**1、判断一个用户输入的字符串中有多少个元音字母即（a e i o u）的个数各是多少个？**

**要求：**

**1、要输入你要测试字符串的个数！**

**2、打印要求为：**

**a:num1;**

**e:num2;**

**i:num3;**

**o:num4;**

**u:num5;**

**列如：输入5个字符的字符串 aeiou**

**打印输出：**

**a:1;**

**e:1;**

**i:1;**

**o:1;**

**u:1;**

#include<stdio.h>

#include<string.h>

int main()

{

char str[80];

int a,e,i,o,u;

a=e=i=o=u=0;

gets(str);

for(i=0;i<strlen(str);i++)

switch(str[i]){

case'a':{a++;break;}

case'e':{e++;break;}

case'i':{i++;break;}

case'o':{o++;break;}

case'u':{u++;break;}

}

printf("a:%d\n",a);

printf("e:%d\n",e);

printf("i:%d\n",i);

printf("o:%d\n",o);

printf("u:%d\n",u);

return 0;

}

**2、用户输入一句话，要求把每个单词的首字母如果是小写的话变为大写字母！**

**要求：**

**1、输入一句话：what's you name?**

**2、打印要求为：What's You Name?**

#include<stdio.h>

#define N 100

void main()

{

char a[N]={"what are you talking about?"};

int i;

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

if(i>0&&a[i-1]==' ')a[i]-=32;

else if(i==0)a[i]-=32;

printf("%s",a);

}

**3、键盘输入两个学生姓名好年龄，输出年龄较大的学生年龄。**

#include "stdio.h"

struct student

{

char name[10];

int age;

}student[2];

void compare(struct student m,struct student n)

{

if(n.age>m.age)

printf("%s\n",n.name);

else

printf("%s\n",m.name);

}

void main(

{

printf("input:\n");

scanf("%s%d%s%d",&student[0].name,&student[0].age,&student[1].name,&student[1].age);

compare(student[0],student[1]);

}

**4、输入一个字符串，编写一个函数将字符串逆置，在主函数中输入和输出？**

#include <stdio.h>

#include <conio.h>

#include <string.h>

#define N 20

void fun(char \*s);

void main()

{

char s[N];

printf("请输入要逆置的字符串:");

gets(s);

fun(s);

printf("逆置后的字符串:");

puts(s);

getch();

}

void fun(char \*s)

{

int front,rear,s\_length;

char ch;

s\_length=strlen(s);

front=0;

rear=s\_length-1;

while(front<rear)

{

ch=s[front];

s[front]=s[rear];

s[rear]=ch;

front++;

rear--;

}

}

**5、判断两个字符相等？**

#include<stdio.h>

int main(){

char a,b;

scanf(" %c %c",&a,&b);//输入两个字符

if(a==b)puts("Yes");

else puts("No");

}

**6、按照学生成绩显示A，B，C，D等级？**

#include <stdio.h>

int main (void)

{

int x;//定义变量

scanf ("%d",&x);//获取输入的值

switch (x/10){

case 10:

case 9: printf ("A\n");break;

case 8: printf ("B\n");break;

case 7: printf ("C\n");break;

case 6: printf ("D\n");break;

default: printf ("E\n");break;

}

return 0;

}

**7、输入一串字符，分别统计数字、字母、和其他字符的个数？**

1 输入一个字符，对该字符进行判断：

a) 如果是数字，则数字累加器加一。

b) 如果是字母，则字母累加器加一。

c)如果是换行，则结束统计(以换行为结束符。如需其他结束符，根据需要更改判断)。

2输出结果。

#include <stdio.h>

int main()

{

int c, n, i;

c=n=0;

while(1)

{

i = getchar();

if(i>='0' && i <= '9') n++;

else if((i>='a' && i <= 'z')||(i>='A' && i <= 'Z'))

c++;

else if(c=='\n') break;

}

printf("数字%d个，字母%d个\n", n,c);

return 0;

}

**8、验证斐波那契数列？**

#include <stdio.h>

int fib(int n)

{

if(n==1||n==2)

return 1;

return fib(n-1)+fib(n-2);

}

int main()

{

int n;

scanf("%d",&n);

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

return 0;

}

**9、一个包含一串降序排列的数字的数组，要求输入任意数，按照原数组大小顺序，将其插入合适位置并输出？**

#include <stdio.h>

int main(){

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

if(x<a[i])

break;

for(j=5;j>i;j--)

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

a[j]=x;

printf("Now array a:\n");

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

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

printf("\n");

return 0;

**}**

**10、输入10个数，把最大的和最小的换个位置？**

#include <stdio.h>

void main()

{

int a[10],i,maxpos,minpos,temp;

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

{

printf("please input number %d:",i+1);

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

printf("\n");

}

maxpos=minpos=0;

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

{

if (a[i]>a[maxpos])

maxpos=i;

if (a[i]<a[minpos])

minpos=i;

}

temp=a[maxpos];

a[maxpos]=a[minpos];

a[minpos]=temp;

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

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

}

**11、在斐波那契数列中，求出一个不大于N的最大值和一个不小于N的最小值，并输出最大值和最小值？**

#include <stdio.h>

int main()

