printf**("The inversed string is:\n");

**puts**(str);

}

**void** Inverse(**char** rstr[])

{

**int** i,n;

**char** temp;

**for**(i=0, n=(**strlen**(rstr)-1); i<n; i++,n--)

    {

        temp = rstr[i];

        rstr[i] = rstr[n];

        rstr[n] = temp;

    }

}

请按给定的函数原型，编写程序判断输入的字符串是否为“回文”。所谓“回文”，是指顺读和倒读都一样的字符串。如：“level”、“ABCCBA”都是回文。如是回文输出这个字符串，否则输出“No!”。要求：在主函数读入字符串，判断是否是回文调用Judger()函数实现。若是回文返回值为1，否则返回值为0。Judger函数函数原型：int Judger( char a[ ]);

\*\*输入格式要求：无

\*\*输出格式要求："No!" （不是回文） 说明：是回文则输出串本身

程序运行的输入输出样例1：

输入：

abc

输出：

No!

程序运行的输入输出样例2：

输入：

12321

输出：

12321

代码：

#include <stdio.h>

#include <string.h>

**int**  Judger( **char**  a[ ]);

**int** main()

{

**char** a[80];

**int** flag;

**gets**(a);

    flag = Judger(a);

**if** (flag == 1)

    {

**puts**(a);

    }

**else**

    {

**printf**("No!");

    }

**return** 0;

}

**int**  Judger( **char**  a[ ])

{

**char** \*pStar, \*pEnd;

**int** i, j, flag = 1;

    j = **strlen**(a);

    pStar = &a[0];

    pEnd = &a[j - 1];

**for** (i = 0; i < (j / 2); i++)

    {

**if** (\*pStar != \*pEnd)

        {

            flag = 0;

        }

        pStar++;

        pEnd--;

    }

**return** flag;

}

输入英文的星期几，查找星期表后输出对应的数字。具体的算法是：若发现与表中某项相同，则输出该字符串在表中的位置（序号）；若查到表尾仍不同，则输出错误信息。设星期表的内容如下图所示。

0 sunday

1 monday

2 tuesday

3 wednesday

4 thursday

5 friday

6 saturday

\*\*输入格式要求："%s" 提示信息："Enter a string:"

\*\*输出格式要求："%s is %d\n" "Error\n"

程序运行示例如下：

Enter a string:wednesday

wednesday is 3

代码：

#include <stdio.h>

**char** week\_day[][10]={"sunday","monday","tuesday","wednesday","thursday","friday","saturday"};

main()

{

**int** i,j;

**char** id[10],c;

**printf**("Enter a string:");

**scanf**("%s",id);

**for**(i=0;i<7;i++)

    {

**for**(j=0;(c=week\_day[i][j])==id[j]&&c!='\0';j++)

            ;

**if**(c=='\0')

**break**;

    }

**if**(i<7)

**printf**("%s is %d\n",id,i);

**else**

**printf**("Error\n");

}

有一篇文章，共有三行文字，每行有80个字符。要求分别统计出其中英文大写字母、小写字母、数字、空格、以及其它字符的个数。

程序的运行示例如下：

请输入第0行：

helloworld

请输入第1行：

12345gogogo

请输入第2行：

end bye bye.

helloworld

12345gogogo

end bye bye.

大写字母数：0

小写字母数：25

数字个数 ：5

空格个数 ：2

其它字符 ：1

\*\*\*输入提示信息\*\*\*："\n请输入第%d行：\n"

\*\*\*输入数据格式\*\*\*：使用gets()

\*\*\*输出数据格式\*\*\*："%s\n"

"大写字母数：%d\n"

"小写字母数：%d\n"

"数字个数 ：%d\n"

"空格个数 ：%d\n"

"其它字符 ：%d\n"

代码：

#include <stdio.h>

**int** main()

{

**int** i, j, uppn, lown, dign, span, othn;

**char** text[3][80];

    uppn = lown = dign = span = othn = 0;

**for** (i = 0; i < 3; i++)

    {

**printf**("\n请输入第%d行：\n", i);

**gets**(text[i]);

**for** (j = 0; j < 80 && text[i][j] != '\0'; j++)

        {

**if** (text[i][j] >= 'A' && text[i][j] <= 'Z')

            {

                uppn += 1;

            }

**else** **if** (text[i][j] >= 'a' && text[i][j] <= 'z')

            {

                lown += 1;

            }

**else** **if** (text[i][j] >= '0' && text[i][j] <= '9')

            {

                dign += 1;

            }

**else** **if** (text[i][j] == ' ')

            {

                span += 1;

            }

**else**

            {

                othn += 1;

            }

        }

    }

**for** (i = 0; i < 3; i++)

    {

**printf**("%s\n", text[i]);

    }

**printf**("大写字母数：%d\n", uppn);

**printf**("小写字母数：%d\n", lown);

**printf**("数字个数  ：%d\n", dign);

**printf**("空格个数  ：%d\n", span);

**printf**("其它字符  ：%d\n", othn);

**return** 0;

}

找出按字典顺序排在最前面的国名

输入5个国名，编程找出并输出按字典顺序排在最前面的国名。

提示：所谓字典顺序就是将字符串按由小到大的顺序排列，因此找出按字典顺序排在最前面的国名指的就是最小的字符串。

程序的运行结果示例：

Input five countries' names:

