编程题参考答案（**10-12** 周）

第十周**\_**在线测试

**1.** 水手分椰子

#include <stdio.h>

int Coconut(int n);

int main()

{

int num;

printf("Input n(n<=5):\n");

scanf("%d",&num);

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

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;

}

**2.** 找最值

#include <stdio.h>

int FindMax(int num[], int n, int \*pMaxPos);

int FindMin(int num[], int n, int \*pMinPos);

int main(void)

{

int num[10], maxValue, maxPos, minValue, minPos, i;

printf("Input 10 numbers:\n");

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

{

}

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

// 输入 10 个数

maxValue = FindMax(num, 10, &maxPos); // 找最大值及其所在下标位置

minValue = FindMin(num, 10, &minPos); // 找最小值及其所在下标位置

printf("Max=%d,Position=%d,Min=%d,Position=%d\n", maxValue, maxPos, minValue,

minPos);

return 0;

}

//函数功能：求有 n 个元素的整型数组 num 中的最大值及其所在下标位置，函数返回最大

值

int FindMax(int num[], int n, int \*pMaxPos)

{

int i, max;

max = num[0];

\*pMaxPos = 0;

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

{

//假设 num[0]为最大值

//假设最大值在数组中的下标位置为 0

if (num[i] > max)

{

max = num[i];

\*pMaxPos = i;

//pMaxPos 指向最大值数组元素的下标位置

}

}

return max ;

}

//函数功能：求有 n 个元素的整型数组 num 中的最小值及其所在下标位置，函数返回最小

值

int FindMin(int num[], int n, int \*pMinPos)

{

int i, min;

min = num[0];

//假设 num[0]为最小

\*pMinPos = 0;

//假设最小值在数组中的下标位置为 0

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

{

if (num[i] < min)

{

min = num[i];

\*pMinPos = i; //pMinPos 指向最小值数组元素的下标位置

}

}

return min ;

**3.** 星期查找

}

#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

char

findFlag = 0;

x[MAX\_STR\_LEN];

/\* 置找到标志为假 \*/

char weekDay[][MAX\_STR\_LEN] = {"Sunday", "Monday", "Tuesday",

"Wednesday", "Thursday", "Friday",

"Saturday"};

printf("Please enter a string:");

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;

}

**4.** 杨辉三角形

#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);

YHTriangle(a,n);

PrintYHTriangle(a,n);

return 0;

//根据要求输入杨辉三角形的行数

}

//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");

}

}

第十周**\_**练兵区

**1.** 有趣的“回文”检测

#include <stdio.h>

#include <string.h>

int main(void)

{

char str[80], \*pStart, \*pEnd;

int len;

printf("Input string:");

gets(str);

len = strlen(str);

pStart = str;

pEnd = str + len - 1;

while ((\*pStart == \*pEnd) && (pStart < pEnd))

{

pStart++;

pEnd--;

}

if (pStart < pEnd)

{

printf("No!\n");

}

else

{

printf("Yes!\n");

}

return 0;

}

**2.** 学生成绩管理系统 **V1.0**

#include <stdio.h>

#define STU\_NUM 30

/\* 最多的学生人数 \*/

int

Menu(void);

void ReadScore(long num[], float score[], int n);

void AverSumofScore(float score[], int n);

void DeSortbyScore(long num[], float score[], int n);

void AsSortbyNum(long num[], float score[], int n);

void SearchbyNum(long num[], float score[], int n);

void StatisticAnalysis(float score[], int n);

void PrintScore(long num[], float score[], int n) ;

int main()

{

char

int

itemSelected;

n = 0;

float score[STU\_NUM];

long num[STU\_NUM];

printf("Input student number(n<30):\n");

scanf("%d", &n);

while (1)

{

itemSelected = Menu(); /\* 显示菜单，并读取用户输入 \*/

switch (itemSelected)

{

case 1:

ReadScore(num, score, n);

break;

case 2: AverSumofScore(score, n);

break;

case 3: DeSortbyScore(num, score, n);

printf("Sort in descending order by score:\n");

PrintScore(num, score, n);

break;

case 4: AsSortbyNum(num, score, n);

printf("Sort in ascending order by number:\n");

PrintScore(num, score, n);

break;

case 5: SearchbyNum(num, score, n);

break;

case 6: StatisticAnalysis(score, n);

break;

case 7: PrintScore(num, score, n);

break;

case 0: printf("End of program!");

exit(0);

default: printf("Input error!\n");