{int n,i,a[50]={0,1,1};

printf("请输入N的数值:");

scanf("%d",&n);

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

{a[i]=a[i-2]+a[i-1];

if(a[i]>=n)

break;

}

printf("不大于N的最大值是%d,", a[i]==n?a[i]:a[i-1]);

printf("不小于N的最小值是%d\n", a[i]);

}

**12、求一个5行5列的整型二维数组对角线元素之和（数组的值在运行时输入）？**

#include <stdio.h>

int main()

{

int i,j,sum=0;

int arr[5][5];

printf("请输入数组元素的值:\n");

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

{

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

{

scanf("%d",\*(arr+i)+j);

}

}

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

{

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

{

if(i==j) sum+=\*(\*(arr+i)+j);

if(i+j==4) sum+=\*(\*(arr+i)+j);

}

}

printf("对角线的和为:%d",sum-\*(\*(arr+2)+2));

return 0;

}

**13、输入一组字符串，第一个与最后一个相等 输出yes否输出no？**

oid same(char \*s) {

char \*r;

for(r = s; \*r; r++);

r--;

if (\*s == \*r) printf("yes"); else printf("no");

}

main()

{

char a[100];

printf("输入一个字符串：\n");

gets(a);

same(a);

}

**14、输入4\*3的矩阵，要求求出其中值最大的那个元素的值，以及其所在的行号和列号，要求该矩阵的值由键盘输入**？

main()

{

int a[4][3], i,j, max, m,n;

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

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

scanf( "%d", &a[i][j] ); //输入12个数

max=a[0][0];

m=n=0; //先假定第一个数是最大的,m、n是该数的下标

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

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

if ( max < a[i][j] ) //查找最大的数，并记录下其下标值

{

max=a[i][j];

m=i;n=j;

}

printf("max=%d row=%d col=%d", max, m, n );

}

**15、一个含有大小写的字符串，由调用函数的方法求出与第一个字母相同的字母个数？（先将输入的串放进数组中，取出第一个字符给一个变量，之后根据ASCII码来判断是否大小写**

**a[i]==b||a[i]==b-32||a[i]==b+32）**

**16、编写函数，实现从键盘上输入一个小写字母，将其转化为大写字母。**

#include<stdio.h>

int zhuanhua(char s);

void main(){

char s;

printf("请输入一个字符:");

scanf("%c",&s);

printf("转化前为:%c\n",s);

s=zhuanhua(s);

printf("转化后为:%c\n",s);

}

int zhuanhua(char s){

char S;

S=s-32;

return S;

}

**17、计算并输出500以内最大的10个能被13或17整除的自然数之和**

#include<stdio.h>

int jisuan(int n);

void main(){

int n=500;

printf("和是%d\n",jisuan(n));

}

int jisuan(int n){

int i,sum=0;

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

if(i%13==0||i%17==0)

sum+=i;

return sum;

}

**18、将字符串str中的小写字母全部转换成大写字符串。函数原型可声明为：“void  toUpperCase( char  str[ ]) ;".**

#include <stdio.h>

#include<string.h>

#define N 10

void toUpperCase( char str[ ]);

void main(){

char str[N];

printf("请输入一个字符串:");

gets(str);

toUpperCase(str);

printf("交换后的字符串为:");

puts(str);

}

void toUpperCase( char str[ ]){

int i;

for(i=0;;i++){

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

str[i]=str[i]-32;

else

break;

}

}

1. **编写函数，将一个字符串的全部有效元素逆置。函数原型可声明为“void reverseStr( char \* str ) ;”，参数str为指向字符串的指针。‘**

#include <stdio.h>

#include <string.h> 3

void reverseStr( char \*str ); 5

int main( )

{

char str[100];

puts(“请输入字符串：”);

gets(str);

reverseStr( str );

puts(“逆转后的字符串：”);

puts( str );14 return 0;

}

void reverseStr( char \*str )

{

char tmp;

char \*pi, \*pj;

pi = str;

pj = str + strlen( str ) – 1;

while (pi < pj )

{ tmp = \*pi; \*pi = \*pj; \*pj = tmp;

++pi; --pj;

}

}

**20、Fill the blanks of the following program. Function：output by the character w, such as W-shaped structure into a graphical.  
w       ww         w  
 w     w  w       w  
  w   w    w     w  
   w  w     w   w  
    ww        ww**#include <stdio.h>  
void draw(int n)  
{   
int i, j, k, r, m;  
for(i=1; ① ;i++)  
{  
for(j = 1; j <= 2; j++)  
{  
for(r = 1; r < i; r++) printf(" ");  
printf("w");  
for(k = 1; ② ; k++) printf(" ");  
printf("w");  
for(m = 1; m < i; m++) printf(" ");  
}  
③ ;  
}  
}  
#include <conio.h>  
int main( )  
{  
int n;  
printf("input a number:");  
④ ;  
draw(n);  
return 0;  
}  
答案：①i <= n;②k<=2\*(n-i);③ printf(“\n”);④scanf(“%d”, &n);