America↙

China↙

Japan↙

England↙

Sweden↙

The minimum is:America

输入格式: 国名输入用gets()函数

输出格式：

输入提示信息："Input five countries' names:\n"

输出提示信息："The minimum is:%s\n"

代码：

#include  <stdio.h>

#include  <string.h>

#define N 80                           /\* 字符串最大长度 \*/

**int** main()

{

**int**    n;

**char**  str[N], min[N];

**printf**("Input five countries' names:\n");

**gets**(str);                          /\* 输入一个字符串 \*/

**strcpy**(min, str);                  /\* 将其作为最小字符串暂存 \*/

**for** (n = 1; n < 5; n++)

    {

**gets**(str);                      /\* 每次输入一个字符串 \*/

**if** (**strcmp**(str, min) < 0)    /\* 比较两个字符串的大小 \*/

        {

**strcpy**(min, str);

        }

    }

**printf**("The minimum is:%s\n", min);/\* 输出最小字符串min \*/

**return** 0;

}

/\* 函数功能：找出并返回按字典顺序排在最前面的字符串 \*/

**int** MinString(**char** str[][N], **int** n)

{

**int** i, minIndex;

**char** min[N];

**strcpy**(min, str[0]);

    minIndex = 0;

**for** (i = 1; i < n; i++)

    {

**if** (**strcmp**(str[i], min) < 0)

        {

**strcpy**(min, str[i]);

            minIndex = i;

        }

    }

**return** minIndex;         /\* 返回最小的字符串在二维字符数组中的位置 \*/

}

请进行程序设计，以最简形式输出两个分数的加减运算结果。

输入包含多组测试数据，每组数据占一行，每行数据是一个字符串，格式为："a/boc/d"。

其中a, b, c, d是一个0-9的整数。o是运算符"+"或者"-"。

输入数据保证合法并且以EOF结束 (CTRL+Z键)

对于输入数据的每一行，输出两个分数的运算结果。

注意结果应符合书写习惯，没有多余的符号、分子、分母，并且化简至最简分数

样例输入：

1/8+3/8

1/4-1/2

1/3-1/3

样例输出：

1/2

-1/4

0

代码：

#include<stdio.h>

**char** str[20];

**int** Gcd(**int** m,**int** n)

{

**if** (m==0) **return** n;

**return** Gcd(n%m,m);

}

**int** main()

{

**int** fz,fm,gcd;

**while**(**scanf**("%s",str)!=EOF)

    {

**if**(str[3]=='-')

            fz=(str[0]-'0')\*(str[6]-'0')-(str[2]-'0')\*(str[4]-'0');

**else** fz=(str[0]-'0')\*(str[6]-'0')+(str[2]-'0')\*(str[4]-'0');

**if**(fz)

        {

            fm=(str[2]-'0')\*(str[6]-'0');

            gcd=Gcd(fz,fm);

**if**(gcd<0) gcd=-gcd;

**if**(fm/gcd==1) **printf**("%d\n",fz/gcd);

**else** **printf**("%d/%d\n",fz/gcd,fm/gcd);

        }

**else** **puts**("0");

    }

}

在一个有序(按非递减顺序)的链表中插入一个元素为x的结点，使插入后的链表仍然有序（链表数据域为整型数，N为6）。

\*\*输入提示："输入数组6个元素的值。\n"

\*\*输入格式："%d"

\*\*输出提示："此链表各个结点的数据域为："

\*\*输出格式："%d "

\*\*输入提示："输入要插入的数据x:"

\*\*输入格式："%d"

\*\*输出提示："插入后链表各个结点的数据域为："

\*\*输出格式："%d "

程序运行示例如下：

输入数组6个元素的值。

12 23 34 45 56 67

此链表各个结点的数据域为：12 23 34 45 56 67

输入要插入的数据x:36

插入后链表各个结点的数据域为：12 23 34 36 45 56 67

代码：

#include <stdio.h>

#include <stdlib.h>

#define N 6

**struct** LNode

{

**int** data;

**struct** LNode \*next;

}                   ;

**struct** LNode\* create\_rear(**int** a[], **int** n);

**void** output(**struct** LNode \*h) ;

**struct** LNode\* insert\_sort(**struct** LNode\* h, **int** x);

**int** main(**int** argc, **char** \*argv[])

{

**int** a[N], i, x;

**struct** LNode\* head;

**printf**("输入数组%d个元素的值。\n", N);

**for** (i = 0; i < N; i++)

**scanf**("%d", &a[i]);

    /\*①创建链表head,其结点的值依次为数组a元素的值\*/

    head = create\_rear(a, N);

    /\*②输出链表head\*/

**printf**("此链表各个结点的数据域为：");

    output(head);

**printf**("输入要插入的数据x:");

**scanf**("%d", &x);

    /\*③调用函数insert\_sort插入一个元素\*/

    head = insert\_sort(head, x);

**printf**("插入后链表各个结点的数据域为：");

    /\*④输出插入后的链表head\*/

    output(head);