}

}

return 0;

}

/\* 函数功能：显示菜单并获得用户键盘输入的选项 \*/

int Menu(void)

{

int itemSelected;

printf("Management for Students' scores\n");

printf("1.Input record\n");

printf("2.Caculate total and average score of course\n");

printf("3.Sort in descending order by score\n");

printf("4.Sort in ascending order by number\n");

printf("5.Search by number\n");

printf("6.Statistic analysis\n");

printf("7.List record\n");

printf("0.Exit\n");

printf("Please Input your choice:\n");

scanf("%d", &itemSelected); /\* 读入用户输入 \*/

return itemSelected;

}

/\* 函数功能：输入 n 个学生的某门课成绩 \*/

void ReadScore(long num[], float score[], int n)

{

int i;

printf("Input student's ID, name and score:\n");

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

{

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

}

}

/\* 函数功能：计算全班总分和平均分 \*/

void AverSumofScore(float score[], int n)

{

int

i;

float sum = 0;

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

{

sum = sum + score[i];

}

printf("sum=%.0f,aver=%.2f\n", sum, n>0 ? sum/n : 0);

}

/\* 函数功能：按选择法将数组 score 的元素值按从高到低排序 \*/

void DeSortbyScore(long num[], float score[], int n)

{

int

i, j, k;

float temp1;

long temp2;

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

{

k = i;

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

{

if (score[j] > score[k]) k = j;

}

if (k != i)

{

/\* 交换成绩 \*/

temp1 = score[k]; score[k] = score[i]; score[i] = temp1;

/\* 交换学号 \*/

temp2 = num[k]; num[k] = num[i]; num[i] = temp2;

}

}

}

/\* 函数功能：按选择法将数组 num 的元素值按从低到高排序 \*/

void AsSortbyNum(long num[], float score[], int n)

{

int

i, j, k;

float temp1;

long temp2;

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

{

k = i;

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

{

if (num[j] < num[k]) k = j;

}

if (k != i)

{

/\* 交换成绩 \*/

temp1 = score[k]; score[k] = score[i]; score[i] = temp1;

/\* 交换学号 \*/

temp2 = num[k]; num[k] = num[i]; num[i] = temp2;

}

}

}

/\* 函数功能：按学号查找学生成绩并显示查找结果 \*/

void SearchbyNum(long num[], float score[], int n)

{

long number;

int

i;

printf("Input the number you want to search:\n");

scanf("%ld", &number);

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

{

if (num[i] == number)

{

printf("%ld\t%.0f\n", num[i], score[i]);

return;

}

}

printf("Not found!\n");

}

/\* 函数功能：统计各分数段的学生人数及所占的百分比 \*/

void StatisticAnalysis(float score[], int n)

{

int i, total, t[6] = {0,0,0,0,0,0};

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

{

if (score[i]>=0 && score[i]<60) t[0]++;

else if (score[i]<70)

else if (score[i]<80)

else if (score[i]<90)

t[1]++;

t[2]++;

t[3]++;

else if (score[i]<100)

t[4]++;

t[5]++;

else if (score[i] == 100)

}

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

{

total = total + t[i];

}

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

{

if (i == 0)

{

printf("<60\t%d\t%.2f%%\n", t[i], (float)t[i]/n\*100);

}

else if (i == 5)

{

printf("%d\t%d\t%.2f%%\n",(i+5)\*10, t[i],(float)t[i]/n\*100);

}

else

{

printf("%d-%d\t%d\t%.2f%%\n", (i+5)\*10, (i+5)\*10+9, t[i],

(float)t[i]/n\*100);

}

}

}

/\* 函数功能： 打印学生成绩 \*/

void PrintScore(long num[], float score[], int n)

{

int i;

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

{

printf("%ld\t%.0f\n", num[i], score[i]);

}

}

**3.** 程序改错——**1**

#include <stdio.h>

#include <string.h>

char\* MyStrcat(char \*dest, char \*source);

int main(void)

{

char first[80], second[80];

printf("Input the first string:\n");

gets(first);

printf("Input the second string:\n");

gets(second);

printf("The result is : %s\n", MyStrcat(first, second));

return 0;

}

char\* MyStrcat(char \*dest, char \*source)

{

int i = 0;

while (\*(dest+i)!='\0') i++;

for (; \*source!='\0'; i++,source++)