**21、编写函数，将一个整型数组的全部元素逆序存储，即若原来数组元素分别为12345，逆序存储后数组各元素变为54321。函数原型可声明为“void reverse( int \* p , int n );”，参数p为指向数组的指针，n为数组中的元素个数。**

#include <stdio.h>

#include <string.h>

void reverseStr( char \*str );

int main( )

{

char str[100];

puts(“请输入字符串：”);

gets(str);

reverseStr( str );

puts(“逆转后的字符串：”);

puts( str );

return 0; }

void reverseStr( char \*str )

{

char tmp;

char \*pi, \*pj;

pi = str;

pj = str + strlen( str ) – 1;

while (pi < pj )

{ tmp = \*pi; \*pi = \*pj; \*pj = tmp;

++pi; --pj;

}

}

**22、编写程序，用有参有返回值函数实现判断三个数是否能构成三角形。函数原型可声明为：“int isTriangle( double a, double b, double c );”，其中，a,b,c为三角形的三条边，返回值为0或1，0代表不能构成三角形，1代表能构成三角形。请在主函数中调用该函数完成程序的功能。**

#include <stdio.h>

int isTriangle( double a, double b, double c);//函数声明

int main( )

{

double ea, eb, ec;//三角形的三条边

int result; //是否是三角形的判断结果

printf(“Please input the three edges of a triangle:”);

scanf(“%lf%lf%lf”, &ea, &eb, &ec);

result = isTriangle( ea, eb, ec ); //函数调用

if( 0 == result )

printf(“这组边长不能构成三角形\n”);

else

printf(“这组边长可以构成三角形\n”);

return 0;

}

int isTriangle( double a, double b, double c ) //函数定义

{

if( a > 0 && b > 0 && c > 0 && a + b > c && b + c > a && c + a > b )

//三角形任意边长大于0，且 任意两边之和大于第三边

return 1; //可以构成三角形

else

return 0; //不能构成三角形

}

**23、Write a program that takes an integer keyed in from the terminal and extracts and displays each digit of the integer in English. So, if the user types 932, the program should display “ nine three two”.**

#include <stdio.h> int main( )

{

int num;int iDigit; //每一位上的数字

printf(“Pleasse input an integer:”);

scanf(“%d”, &num);

while ( num )

//当该数不为0时，继续提取其个位上的数字{

iDigit = num % 10; //提取个位

switch ( iDigit )

{

case 0: printf(“Zero “); break;

case 1: printf(“One “); break;

case 2: printf(“Two “); break;

case 3: printf(“Three “); break;

case 4: printf(“Four “); break;

case 5: printf(“Five “); break;

case 6: printf(“Six “); break;

case 7: printf(“Seven “); break;

case 8: printf(“Eight “); break;

case 9: printf(“Night “); break;

}

num /= 10; //去掉该数的个位}

printf(“\n”);return 0;

}

**24、Write a function to calculate the absolute value of x.**

#include <stdio.h>

double absoluteValue( double x );

int main( )

{

double x;

printf(“请输入x:”);

scanf(“%lf”, &x);

printf(“Absolute Value of x is: %lf\n”, absoluteValue( x ) );

printf(“请输入x:”); //第二次调用函数

scanf(“%lf”, &x);

printf(“Absolute Value of x is: %lf\n”, absoluteValue( x ) );

return 0;

}

double absoluteValue( double x )

{

if ( x > 0 )

return x;

else

return - x;

}

**25、从键盘输入10个互不相同的整数，找出其中最小的元素将其与数组中的第一个元进行交换。**

#include <stdio.h>

void getMin( int a[ ], int n); //函数声明

int main( )

{

int array[10];

int i;

printf(“请输入10个互不相等的整数：”);

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

scanf(“%d”, &array[i]);

getMin( array , 10 ); //调用函数来实现功能

printf(“将最小值换到第1个元素后：”);

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

printf(“%5d”, array[i]);

printf(“\n”);

return 0;

}

void getMin( int a[ ], int n) //函数定义

{

int i;

int tmp;

int minid=0;

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

if ( a [i] < a [minid] )

minid = i;

//将最小值换到第1个元素

tmp = a[0];

a[0] = a[minid];

a[minid] = tmp;

}

**26、有10本图书，从键盘输入每本图书的价格后，找出价格最高的图书并输出其价格。**

#include <stdio.h>

double getMaxPrice( double a[ ], int n); //函数声明

int main( )

{

double book[10];

int i;

int maxid;

printf(“请输入10本图书的价格：”);

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

scanf(“%lf”, &book[i]);

maxid = getMaxPrice ( book , 10 ); //调用函数来实现功能

printf(“最高书价：%.2lf\n” , book[maxid] );

return 0;

}

double getMaxPrice ( double a[ ], int n) //函数定义

{

int i;

int maxid =0;

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

if ( a [i] > a [maxid] )

maxid = i;

return maxid;

}

**27 、Writing a function to find the minimum subscript of element in an array ,and return the subscript to the function who calls it.**

#include <stdio.h>

int getMinid( int a[ ], int n); //函数声明

int main( )

{

int array[10];

int i;

int minid;

printf(“请输入10个整数：”);