**return** 0;

}

**struct** LNode\* create\_rear(**int** a[], **int** n)

{

    /\*新建一个链表h，每个结点依次插入到链尾，将链表的头指针返回 \*/

**struct** LNode \*h = NULL;

**struct** LNode \*s, \*r; /\*用s指向要插入结点，r指向链表的尾结点\*/

**int** i;

**for** (i = 0; i < n; i++)

    {

        s = (**struct** LNode \*)**malloc**(**sizeof**(**struct** LNode));

        s->data = a[i];

        s->next = NULL;

**if** (h == NULL)

            h = s;  /\*如果链表为空，则头指针h指向s\*/

**else**

            r->next = s;    /\*否则将s链接到尾结点r之后\*/

        r = s;            /\*将r指向尾结点\*/

    }

**return** h;  /\*返回链表的头指针\*/

}

**void** output(**struct** LNode \*h)

{

    /\*将链表h的各个结点的数据域依次输出，即遍历该链表\*/

**struct** LNode \*p = h;/\*从第一个结点开始，用p依次指向各个结点\*/

**while** (p)

    {

        /\*只要p是一个非空结点，则输出其数据域，然后将p后移\*/

**printf**("%d ", p->data);

        p = p->next;  //将p后移

    }

**printf**("\n");

}

**struct** LNode\* insert\_sort(**struct** LNode\* h, **int** x)

{

    /\*在已经有序有链表h中，插入值为x的结点，插入后仍有序，并返回头指针。\*/

**struct** LNode \*s, \*pre, \*p;/\*用s表示插入结点，pre为p的前驱\*/

    s = (**struct** LNode \*)**malloc**(**sizeof**(**struct** LNode));

    s->data = x;

**if** (h == NULL)

    {

        /\*h为空链表\*/

        s->next = NULL;

        h = s;

    }

**if** (x <= h->data)

    {

        /\*x不大于链表中第一个结点的数据域，将s插入链首\*/

        s->next = h;

        h = s;

    }

**else**

    {

        p = h;

**while** (p && x > p->data)

        {

            /\*如果p不空，且x不大于p所指结点的数据域,p后移，pre为p的前驱\*/

            pre = p;

            p = p->next;

        }

        /\*将s插入到pre所指结点之后。\*/

        s->next = pre->next;

        pre->next = s;

    }

**return** h;

}

----------------------------------------------------------------------------------------------------------------------------

建立一个链表，使链表中从头到尾的结点数据域依次是一个数组的各个元素的值。程序先建立链表然后再遍历输出（假定链表和数组均有6个整型元素）。

\*\*输入提示："输入数组%d个元素的值。\n"

\*\*输入格式："%d"

\*\*输出格式："%d "

程序运行示例如下：

输入数组6个元素的值。↙

1 3 5 7 9 11↙

此链表各个结点的数据域为：1 3 5 7 9 11

代码：

#include <stdio.h>

#include <stdlib.h>

#define N 6

**struct** LNode

{

**int** data;

**struct** LNode \*next;

}                        ;

**struct** LNode\* create\_rear(**int** a[], **int** n);

**void** output(**struct** LNode \*h);

**int** main(**int** argc, **char** \*argv[])

{

    /\*①定义一个数组、头指针\*/

**int** a[N], i;

**struct** LNode\* head;

    /\*②输入数组元素的值\*/

**printf**("输入数组%d个元素的值。\n", N);

**for** (i = 0; i < N; i++)

**scanf**("%d", &a[i]);

    /\*③创建链表head,其结点的值依次为数组a元素的值\*/

    head = create\_rear(a, N);

    /\*④输出链表head\*/

**printf**("此链表各个结点的数据域为：");

    output(head);

**return** 0;

}

**struct** LNode\* create\_rear(**int** a[], **int** n)

{

    /\*新建一个链表h，每个结点依次插入到链尾，将链表的头指针返回 \*/

**struct** LNode \*h = NULL;

**struct** LNode \*s, \*r; /\*用s指向要插入结点，r指向链表的尾结点\*/

**int** i;

**for** (i = 0; i < n; i++)

    {

        s = (**struct** LNode \*)**malloc**(**sizeof**(**struct** LNode));

        s->data = a[i];

        s->next = NULL;

**if** (h == NULL)

            h = s;       /\*如果链表为空，则头指针h指向s \*/

**else**

            r->next = s; /\*否则将s链接到尾结点r之后     \*/

        r = s;           /\*将r指向尾结点               \*/

    }

**return** h;  /\*返回链表的头指针\*/

}

**void** output(**struct** LNode \*h)

{

    /\*将链表h的各个结点的数据域依次输出，即遍历该链表\*/

**struct** LNode \*p = h;/\*从第一个结点开始，用p依次指向各个结点\*/

**while** (p)

    {

        /\*只要p是一个非空结点，则输出其数据域，然后将p后移\*/

**printf**("%d ", p->data);

        p = p->next;  //将p后移

    }

}

-

1.对输入的字符串进行长度验证，保证输入的字符串的长度在指定的范围内，如果不在指定的范围内，则一直提示用户输入，直到输入合法长度的字符串为止。

程序的示例运行如下：

请输入一个字符串(长度为[3..5]个字符)：a

请输入一个字符串(长度为[3..5]个字符)：ab

请输入一个字符串(长度为[3..5]个字符)：abcdef

请输入一个字符串(长度为[3..5]个字符)：abc

你输入的字符串为：abc

#include <stdio.h>

void getString(char \*str, int min, int max)