{

\*(dest+i) = \*source;

}

\*(dest+i)='\0';

return dest;

}

**4. }**程序改错——**2**

#include<stdio.h>

#define ARR\_SIZE

5

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

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

int main(void)

{

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

YH(a, ARR\_SIZE);

PrintYH(a, ARR\_SIZE);

return 0;

}

void YH(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];

}

}

}

void PrintYH(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");

}

}

**5.** 出售金鱼

#include <stdio.h>

int main()

{

int i,j,n,x;

n=0;

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;

}

}

return 0;

}

第十一周**\_**在线测试

**1.** 找数组最值

#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;

\*pCol = j;

}

/\*记录行下标\*/

/\*记录列下标\*/

}

}

return max;

}

**2.** 冒泡排序

#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;

}

}

}

}

**3.** 删除字符串中与某字符相同的字符

#include <stdio.h>

void Squeeze(char s[], char c);

int main()

{

char str[20], ch;

printf("Input a string:\n");

gets(str);

printf("Input a character:\n");

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 \*/

}

4. 求最大数和最小数的最大公约数

#include <stdio.h>

#define N 10

int Gcd(int a, int b)

{

int tmp, c;

if (a==0 || b==0)

{

return 0;

}

else

{

if (a < b)

{

tmp = a;

a = b;

b = tmp;

}

}

c = a % b;

while (c)

{

a = b;

b = c;

c = a % b;

}

return b;

}

int main()

{

int a[N];

int i, maxNum, minNum, res;

printf("Input 10 numbers:\n");

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

{

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

}

maxNum = a[0];

minNum = a[0];

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

{

maxNum = a[i] > maxNum ? a[i] : maxNum;

minNum = a[i] < minNum ? a[i] : minNum;

}

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

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

res = Gcd(maxNum, minNum);

if (res!=0)

{

printf("%d", res);

}

return 0;

}

第十一周**\_**练兵区

**1.** 找出按字典顺序排在最前面的国名

#include <stdio.h>

#include <string.h>

#define N 80

int main()

/\* 字符串最大长度 \*/

{

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);

for (n=1; n<5; n++)

{

/\* 将其作为最小字符串暂存 \*/

gets(str);

/\* 每次输入一个字符串 \*/

/\* 比较两个字符串的大小 \*/

if (strcmp(str, min) < 0)

{

}

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 \*/

}

2. 学生成绩管理系统 **V2.0**

#include <stdio.h>

#include <stdlib.h>

#define STU\_NUM 30

/\* 最多的学生人数 \*/

int

Menu(void);

void ReadScore(long num[], float score[], int n);

void AverSumofScore(float score[], int n);

void SortbyScore(long num[], float score[], int n, int (\*compare)(float a, float b));

int

int

Ascending(float a, float b);

Descending(float a, float b);

void SwapFloat(float \*x, float \*y);

void SwapLong(long \*x, long \*y);

void AsSortbyNum(long num[], float score[], int n);

void SearchbyNum(long num[], float score[], int n);

void StatisticAnalysis(float score[], int n);

void PrintScore(long num[], float score[], int n) ;

int main()

{

char

int

itemSelected;

n = 0;

float score[STU\_NUM];

long num[STU\_NUM];

printf("Input student number(n<%d):\n", STU\_NUM);

scanf("%d", &n);

while (1)

{

itemSelected = Menu(); /\* 显示菜单，并读取用户输入 \*/

switch (itemSelected)

{

case 1:

ReadScore(num, score, n);

break;

case 2: AverSumofScore(score, n);

break;

case 3: SortbyScore(num, score, n, Descending);

printf("Sort in descending order by score:\n");

PrintScore(num, score, n);

break;

case 4: SortbyScore(num, score, n, Ascending);

printf("Sort in ascending order by score:\n");

PrintScore(num, score, n);

break;

case 5: AsSortbyNum(num, score, n);

printf("Sort in ascending order by number:\n");

PrintScore(num, score, n);

break;

case 6: SearchbyNum(num, score, n);

break;

case 7: StatisticAnalysis(score, n);

break;

case 8: PrintScore(num, score, n);

break;

case 0: printf("End of program!");

exit(0);

default: printf("Input error!\n");

}

}

return 0;

}

/\* 函数功能：显示菜单并获得用户键盘输入的选项 \*/

int Menu(void)