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

scanf(“%d”, &array[i]);

minid = getMin( array , 10 ); //调用函数来实现功能

printf(“最小元素的下标：%d\n” , minid );

return 0;

}

int getMin( int a[ ], int n) //函数定义

{

int i;

int minid=0;

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

if ( a [i] < a [minid] )

minid = i;

return minid;

}

**28、从键盘输入若干整数（数据个数应小于20），其值在0至4的范围内，用-1作为输入结束的标志。编程统计输入的整数个数。**

#include <stdio.h>

int getCount( int a[ ], int n ); //函数声明

int main( )

{

int array[20] ={0} ; //用来存储最多20个整数

int c;

c = getCount( array , 20 );

printf(“输入的有效数的个数为：%d\n”, c);

return 0;

}

int getCount( int a[ ], int n ) //函数定义

{

int tmp; //临时存储输入的数

int count=0; //输入的有效数的个数

printf(“请输入若干个0-4之间的整数（以一1结束输入）：”);

scanf(“%d” , &tmp);

while ( tmp != -1 && count <20 )

{

if ( tmp >=0 && tmp <=4 ) //在范围之内

a[ count++] = tmp ; //将值转存到数组对应元素中

scanf(“%d”, &tmp);//输入下一个数

}

return count;

}

**29、编写函数，将一个整型数组的全部元素逆序存储，即若原来数组元素分别为1,2,3,4,5，逆序存储后数组各元素变为5,4,3,2,1。函数原型可声明为：“void reverse( int a[ ] , int n );”，参数a为数组，n为数组中的元素个数。**

#include <stdio.h>

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

int main( )

{

int array[10]={0};

int i;

printf(“请输入10个整数：”);

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

scanf(“%d”, &array[i]);

reverse( array, 10); //调用函数逆序存储数组中的数据

printf(“逆序后的元素为：\n”);

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

printf(“%5d”, array[i]);

printf(“\n”);

return 0;

}

void reverse( int a[ ], int n )

{

int i;

int tmp;

for( i=0; i<n/2; ++i)

{

tmp = a[i]; a[i] = a[n-i-1]; a[n-i-1] = tmp;

}

}

**30、编写函数，将一个十进制数转换成一个二进制数（提示：将转换后的二进制数各位的值依次存储在一个一维数组中，要输出时，只要逆序输出这个数组各元素的值即可）。函数原型可声明为：“int transformToBin( int dnum, int bin[ ] ) ;”，参数dnum是要转换的十进制数，bin是存储转换后的二进制值的数组(逆序存储的)，返回值是bin数组中元素的个数。**

1 #include <stdio.h>

2 int transformToBin( int dnum, int bin[ ] ) ;

3

4 int main( )

5 {

6 int array[32]={0}; //保存转换后的二进制数

7 int num; //待转换的整数

8 int cc; //最后得到的二进制总共多少位

9 printf(“请输入一个整数：”);

10 scanf(“%d”, &num);

11 cc = transformToBin( num, array ); //调用转换函数

12 cc--; //往回退一个元素下标，使cc指向最后一个元素

13 for( ; cc>=0; cc-- ) //输出转换后的二进制数

14 printf(“%d”, array[cc]);

15 printf(“\n”);

16 return 0;

17 }

18 int transformToBin( int dnum, int bin[ ] )

19 {

20 int count = 0;

21 while ( dnum ) //当dnum还未转换完毕

22 {

23 bin[count++] = dnum % 2; //余数保留到数组对应元素中

24 dnum /= 2; //数本身除2

25 }

26 return count;

27 }

28 #include <stdio.h>

29 int transformToBin( int dnum, int bin[ ] ) ; 30

31 int main( )

32 {

33 int array[32]={0}; //保存转换后的二进制数

34 int num; //待转换的整数

35 int cc; //最后得到的二进制总共多少位

36 printf(“请输入一个整数：”);

37 scanf(“%d”, &num);

38 cc = transformToBin( num, array ); //调用转换函数

39 cc--; //往回退一个元素下标，使cc指向最后一个元素

40 for( ; cc>=0; cc-- ) //输出转换后的二进制数

41 printf(“%d”, array[cc]);

42 printf(“\n”);

43 return 0;

44 }

45 int transformToBin( int dnum, int bin[ ] )

46 {

47 int count = 0;

48 while ( dnum ) //当dnum还未转换完毕

49 {

50 bin[count++] = dnum % 2; //余数保留到数组对应元素中

51 dnum /= 2; //数本身除2

52 }

53 return count;

54 }

55 #include <stdio.h>

56 int transformToBin( int dnum, int bin[ ] ) ;

57

58 int main( )

59 {

60 int array[32]={0}; //保存转换后的二进制数

61 int num; //待转换的整数

62 int cc; //最后得到的二进制总共多少位

63 printf(“请输入一个整数：”);

64 scanf(“%d”, &num);

65 cc = transformToBin( num, array ); //调用转换函数

66 cc--; //往回退一个元素下标，使cc指向最后一个元素

67 for( ; cc>=0; cc-- ) //输出转换后的二进制数

68 printf(“%d”, array[cc]);