{

int len;

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

while (len < min || len > max)

{

printf("请输入一个字符串(长度为[%d..%d]个字符)：\n", min, max);

scanf("%s", str);

len = strlen(str);

}

}

int main(int argc, char \*argv[])

{

char string[100];

getString(string, 3, 5);

printf("你输入的字符串为：%s\n", string);

return 0;

}

2.利用一个字符数组作函数参数,实现字符串（最大长度为80个字符 ）的逆序存放。

要求如下：

(1)在子函数Inverse中实现字符串的逆序存放。函数原型为：

void Inverse(char str[]);

(2)在主函数中

从键盘输入字符串(使用gets函数)

然后，调用Inverse函数，

最后，输出逆序后的字符串。

(3)\*\*输入提示信息："Input a string:\n"

\*\*输出提示信息："Inversed results:\n"

\*\*输出格式："%s\n"

#include <stdio.h>

#include <string.h>

#define N 80

void Inverse(char str[]);

main()

{

char a[N];

printf("Input a string:\n");

gets(a);//1

Inverse(a);//2

printf("Inversed results:\n");

puts(a);//1

}

void Inverse(char str[])

{

int len, i, j;

char temp;//1

len = strlen(str);//1

for (i=0, j=len-1; i<j; i++, j--)//4

{

temp = str[i];//1

str[i] = str[j];//1

str[j] = temp;//1

}

}

3.输入一个字符串保存到数组str中，将str中的数字字符存储到数组tOrigin中，并将其转换为整数输出，用k记录字符串中数字字符的个数。例如：用户输入字符串1243abc3,则将12433取出以整数形式输出。阅读程序，找出其中的错误，并改正之。

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int k)

{

int i;

k = 0;

for (i = 0; s[i]; i++)

if (0 <= s[i] && s[I] <= 9)

{

t[\*k] = s[i];

\*k += 1;

}

}

void main()

{

char str[100], tOrigin[9];

long int n;

int i, k;

printf("Please enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

printf("The result is: %d", n);

}

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int \*k)// 声明为指针类型

{

int i;

\*k = 0;

for (i = 0; s[i]; i++)

if ('0' <= s[i] && s[i] <= '9')

{

t[\*k] = s[i];

\*k += 1;

}

t[\*k] = '\0';//添加字符串结束标志

}

int main()

{

char str[100], tOrigin[9];

double n = 0;//改成double

int i, k;

printf("Please enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

{

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

}

printf("The result is: %d", (int)n);//输出时再强转为int

return 0;

}

4.下面程序的Squeeze函数的功能是删除字符串s中所出现的与变量c相同的字符。

注意：请将修改正确后的完整源程序拷贝粘贴到答题区内。

对于没有错误的语句，请不要修改，修改原本正确的语句也要扣分。

当且仅当错误全部改正，且程序运行结果调试正确，才给加5分。

改错时不能改变程序原有的意图，不能改变函数原型。

\*\*输入输出格式要求：不要增加输入提示信息；

按原题要求，先输入字符串s，回车换行后再输入字符c

（c为任意字符）。不能修改函数原型。

#include <stdio.h>

void Squeeze(char \*s, char c);

main()

{

char a[80],c, \*s;

scanf("%s",p);

scanf("%c",c);

Squeeze(char \*s, char c);

printf("%s\n",s);

}

void Squeeze(char \*s, char c);

{

int i,j;

for (i=0; \*s+i !=' \0'; i++);

{

if (\*s+i != 'c')

{

\*s+j = \*s+i;

j++;

}

\*s+i = '\0';

}

}

#include <stdio.h>

void Squeeze(char \*s, char c);

main()

{

char a[80], s;

scanf("%s", a);

scanf(" %c", &s);

Squeeze(a, s);

printf("%s\n", a);

}

void Squeeze(char \*s, char c)

{

int i, j;

int flag = 0;

for (i = 0, j = 0; \* (s + i) != '\0'; i++)

{

if (\*(s + i) != c)

{

flag = 1;

\*(s + j) = \*(s + i);

j++;

}

}

if (flag)

\*(s + j) = '\0';

}

5.写一个函数，将一个字符串中的元音字母复制到另一个字符串，然后输出。

程序的运行示例如下：

输入字符串：Hello,world!

字符串中的元音字母是eoo

\*\*\*输入数据格式\*\*\*："\n输入字符串："，使用gets()

\*\*\*输出数据格式\*\*\*："\n字符串中的元音字母是%s"

#include <stdio.h>

int main()

{

char str[80], c[80];

void cpy();

printf("\n输入字符串：");

gets(str);

printf("%s", str);

cpy(str, c);

printf("\n字符串中的元音字母是%s", c);

return 0;

}

void cpy(char s[], char c[])

{

int i, j;

for (i = 0, j = 0; s[i] != '\0'; i++)

{

if (s[i] == 'a' || s[i] == 'A' || s[i] == 'e' || s[i] == 'E' ||

s[i] == 'i' || s[i] == 'I' || s[i] == 'o' || s[i] == 'O' ||

s[i] == 'u' || s[i] == 'U')

{

c[j] = s[i];

j++;

}

}

c[j] = '\0';

}

6.请按给定的函数原型编程实现将字符数组中的字符串的第m个字符开始的n个字符逆序存放。要求在主函数读入字符串，且逆序存放后的字符串也在主函数打印。函数原型：

void inverse(char str[],int m,int n);

输入要求: 输入的一行字符串，应包含字母，数字以及空格字符

友情提示:

在执行输入字符串的函数之前,请用getchar();把输入缓冲区中的换行符读出!!!