{

int itemSelected;

printf("Management for Students' scores\n");

printf("1.Input record\n");

printf("2.Caculate total and average score of course\n");

printf("3.Sort in descending order by score\n");

printf("4.Sort in ascending order by score\n");

printf("5.Sort in ascending order by number\n");

printf("6.Search by number\n");

printf("7.Statistic analysis\n");

printf("8.List record\n");

printf("0.Exit\n");

printf("Please Input your choice:\n");

scanf("%d", &itemSelected); /\* 读入用户输入 \*/

return itemSelected;

}

/\* 函数功能：输入 n 个学生的某门课成绩 \*/

void ReadScore(long num[], float score[], int n)

{

int i;

printf("Input student's ID and score:\n");

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

{

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

}

}

/\* 函数功能：计算全班总分和平均分 \*/

void AverSumofScore(float score[], int n)

{

int

i;

float sum = 0;

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

{

sum = sum + score[i];

}

printf("sum=%.0f,aver=%.2f\n", sum, n>0 ? sum/n : 0);

}

/\* 函数功能：按选择法将数组 score 的元素值排序 \*/

void SortbyScore(long num[], float score[], int n,

int (\*compare)(float a, float b))

{

int

i, j, k;

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

{

k = i;

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

{

if ((\*compare)(score[j], score[k])) k = j;

}

if (k != i)

{

SwapFloat(&score[k], &score[i]); /\* 交换成绩 \*/

SwapLong(&num[k], &num[i]); /\* 交换学号 \*/

}

}

}

/\* 使数据按升序排序 \*/

int Ascending(float a, float b)

{

return a < b;

/\* 这样比较决定了按升序排序，如果 a<b，则交换 \*/

}

/\* 使数据按降序排序 \*/

int Descending(float a, float b)

{

return a > b;

/\* 这样比较决定了按降序排序，如果 a>b，则交换 \*/

}

/\* 交换两个单精度浮点型数据 \*/

void SwapFloat(float \*x, float \*y)

{

float temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* 交换两个长整型数据 \*/

void SwapLong(long \*x, long \*y)

{

long temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* 函数功能：按选择法将数组 num 的元素值按从低到高排序 \*/

void AsSortbyNum(long num[], float score[], int n)

{

int

i, j, k;

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

{

k = i;

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

{

if (num[j] < num[k]) k = j;

}

if (k != i)

{

SwapFloat(&score[k], &score[i]); /\* 交换成绩 \*/

SwapLong(&num[k], &num[i]); /\* 交换学号 \*/

}

}

}

/\* 函数功能：按学号查找学生成绩并显示查找结果 \*/

void SearchbyNum(long num[], float score[], int n)

{

long number;

int

i;

printf("Input the number you want to search:\n");

scanf("%ld", &number);

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

{

if (num[i] == number)

{

printf("%ld\t%.0f\n", num[i], score[i]);

return;

}

}

printf("Not found!\n");

}

/\* 函数功能：统计各分数段的学生人数及所占的百分比 \*/

void StatisticAnalysis(float score[], int n)

{

int i, total, t[6] = {0,0,0,0,0,0};

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

{

if (score[i]>=0 && score[i]<60)t[0]++;

else if (score[i]<70)

else if (score[i]<80)

else if (score[i]<90)

else if (score[i]<100)

else if (score[i] == 100)

t[1]++;

t[2]++;

t[3]++;

t[4]++;

t[5]++;

}

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

{

total = total + t[i];

}

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

{

if (i == 0) printf("<60\t%d\t%.2f%%\n",t[i],(float)t[i]/n\*100);

else if (i == 5) printf("%d\t%d\t%.2f%%\n", (i+5)\*10,t[i],(float)t[i]/n\*100);

else printf("%d-%d\t%d\t%.2f%%\n", (i+5)\*10, (i+5)\*10+9, t[i], (float)t[i]/n\*100);

}

}

/\* 函数功能： 打印学生成绩 \*/

void PrintScore(long num[], float score[], int n)

{

int i;

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

{

printf("%ld\t%.0f\n", num[i], score[i]);

}

}

**3.** 月份表示

#include <stdio.h>

#define ROW 3

#define COL 4

void Transpose(int \*a, int \*at, int row, int col);

void InputMatrix(int \*s, int row, int col);

void PrintMatrix(int \*s, int row, int col);

int main(void)