69 printf(“\n”);

70 return 0;

71 }

72 int transformToBin( int dnum, int bin[ ] )

73 {

74 int count = 0;

75 while ( dnum ) //当dnum还未转换完毕

76 {

77 bin[count++] = dnum % 2; //余数保留到数组对应元素中

78 dnum /= 2; //数本身除2

79 }

80 return count;

81 }

82 #include <stdio.h>

83 int transformToBin( int dnum, int bin[ ] ) ;

84

85 int main( )

86 {

87 int array[32]={0}; //保存转换后的二进制数

88 int num; //待转换的整数

89 int cc; //最后得到的二进制总共多少位

90 printf(“请输入一个整数：”);

91 scanf(“%d”, &num);

92 cc = transformToBin( num, array ); //调用转换函数

93 cc--; //往回退一个元素下标，使cc指向最后一个元素

94 for( ; cc>=0; cc-- ) //输出转换后的二进制数

95 printf(“%d”, array[cc]);

96 printf(“\n”);

97 return 0;

98 }

99 int transformToBin( int dnum, int bin[ ] )

100 {

101 int count = 0;

102 while ( dnum ) //当dnum还未转换完毕

103 {

104 bin[count++] = dnum % 2; //余数保留到数组对应元素中

105 dnum /= 2; //数本身除2

106 }

107 return count;

108 }

**31、假设圆柱的底面积半径为r（= 2.5），高为h（= 3.5），编写求体积（体积=底面积\*高）的程序。**

1 #include <stdio.h>

2 #define PI 3.1415926

3 int main( )

4 {

5 double r = 2.5;

6 double h = 3.5;

7 double v=0;

8 v = (PI \* r \* r)\*h;

9 printf(“v = %lf\n”, v);

10 return 0;

11 }

**32、编写函数，将字符串str中的小写字母全部转换成大写字符串。函数原型可声明为：“void toUpperCase( char str[ ]) ; ”。**

#include <stdio.h>

#include <string.h>

void toUpperCase( char str[ ]);

int main( )

{char str[100];

puts(“请输入字符串：”);

gets(str);

toUpperCase ( str );

puts(“转换后的字符串：”);

puts( str );return 0;

}

void toUpperCase( char str[ ])

{int i;

i = 0;while ( str[i] != ‘\0’ )

{if( str[i] >= ‘a’ && str[i] <= ‘z’ )

str[i] -= 32; //小写字母转换成大写

++i;

}

}

**33、编写函数，删除字符串str中的所有ch字符。函数原型可声明为“void deleteAll(char str[ ] , char ch ) ;”，参数str为待处理的字符串，ch为要删除的字符。**

1 #include <stdio.h>

2 #include <string.h>

3 void deleteAll( char str[ ] , char ch );

4

5 int main( )

6 {

7 char str[100] , ch ;

8 puts(“请输入字符串：”);

9 gets(str);

10 printf(“请输入待删除的字符：”);

11 ch = getchar( );

12 deleteAll ( str , ch );

13 puts(“删除之后的字符串：”);

14 puts( str );

15 return 0;

16 }

17 void deleteAll( char str[ ] , char ch )

18 {

19 int oldi; //指向原str的

20 int newi; //指向删除ch后的str

21 oldi = 0 ;

22 newi = 0;

23 while ( str[ oldi ] != ‘\0’ )

24 {

25 if( str[oldi] != ch ) //不是要删除的字符

26 {

27 str[newi] = str[oldi]; //将其复制到新的字符串中

28 ++newi; //新字符串增长一个元素

29 }

30 ++oldi; //如果是要删除的字符，则会默认被跳过

31 }

32 str[newi] = '\0'; //新字符串置结束标志

33 }

**34、编写函数，用字符ch2替换字符串str中的字符ch1（注意：要全部都替换掉）。函数原型可声明为“void replaceAll( char str[ ], char ch1, char ch2 ) ;”。**

1 #include <stdio.h>

2 #include <string.h>

3 void replaceAll( char str[ ], char ch1, char ch2 );

4

5 int main( )

6 {

7 char str[100] , ch1 , ch2 ;

8 puts(“请输入字符串：”);

9 gets(str);

10 printf(“请输入要被替换的字符及用来替换的字符（中间不要用空格分隔）：”);

11 ch1 = getchar( );

12 ch2 = getchar( );

13 replaceAll ( str , ch1, ch2 );

14 puts(“替换之后的字符串：”);

15 puts( str );

16 return 0;

17 }

18

19 void replaceAll( char str[ ], char ch1, char ch2 )

20 {

21 int i;

22 i = 0;

23 while ( str[i] != ‘\0’ )

24 {

25 if( str[i] == ch1 )

26 str[i] = ch2;

27 ++i;

28 }

29 }

**35、编写程序，统计一个字符串中26个字母出现的次数（不区分大小写）。函数原型可声明为：“void countAlpha (char str[ ], int count[] );”，参数str为待处理的字符串，数组count长度为26，用于存放26个字母出现的次数。**

1 #include <stdio.h>

2 #include <string.h>

3 void countAlpha (char str[ ], int count[] );