\*\*\*\*输入提示信息和格式要求为：

"input m,n:"

"%d,%d"

"input the string:"

\*\*\*\*输出格式为:"the inverse string:%s"

#include <stdio.h>

#define N 100

void inverse(char str[],int m,int n);

main()

{

char str[N];

int m,n;

printf("input m,n:"); //1

scanf("%d,%d",&m,&n);

getchar();

printf("input the string:"); //1

gets(str);

inverse(str,m,n); //2

printf("the inverse string:%s",str); //1

}

void inverse(char str[],int m,int n)

{

int i;

char a[N];

for(i=0;i<n;i++)

{

a[i]=str[m-1+i]; //1

}

for(i=0;i<n;i++)

{

str[m+i-1]=a[n-1-i]; //1

}

}

7.编写一个程序，将用户输入的由数字字符和非数字字符组成的字符串中的数字提取出来（例如：输入asd123,34fgh\_566kkk789，则产生的数字分别是123、34、789）。

\*\*输入格式要求：提示信息："Please enter a string:"

\*\*输出格式要求："the result of output:\n" "%10d\n"

程序运行示例如下：

Please enter a string:

abc123def456ghi111bbbccc99go100

the result of output:

123

456

111

99

100

#include <stdio.h>

#include <stdlib.h>

#define N 100

int getline(char \*s, int len);

int main()

{

char line[N], b[20], \*ptr1, \*ptrb;

int n = 0, j, a[50];

printf("Please enter a string:");

getline(line, N);

ptr1 = line;

while (\*ptr1 != '\0')

{

ptrb = b;

for (j = 0; \*ptr1 >= '0' && \*ptr1 <= '9'; ptr1++, ptrb++, j++)

{

\*ptrb = \*ptr1;

}

if (j)

{

\*ptrb = '\0';

\*(a + n) = atoi(b);

n++;

}

if (\*ptr1 != '\0')ptr1++;

}

printf("the result of output:\n");

for (j = 0; j < n; j++)

{

printf("%10d\n", \*(a + j));

}

return 0;

}

int getline(char \*s, int len)

{

int c;

char \*p = s;

while (--len > 0 && (c = getchar()) != '\n')

{

\*s++ = c;

}

\*s = '\0';

return s - p;

}

8.输入一个字符串保存到数组str中，将str中的数字字符存储到数组tOrigin中，并将其转换为整数输出，用k记录字符串中数字字符的个数。例如：用户输入字符串1243abc3,则将12433取出以整数形式输出。阅读程序，找出其中的错误，并改正之。

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int k)

{

int i;

k = 0;

for (i = 0; s[i]; i++)

if (0 <= s[i] && s[I] <= 9)

{

t[\*k] = s[i];

\*k += 1;

}

}

void main()

{

char str[100], tOrigin[9];

long int n;

int i, k;

printf("Please enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

printf("The result is: %d", n);

}

#include <math.h>

#include <stdio.h>

void fun(char \*s, char \*t, int \*k)// 声明为指针类型

{

int i;

\*k = 0;

for (i = 0; s[i]; i++)

if ('0' <= s[i] && s[i] <= '9')

{

t[\*k] = s[i];

\*k += 1;

}

t[\*k] = '\0';//添加字符串结束标志

}

int main()

{

char str[100], tOrigin[9];

double n = 0;//改成double

int i, k;

printf("Please enter string s:");

scanf("%s", str);

fun(str, tOrigin, &k);

for (i = 0; i < k; i++)

{

n += (tOrigin [k - 1 - i] - '0') \* pow(10, i);

}

printf("The result is: %d", (int)n);//输出时再强转为int

return 0;

}

9.写一个函数，输入一行字符(字符串长度<=80)，单词之间可以以空格作为分隔符，将此字符串中最长的单词输出，字符串中出现的非英文字符不统计不输出。

程序运行示例1：

输入一行文本：

I am a student

最长的单词是：

student (后面没有回车)

程序运行示例2：

输入一行文本：

I am teacher11

最长的单词是：

teacher (后面没有回车)

程序运行示例3：

输入一行文本：

house 123@4#567\*9 clothes food

最长的单词是：

clothes (后面没有回车)

输入提示："输入一行文本：\n"

输入格式：gets()

输出提示："最长的单词是：\n"

输出格式：%c (用循环输出)

#include <stdio.h>

int alphabetic(char c) //2

{

if ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z')) //2

{

return 1; //1

}

else

{

return 0;//1

}

}

/\* 寻找最长单词 \*/

int longest(char string[]) //2

{

int len = 0, i, length = 0, flag = 1, place, point; //2

for (i = 0; i <= strlen(string); i++) //2

{

if (alphabetic(string[i])) //1

{

if (flag) //1

{

point = i;//1

flag = 0;