{

int n;

char \*monthName[]={"Illegal month", "January", "February","March","April", "May", "June",

"July", "August", "September", "October", "November", "December"};

printf("Input month number:\n");

scanf("%d", &n);

//输入月份

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

}

return 0;

}

**4.** 程序改错——**1**

#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 \*score, int sum[], float aver[], int m, int n);

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

int main(void)

{

int

i, j, m, n, score[STUD][COURSE]={0}, sum[STUD];

aver[STUD];

float

printf("Enter the total number of students and courses:\n");

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

printf("Enter score:\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 \*score, 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] + \*(score + i \* n + j);

}

if (n<COURSE) score=score+(COURSE-n);

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

}

}

void Print(int \*score, 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", \*(score + i \* n + j));

}

if (n<COURSE) score=score+(COURSE-n);

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

}

}

**5.** 程序改错——**2**

#include <stdio.h>

#include <string.h>

#define

#define

M

N

150 /\* 最多的字符串个数 \*/

10 /\* 字符串最大长度 \*/

void SortString(char \*ptr[], int n);

int main()

{

int

i, n;

char name[M][N];

char \*pStr[M];

printf("How many countries?\n");

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++)

{

pStr[i]=name[i];

gets(name[i]); /\* 输入 n 个字符串 \*/

}

SortString(pStr, n); /\* 字符串按字典顺序排序 \*/

printf("Sorted results:\n");

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

{

puts(pStr[i]); /\* 输出排序后的 n 个字符串 \*/

}

return 0;

}

void SortString(char \*ptr[], int n)

{

int

i, j;

char \*temp = NULL;

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

{

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

{

if (strcmp(ptr[j], ptr[i]) < 0)

{

temp = ptr[i];

ptr[i] = ptr[j];

ptr[j] = temp;

}

}

}

}

第十二周**\_**在线测试

**1.** 百万富翁的换钱计划

#include <stdio.h>

int main()

{

int j;

double toStranger = 0; /\* 富翁给陌生人的钱，以'元'为单位 \*/

double toRichman = 0; /\* 陌生人给富翁的钱，以'元'为单位 \*/

double term = 0.01;

/\* 富翁第一天给陌生人 0.01 元 \*/

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

{

toRichman += 100000; /\* 陌生人每天给富翁 10 万元 \*/

toStranger += term;

term = term \* 2;

/\* 富翁每天给陌生人的钱是前一天的两倍 \*/

}

printf("to Stranger: %.2f yuan\n", toStranger);

printf("to Richman: %.2f yuan\n", toRichman);

return 0;

}

2. 用计数控制的循环实现正数累加求和

#include <stdio.h>

#include <string.h>

int main()

{

int

sum=0, n,i=0;

do{

printf("Input a number:\n");

scanf("%d",&n);

if (n>0)

{

sum=sum+n;

i=i+1;

}

} while (n!=0);

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

return 0;

}

**3.** 平方根表

#include <stdio.h>

#include <math.h>

int main()

{

int m, n;

int num;

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

scanf("%d",&num);

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

{

printf("%7d", m);

}

//输出表头

printf("\n");

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

{

//乘数 n 从 1 变化到 9

//输出每行的开头数字

printf("%d", n);

for (m=0; m<num; 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,准备输出下一行

}

return 0;

}

**4.** 最大公约数

#include <stdio.h>

int CommonFactors(int a, int b);

int main(void)

{

int sub;

int a,b;

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

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

while((sub=CommonFactors(a, b)) > 0)

{

static int counter = 1;

printf("Common factor %d is %d\n", counter++, sub);

}

return 0;

}

//函数功能： 指明计算哪两个数的公约数

int CommonFactors(int a, int b)

{

static int num1 = -1;

static int num2 = -1;

static int curFactor;

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

if (num1 != a || num2 != b)

// 使用了新的参数

{

num1 = a;

num2 = b;

curFactor = a > b ? b : a; // curFactor 置为两个数中较小的那个

}

// 因从大到小求公约数，所以从 a,b 的最小值开始查找公约数，直到全部找到为止

while(curFactor>0)

{

if (a%curFactor == 0 && b%curFactor == 0)

{

return curFactor--;

curFactor--;

return -1;