4

5 int main( )

6 {

7 char str[100];

8 int count[26]={0};

9 int i;

10 puts(“请输入字符串：”);

11 gets(str);

12 countAlpha ( str , count );

13 puts(“统计结果：”);

14 for( i=0; i<26; ++i) //输出个数不为0的字母及其个数

15 if ( count[i] )

16 printf(“%c------%d\n”, i+’a’, count[i]);

17 return 0;

18 }

19 void countAlpha (char str[ ], int count[] )

20 {

21 int i;

22 i = 0;

23 while (str[i] != ‘\0’ )

24 {

25 if( str[i]>=’a’ && str[i] <= ‘z’)

26 ++count[ str[i] – ‘a’ ];

27 else if( str[i]>=’A’ && str[i] <= ‘Z’)

28 ++count[ str[i] – ‘A’ ];

29 ++i;

30 }

31 }

**36、编写程序输出下列图案 ：   
\*  
\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*\*\***

1 #include <stdio.h>

2 int main( )

3 {

4 printf(“ \*\n”);

5 printf(“ \* \* \*\n”);

6 printf(“ \* \* \* \* \*\n”);

7 printf(“ \* \* \* \* \* \* \*\n”);

8 return 0;

9 }

**37、编写程序，从键盘输入两个字符分别存放在变量x和y中，要求通过程序交换它们的值。**

1 #include <stdio.h>

2 int main( )

3 {

4 char x, y;

5 char tmp;

6 printf(“Input two characters:”);

7 scanf(“%c%c”, &x, &y);

8 printf(“Before swap: x=%c, y=%c\n”, x, y);

9 tmp = x;

10 x = y;

11 y = tmp;

12 printf(“After swap: x=%c, y=%c\n”, x, y);

13 return 0;

14 }

**38、**

**Write a program to evaluate the polynomial shown here: for x = 2.55.  
3x3-5x2+6.**

1 #include <stdio.h>

2 #include <math.h>

3 int main( )

4 {

5 double a=3, b=-5,c=2;

6 double x = 2.55;

7 double root1, root2;

8 double delt;

9 delt = b\*b – 4\*a\*c;

10 root1 = ( -b + sqrt( delt ) ) / (2 \* a) ;

11 root2 = ( -b - sqrt( delt ) ) / (2 \* a) ;

12 printf(“The two roots are:\n”);

13 printf(“root1 = %lf\n root2 = %lf\n”, root1, root2);

14 return 0;

15 }

**39、编写程序。功能：从读入的整数数据中，统计大于零的整数个数和小于零的整数个数。用输入零来结束输入，程序中用变量i统计大于零的整数个数，用变量j统计小于零的整数个数。**

1 #include <stdio.h>

2 int main( )

3 {

4 int num; //输入的整数

5 int iCount = 0; //大于0的整数的计数变量

6 int jCount = 0; //小于0的整数的计数变量

7 printf(“请输入多个整数：(以0结束输入)”);

8 scanf(“%d”, &num);

9 while ( num ) // while ( num != 0 )

10 {

11 if ( num > 0 ) ++iCount;

12 else if ( num < 0 ) ++jCount;

13 scanf(“%d”, &num);

14 }

15 printf(“大于0的个数：%d\n”, iCount);

16 printf(“小于0的个数：%d\n”, jCount);

17 return 0;

18 }

**40、功能：以每行5个数来输出300以内能被7或17整除的偶数，并求出其和。**

1 #include <stdio.h>

2 int main( )

3 {

4 int n; //300以内的数

5 int count = 0; //满足条件的数的统计变量

6 int sum = 0; //求和变量

7 for( n = 1; n <= 300; ++n)

8 {

9 if ( n % 2 == 0 && (n % 7 ==0 || n % 17 == 0) ) //是偶数，且能被7或17整除

10 {

11 sum += n; //求和

12 printf(“%5d”, n);//输出n值

13 ++count;

14 if( count % 5 == 0 ) //某行够5个数了，则换行

15 printf(“\n”);

16 }

17 }

18 printf(“\n”);

19 return 0;

20 }

**41、功能：计算并输出500以内最大的10个能被13或17整除的自然数之和。**

1 #include <stdio.h>

2 int main( )

3 {

4 int n; //500-1之间的整数

5 int sum = 0; //求和变量

6 int count = 0; //满足条件的数的个数

7 for ( n=500; n>0; --n)

8 {

9 if ( n % 13 == 0 || n % 17 == 0 )

10 {

11 sum += n;

12 ++count;

13 if ( count >= 10 ) //有10个满足条件的整数，提前结束循环14 break;

15 }

16 }

17 printf(“sum = %d\n”, sum );

18 return 0;

19 }

**42、功能：分别求出一批非零整数中的偶数、奇数的平均值，用零作为终止标记。**

1 #include <stdio.h>

2 int main( )

3 {

4 int n;//整数

5 int eSum = 0; //奇数的和

6 int oSum = 0; //偶数的和

7 int eCount = 0; //奇数的个数

8 int oCount = 0; //偶数的个数

9 printf(“请输入若干个整数（以0结束输入）：”);