}

else //1

{

len++; //1

}

}

else //1

{

flag = 1;//1

if (len > length)//1

{

length = len;//1

place = point;//1

len = 0;

}

}

}

return place;//2

}

int main()

{

int i;

char line[100]; //2

printf("输入一行文本：\n");

gets(line); //1

printf("最长的单词是：\n");

for (i = longest(line); alphabetic(line[i]); i++) //2

{

printf("%c", line[i]); //1

}

return 0;

}

10.对输入的8个字符串(每个字符串长度不超过20)按照字典顺序进行排序并输出。不限定排序方法，不能使用goto语句。

输入输出格式：

输入：%s

输出：%s\n

输入：多个字符串用空格分隔

输出：排序好的字符串，每行一个字符串，末尾有空行。

输入输出样例：

输入：

diankuang liuxu sui fengwu qingbo taohua zhu shuiliu

输出：

diankuang

fengwu

liuxu

qingbo

shuiliu

sui

taohua

zhu

#include<stdio.h>

#include<string.h>

#define MAX 8

#define CMAX 10

int main()

{

char a[MAX][CMAX], term[CMAX]; //1

int i, j;

for (i = 0; i < 8; i++)

{

scanf("%s", a[i]); //1

}

for (i = 0; i < 8 - 1; i++) //1

{

for (j = 1; j < 8 - i; j++) //1

{

if (strcmp(a[j], a[j - 1]) < 0) //1

{

strcpy(term, a[j]); //1

strcpy(a[j], a[j - 1]); //1

strcpy(a[j - 1], term); //1

}

}

}

for (i = 0; i < 8; i++)

{

printf("%s\n", a[i]);

}

return 0;

}

11.输入字符串，包含：字母、数字、标点符号，以及空格字符，并将其逆序输出。

例如，当输入字符串为“I am a student.”,输出为“.tneduts a ma I”，假设字符数组最大长度为30。

输入提示信息："Please Enter String1:\n"

输入格式：gets()

输出格式："Result is:\n%s\n"

程序运行结果示例：

Please Enter String1:↙

I am a student.

Result is:↙

.tneduts a ma I↙

#include <stdio.h>

#include <string.h>

#define N 30

int main( )

{

char str[N], c;

int n, i, j;

printf("Please Enter String1:\n");

gets(str); //1

n = strlen(str) - 1; //1

for (i = 0; i <= n / 2; i++) //2

{

c = str[i]; //1

str[i] = str[n - i]; //1

str[n - i] = c; //1

}

printf("Result is:\n%s\n", str);

return 0;

}

堆栈（Strack）是指这样一段内存，它可以理解为一个筒结构，先放进筒中的数据被后放进筒中的数据“压住”，只有后放进筒中的数据都取出后，先放进去的数据才能被取出，称为“后进先出”。堆栈的长度可随意增加。堆栈结构可用链表实现。设计一个链表结构需包含两个成员：一个存放数据，一个为指向下一个节点的指针。当每次有一个新数据要放入堆栈时，称为“压入堆栈”，这时动态建立一个链表的节点，并连接到链表的结尾；当每次从堆栈中取出一个数据时，称为“弹出堆栈”，这意味着从链表的最后一个节点中取出该节点的数据成员，同时删除该节点，释放该节点所占的内存。 编程用链表方法实现堆栈数据结构。程序至少应该包含三个函数： 1、主函数 2、压栈操作：将5个数据依次压入堆栈 3、出栈操作：将5个数据依次弹出堆栈（注意：因为后进先出的原因，弹出时是逆序，另外，弹出后要删除该节点内存）; \*\*输入提示：printf("请输入要压栈的整数："); \*\*输入格式："%d" \*\*压栈后输出格示："Push %dth Data:%d\n" \*\*出栈后输出格式："Pop %dth Data:%d\n" 程序运行示例： 请输入要压栈的整数：1↙ Push 1th Data:1↙ 请输入要压栈的整数：2↙ Push 2th Data:2↙ 请输入要压栈的整数：3↙ Push 3th Data:3↙ 请输入要压栈的整数：4↙ Push 4th Data:4↙ 请输入要压栈的整数：5↙ Push 5th Data:5↙ Pop 5th Data:5↙ Pop 4th Data:4↙ Pop 3th Data:3↙ Pop 2th Data:2↙ Pop 1th Data:1↙#include <stdio.h>

#include <stdlib.h>

typedef struct stack

{

    int data;

    struct stack \*next;

}                     STACK;

STACK \*head, \*pr;

int nodeNum = 0;                            /\* 堆栈节点数寄存器 \*/

STACK \*CreateNode(int num);

STACK \*PushStack(int num);

int PopStack(void);

int main()

{

    int pushNum[5] , popNum[5], i;

    for (i = 0; i < 5; i++)

    {

        printf("请输入要压栈的整数：");

        scanf("%d",&pushNum[i]);

        PushStack(pushNum[i]);

        printf("Push %dth Data:%d\n", i + 1, pushNum[i]);

    }

    for (i = 0; i < 5; i++)

    {

        popNum[i] = PopStack();

        printf("Pop %dth Data:%d\n", 5 - i, popNum[i]);

    }

    return 0;

}

/\* 函数功能：生成一个新的节点，并为该节点赋初值，返回指向新的节点的指针 \*/

STACK \*CreateNode(int num)

{

    STACK \*p;

    p = (STACK \*)malloc(sizeof(STACK));

    if (p == NULL)

    {

        printf("No enough memory!\n");

        exit(0);

    }

    p->next = NULL;              /\* 为新建的节点指针域赋空指针 \*/

    p->data = num;                   /\* 为新建的节点数据区赋值 \*/

    return p;

}

/\*  函数功能：将整型变量num的值压入堆栈，返回指向链表新节点的指针 \*/

STACK \*PushStack(int num)

{

    if (nodeNum == 0) /\* 若为首节点，则保留该节点地址在head中\*/

    {

        head = CreateNode(num);

        pr = head;

        nodeNum++;  /\* 堆栈节点数寄存器+1 \*/

    }

    else             /\* 若不是首节点，则将新建节点连到链表的结尾处 \*/

    {

        pr->next = CreateNode(num);

        pr = pr->next;

        nodeNum++;  /\* 堆栈节点数寄存器+1 \*/

    }

    return pr;

}

/\*  函数功能：将当前栈顶的数据弹出堆栈，返回从堆栈中弹出的数据 \*/

int PopStack(void)

{

    STACK \*p = head;

    int result;

    for (;;)

    {

        if (p->next == NULL)     /\* 查找最后一个节点 \*/

        {

            break;

        }

        else

        {

            pr = p;             /\* 记录最后一个节点的前一个节点的地址 \*/

            p = p->next;

            nodeNum--;          /\* 堆栈节点数寄存器-1 \*/

        }

    }

    pr->next = NULL;          /\* 将末节点的前一个节点置成末节点 \*/

    result = p->data;

    free(p);

    return result;

}

将一个链表中元素值为x的结点删除。（链表数据域为整数，初始长为6个元素） \*\*输入提示信息："输入数组%d个元素的值。\n" \*\*输入格式："%d" \*\*输出提示："此链表各个结点的数据域为：" \*\*输出格式："%d " \*\*输入提示信息："输入要删除的数据x: " \*\*输入格式："%d" \*\*输出提示： "删除后链表各个结点的数据域为：" \*\*输出格式："%d " 程序运行示例1： 输入数组6个元素的值。 11 22 33 44 55 66 此链表各个结点的数据域为：11 22 33 44 55 66 输入要删除的数据x: 33 删除后链表各个结点的数据域为：11 22 44 55 66 程序运行示例2： 输入数组6个元素的值。↙ 6 2 7 4 5 9 此链表各个结点的数据域为：6 2 7 4 5 9 ↙ 输入要删除的数据x: 8 删除后链表各个结点的数据域为：6 2 7 4 5 9 ↙

#include <stdio.h>

#include <stdlib.h>

#define N 6

struct LNode

{

    int data;

    struct LNode \*next;

}                 ;

struct LNode\* create\_rear(int a[], int n);

void output(struct LNode \*h) ;

struct LNode\* delete\_node(struct LNode\* h, int x);

int main(int argc, char \*argv[])

{

    int a[N], i, x;

    struct LNode\* head;

    printf("输入数组%d个元素的值。\n", N);

    for (i = 0; i < N; i++)

        scanf("%d", &a[i]);

    /\*创建链表head,其结点的值依次为数组a元素的值\*/

    head = create\_rear(a, N);

    /\*删除前输出链表head\*/

    printf("此链表各个结点的数据域为：");

    output(head);

    printf("输入要删除的数据x: ");

    scanf("%d", &x);

    head = delete\_node(head, x);   /\*调用删除函数\*/

    printf("删除后链表各个结点的数据域为：");

    output(head);  /\*删除后输出链表head\*/

    return 0;

}

struct LNode\* create\_rear(int a[], int n)

{

    /\*新建一个链表h，每个结点依次插入到链尾，将链表的头指针返回 \*/

    struct LNode \*h = NULL;

    struct LNode \*s, \*r; /\*用s指向要插入结点，r指向链表的尾结点\*/

    int i;

    for (i = 0; i < n; i++)

    {

        s = (struct LNode \*) malloc(sizeof(struct LNode));

        s->data = a[i];

        s->next = NULL;

        if (h == NULL)

            h = s;        /\*如果链表为空，则头指针h指向s \*/

        else

            r->next = s;  /\*否则将s链接到尾结点r之后     \*/

        r = s;                /\*将r指向尾结点                \*/

    }

    return h;  /\*返回链表的头指针\*/

}

void output(struct LNode \*h)

{

    /\*将链表h的各个结点的数据域依次输出，即遍历该链表\*/

    struct LNode \*p = h;/\*从第一个结点开始，用p依次指向各个结点\*/

    while (p)

    {

        /\*只要p是一个非空结点，则输出其数据域，然后将p后移\*/

        printf("%d ", p->data);

        p = p->next;  //将p后移

    }

    printf("\n");