10 scanf(“%d”, &n);

11 while ( n ) //当n不为0

12 {

13 if ( n % 2 ) //n为奇数

14 { eSum += n; ++eCount; }

15 else //否则， n为偶数

16 { oSum += n; ++oCount; }

17 scanf(“%d”, &n); //输入下一个整数

18 }

19 printf(“奇数平均值：%f\n”, 1.0 \* eSum / eCount );

20 printf(“偶数平均值：%f\n”, 1.0 \* oSum / oCount );

21 return 0;

22 }

**43、功能：百马百担问题：有100匹马，驮100担货，大马驮三担，中马驮2担，两匹小马驮一担，求大、中、小马各多少匹？**

1 #include <stdio.h>

2 int main( )

3 {

4 int bigHorse, midHorse, littleHorse; //大、中、小马的数量

5 for ( bigHorse = 0; bigHorse <= 33; ++bigHorse )

6 for( midHorse = 0; midHorse <= 50; ++midHorse )

7 {

8 littleHorse = 100 – bigHorse – midHorse; //小马数量

9 if ( littleHorse % 2 == 0 && (3\*bigHorse + 2\*midHorse + littleHorse / 2 == 100 ) )

//小马数量是偶数，且三种马驮的货是100担

10 printf(“大马：%d, 中马：%d, 小马：%d\n”, bigHorse, midHorse, littleHorse );

11 }

12 return 0;

13 }

**44、功能：百鸡问题：100元买100只鸡，公鸡一只5元钱，母鸡一只3元钱，小鸡一元钱三只，求100元钱能买公鸡、母鸡、小鸡各多少只？**

1 #include <stdio.h>

2 int main( )

3 {

4 int cock, hen, chick; //公鸡、母鸡、小鸡的数量

5 for (cock = 0; cock <= 20; ++ cock)

6 for(hen = 0; hen <= 33; ++ hen) 7 {

8 chick = 100 –cock–hen; //小鸡数量

9 if (chick % 3 == 0 && (5\* cock + 3\* hen + chick / 3 == 100 ) ) //小鸡数量是3的倍数，且三种鸡的总钱数是100

10 printf(“公鸡：%d, 母鸡：%d, 小鸡：%d\n”, cock, hen, chick);

11 }

12 return 0;

13 }

**45、功能：在屏幕上用\*输出一个漏斗状的图形。**

1 #include <stdio.h>

2 void printFunnel( int n); //函数声明

3 int main( )

4 {

5 int n; //上半部的总行数

6 printf(“请输入漏斗上半部的总行数：”);

7 scanf(“%d”, &n);

8 printFunnel( n ); //调用函数打印漏斗形

9 return 0;

10 }

11

12 void printFunnel( int n)

13 {

14 int row; //行号

15 int starCount; // 某行上星号的数量

16 int spaceCount; //某行上空格的数量

17 for( row = 1; row <= n; ++row)

18 {

19 for( spaceCount = 1; spaceCount <= row – 1 ; ++ spaceCount)

20 printf(“ “); //打印出某行上星号前的空格

21 for( starCount = 1; starCount <= 2\*( n – row ) + 1; ++starCount )

22 printf(“\* “); //打印出某行上的所有星号

23 printf(“\n”); //换行

24 }

25

26 //打印下半部分(可看成一个n-2行的三角形状)

27 for( row = 2; row <= n; ++row)

28 {

29 for( spaceCount = 1; spaceCount <= n - row ; ++ spaceCount)

30 printf(“ “); //打印出某行上星号前的空格

31 for( starCount = 1; starCount <= 2\* row - 1; ++starCount )

32 printf(“\* “); //打印出某行上的所有星号

33 printf(“\n”); //换行

34 }

35 }

**46、功能：输出100到1000之间的各位数字之和能被15整除的所有数，输出时每10个数一行。**

1 #include <stdio.h>

2 int main( )

3 {

4 int n;//100到1000之间的整数

5 int tmpn;//临时存储n

6 int sum; //统计n的各位数字之和

7 int count = 0; //输出数的个数

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

9 {

10 tmpn = n; //将n临时存储到tmpn中

11 sum = 0; //每个n要重新计算各位数字之和

12 while ( tmpn ) //当tmpn不为0时

13 {

14 sum += tmpn % 10;

15 tmpn /= 10;

16 }

17 if ( sum % 15 == 0 ) //如果n满足指定的条件

18 {

19 printf(“%5d”, n); //输出当前n值

20 ++count;

21 if ( count % 10 == 0 ) //一行输出了10个整数，则换行

22 printf(“\n”);

23 }

24 }

25 printf(“\n”);

26 return 0;

27 }

**47、编写程序，从键盘输入两个数字字符并分别存放在字符型变量x和y中，要求通过程序将这两个字符对应的数字相加后输出。**

1 #include <stdio.h>

2 int main( )

3 {

4 char x , y;

5 printf(“Input two characters:”);

6 scanf(“%c%c”, &x, &y);

7 if( x>=’0’&&x<=’9’ && y>=’0’&&y<=’9’ )

8 printf(“%d \n”, x-'0'+y-'0');

9 return 0;

10 }

**48、某市不同车牌的出租车3公里的起步价和计费分别为：夏利7元/公里，3公里以外2.1元/公里；富康8元/公里，3公里以外2.4元/公里；桑塔纳9元，3公里以外2.7元/公里。编程：从键盘输入乘车的车型及公里数，输出应付的车资。**

1 #include <stdio.h>

2 int main( )

3 {

4 int carType;//车型。1代表夏利；2代表富康；3代表桑塔纳

5 double xiali = 2.1; //每公里价格

6 double fukang = 2.4;

7 double sangtana = 2.7;

8 double distance; //距离

9 double totalMoney;//总的收费

10 printf("请输入您乘坐的车型：1代表夏利；2代表富康；3代表桑塔纳：");

11 scanf("%d", &carType);

12 printf("请输入您乘车的总路程：");

13 scanf("%lf", &distance);

14 if( carType == 1)//夏利

15 {

16 if( distance < 3 )

17 totalMoney = 7.0;

18 else

19 totalMoney = 7 + xiali \* (distance – 3);

20 }

21 else if( carType == 2 ) //富康

22 {

23 if( distance < 3 )

24 totalMoney = 8.0;

25 else

26 totalMoney = 8 + fukang \* (distance – 3);

27 }

28 else if( carType == 3 ) //富康

29 {

30 if( distance < 3 )

31 totalMoney = 9.0;

32 else

33 totalMoney = 9 + sangtana \* (distance – 3);

34 }

35

36 printf("（四舍五入）您的车费为：%.0lf\n", totalMoney );

37 return 0;

38 }

**49、编写程序，从键盘上输入一个小写字母，将其转化为大写字母。**

1 #include <stdio.h>

2 int main( )

3 {

4 char lower;

5 printf( “Input a lower alpha:”);

6 scanf(“%c”, &lower);

7 if( lower>=’a’ && lower<=’z’ )

8 printf(“%c \n”, lower - 32);

9 return 0;

10 }

**50、编写函数，将数组s1中的全部奇数都复制到数组s2中。函数原型可声明为：“int copyTo( int \* s1 , int n, int \* s2 ) ; ”，参数s1和s2为指向两个数组的指针，n为数组s1中元素的个数，返回值为复制完成后s2中元素的个数。**

1 #include <stdio.h>

2 #include <stdlib.h>

3 #include <time.h>

4 int copyTo( int \*s1, int n , int \*s2 );

5 int main( )

6 {

7 int s1[100], s2[100];

8 int count; //最终奇数的个数

9 int i;

10 srand( time ( NULL ) ); //随机种子

11 for ( i = 0; i < 100; ++i ) //避免每次执行程序重复输入

12 s1[i] = rand( ) % 1001 ; //随机生成0-1000之间的整数

13 for ( i = 0; i < 100; ++i ) //打印出数组s1元素

14 printf(“%5d”, s1[i] ) ;

15 printf(“\n”);

16 count = copyTo( s1, 100, s2 );

17

18 printf(“奇数如下：\n”);

19 for ( i = 0; i < count; ++i ) //打印出数组s2元素

20 printf(“%5d”, s2[i] ) ;

21 printf(“\n”);

22 return 0;

23 }

24 int copyTo( int \*s1, int n , int \*s2 )

25 {

26 int \*ps1, \*ps2;

27 ps1 = s1; ps2 = s2;

28 while ( ps1 < s1 + n )

29 {

30 if( \*ps1 % 2) //是奇数

31 \*ps2++ = \*ps1;

32 ps1++;

33 }

34 return ps2 – s2;

35 }

**51、编写函数，将字符串中的大写字母转换为小写字母。函数原型可声明为“void strToLow( char \* str ) ;”，参数str是要转换的字符串。**

1 #include <stdio.h>

2 #include <string.h>

3

4 void toLowerCase( char \*str);

5

6 int main( )

7 {

8 char str[100];

9 puts(“请输入字符串：”);

10 gets(str);

11 toLowerCase ( str );

12 puts(“转换后的字符串：”);

13 puts( str );

14 return 0;

15 }

16

17 void toLowerCase ( char \*str )

18 {

19 while ( \*str )

20 {

21 if(\*str >= ‘A’ && \*str <= ‘Z’ )

22 \*str += 32; //小写字母转换成大写

23 ++str;

24 }

25 }

**注**：从18年开始机试题目偏难，有一些是从CCF上抽的类似题，建议有时间的从上面多练习，掌握一些算法思想；第一个链接给出历年的CCF真题和答案链接，自己也可以去网上搜一些近年的题目练习。

第二个链接是我根据云大的题型找到的一些经典算法，有时间也可以看看，做一些参考。

1. <https://wenku.baidu.com/view/bead478d336c1eb91b375dab.html>

[https://blog.csdn.net/tigerisland45/article/details/54460662、https://wenku.baidu.com/view/2a5eced231126edb6e1a1033.html](https://blog.csdn.net/tigerisland45/article/details/54460662、https:/wenku.baidu.com/view/2a5eced231126edb6e1a1033.html)

3、http://www.cspro.org（CCF认证系统）