}

struct LNode\* delete\_node(struct LNode\* h, int x)

{

    /\*将链表h中值为x的结点第一个结点删除，并返回头指针。\*/

    struct LNode \*pre, \*p;/\*pre所指结点为p所指结点的前驱\*/

    p = h;

    while (p && x != p->data)

    {

        /\*如果p不空，且x不等于p所指结点的数据域,p后移，pre为p的前驱\*/

        pre = p;

        p = p->next;

    }

    if (p)

    {

        /\*在链表中找到了要删除的结点p，即p->data为x\*/

        if (p == h)

        {

            /\*p为链首结点，由于p没有前驱，删除后p的后继结点成为链首，需修改头指针\*/

            h = p->next;

        }

        else

        {

            /\*删除的p非链首结点，则p有前驱pre，删除时需将pre后面链接到p的后继结点\*/

            pre->next = p->next;

        }

    }

    return h;

}

创建动态链表并打印链表内容。要求通过键盘读入学生的学号和分数，当输入的学号为0时表示链表创建结束，并将读入的学生学号和成绩打印输出。下面程序中存在比较隐蔽的错误，请通过分析和调试程序，发现并改正程序中的错误。改错时不能改变程序原有的意图。

注意：将修改后的完整的源程序写在答题区内。

#include <stdio.h>

#define SIZE sizeof(struct node)

struct node

{

long num;

float score;

struct node \*next;

}

int main()

{

int n = 0;

struct node \*head = NULL, \*p1, \*p2;

p1 = p2 = (struct node\*)malloc(SIZE);

printf("Input %d node data:\n", n + 1);

scanf("%d%f", &p1->num, &p1->score);

while (p1->num != 0)

{

if (++n == 1)

head = p1;

else

p2->next = p1;

p1 = p2;

p1 = (struct node\*)malloc(SIZE);

printf("Input %d node data:\n", n + 1);

scanf("%d%f", &p1->num, &p1->score);

}

free(p1);

printf("Prit list:\n");

p2->next = NULL;

if (head != NULL)

{

p2=head;

while (p1 != NULL)

{

printf("num:%d\tscore:%.2f\n", p1->num, p1->score);

p1 = p1->next;

}

}

return 0;

}

#include <stdio.h>

#include <stdlib.h>

#define SIZE sizeof(struct node)

**struct** node

{

**long** num;

**float** score;

**struct** node \*next;

}              ;

**int** main()

{

**int** n = 0;

**struct** node \*head = NULL, \*p1, \*p2;

    p1 = p2 = (**struct** node\*)**malloc**(SIZE);

**printf**("Input %d node data:\n", n + 1);

**scanf**("%d%f", &p1->num, &p1->score);

**while** (p1->num != 0)

    {

**if** (++n == 1)

            head = p1;

**else**

            p2->next = p1;

        p2 = p1;

        p1 = (**struct** node\*)**malloc**(SIZE);

**printf**("Input %d node data:\n", n + 1);

**scanf**("%d%f", &p1->num, &p1->score);

    }

**free**(p1);

**printf**("Prit list:\n");

    p2->next = NULL;

**if** (head != NULL)

    {

        p1 = head;

**while** (p1 != NULL)

        {

**printf**("num:%d\tscore:%.2f\n", p1->num, p1->score);

            p1 = p1->next;

        }

    }

**return** 0;

}

将一个链表按逆序排列，即将链头当链尾，链尾当链头。

\*\*输入提示："\n请输入链表（非数表示结束）\n"

\*\*输入提示："结点值："

\*\*输出提示："\n原来表：\n"

\*\*输出格式："%4d"

\*\*输出提示："\n\n反转表：\n"

\*\*输出格式："%4d"

程序的运行示例如下：

请输入链表（非数表示结束）

结点值：3

结点值：4

结点值：5

结点值：6

结点值：7

结点值：end

原来表：

3 4 5 6 7

反转表：

7 6 5 4 3

#include <stdio.h>

struct line

{

int num;

struct line \*next;

} \*p1, \*p2;

struct line \*creat()

{

int temp;

struct line \*head = NULL;

printf("\n请输入链表（非数表示结束）\n结点值：");

while (scanf("%d", &temp))

{

p1 = (struct line \*)malloc(sizeof(struct line));

(head == NULL) ? (head = p1) : (p2->next = p1);

p1->num = temp;

printf("结点值：");

p2 = p1;

}

p2->next = NULL;

return head;

}

output(struct line \*outhead)

{

for (p1 = outhead; p1 != NULL; printf("%4d", p1->num), p1 = p1->next);

{

}

}

struct line \*turnback(struct line \*head)

{

struct line \*new, \*newhead = NULL;

do

{

p2 = NULL;

p1 = head;

while (p1->next != NULL)

{

p2 = p1;

p1 = p1->next;

}

if (newhead == NULL)

{

newhead = p1;

new = newhead->next = p2;

}

new = new->next = p2;

p2->next = NULL;

}

while (head->next != NULL);

return newhead;

}

int main(void)

{

struct line \*head;

head = creat();

printf("\n原来表：\n");

output(head);

head = turnback(head);

printf("\n\n反转表：\n");

output(head);

return 0;

}