// 如果不减 1，则下次还会测试这个数

}

}

}

第十二周**\_**练兵区

**1.** 大奖赛现场统分

#include <stdio.h>

#include <math.h>

#define ATHLETE 40

#define JUDGE

/\* 选手人数最高限 \*/

/\* 评委人数最高限 \*/

20

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];

int ph[JUDGE];

/\* 选手的编号 \*/

/\* 评委的编号 \*/

/\* 选手的最后得分 \*/

/\* 评委的得分 \*/

float sf[ATHLETE];

float pf[JUDGE];

float f[ATHLETE][JUDGE];

printf("How many Athletes?\n");

scanf("%d", &n);

/\* 评委给选手的评分 \*/

/\* 输入选手人数 \*/

/\* 输入评委人数 \*/

printf("How many judges?\n");

scanf("%d", &m);

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

{

ph[j] = j;

}

printf("Scores of Athletes:\n");

CountAthleteScore(sh, sf, n, \*f, m);

CountJudgeScore(ph, pf, m, sf, \*f, n);

printf("Order of Athletes:\n");

Sort(sh, sf, n);

/\* 现场为选手统计分数 \*/

/\* 为各个评委打分 \*/

/\* 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("Athlete %d is playing.\n", i);

printf("Please enter his number code:\n");

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

sf[i] = 0;

max = 0;

/\* 最高分初值设为最小值 \*/

/\* 最低分初值设为最大值 \*/

/\* 第 j 个评委 \*/

min = 100;

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

{

printf("Judge %d gives score:\n", j);

scanf("%f", &f[i\*m+j]);

sf[i] = sf[i] + f[i\*m+j];

if (max < f[i\*m+j])

max = f[i\*m+j];

/\* 累加评委对第 i 个选手的评分 \*/

/\* 找出最高分 \*/

if (min > f[i\*m+j])

min = f[i\*m+j];

/\* 找出最低分 \*/

}

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 \*/

/\* 第 i 个选手 \*/

pf[j] = 0;

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

{

pf[j] = pf[j] + (f[i\*m+j] - sf[i]) \* (f[i\*m+j] - sf[i]);

pf[j] = 10 - sqrt(pf[j]/n);

}

}

}

**2.** 学生成绩管理系统 **V3.0**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_LEN 10

#define STU\_NUM 30

/\* 字符串最大长度 \*/

/\* 最多的学生人数 \*/

int

Menu(void);

void ReadScore(long num[], char name[][MAX\_LEN], float score[], int n);

void AverSumofScore(float score[], int n);

void SortbyScore(long num[], char name[][MAX\_LEN], float score[], int n, int (\*compare)(float a,

float b));

int

Ascending(float a, float b);

int

Descending(float a, float b);

void SwapFloat(float \*x, float \*y);

void SwapLong(long \*x, long \*y);

void SwapChar(char x[], char y[]);

void AsSortbyNum(long num[], char name[][MAX\_LEN], float score[], int n);

void SortbyName(long num[], char name[][MAX\_LEN], float score[], int n);

void SearchbyNum(long num[], char name[][MAX\_LEN], float score[], int n);

void SearchbyName(long num[], char name[][MAX\_LEN], float score[], int n);

void StatisticAnalysis(float score[], int n);

void PrintScore(long num[], char name[][MAX\_LEN], float score[], int n) ;

int main()

{

char ch;

int

n = 0;

float score[STU\_NUM];

long num[STU\_NUM];

char name[STU\_NUM][MAX\_LEN];

printf("Input student number(n<30):\n");

scanf("%d", &n);

while (1)

{

ch = Menu();

switch (ch)

{

/\* 显示菜单，并读取用户输入 \*/

case 1:

ReadScore(num, name, score, n);

break;

case 2: AverSumofScore(score, n);

break;

case 3: SortbyScore(num, name, score, n, Descending);

printf("Sort in descending order by score:\n");

PrintScore(num, name, score, n);

break;

case 4: SortbyScore(num,name, score, n, Ascending);

printf("Sort in ascending order by score:\n");

PrintScore(num, name, score, n);

break;

case 5: AsSortbyNum(num, name, score, n);

printf("Sort in ascending order by number:\n");

PrintScore(num, name, score, n);

break;

case 6: SortbyName(num, name, score, n);

printf("Sort in dictionary order by name:\n");

PrintScore(num, name, score, n);

break;

case 7: SearchbyNum(num, name, score, n);

break;

case 8: SearchbyName(num, name, score, n);

break;

case 9: StatisticAnalysis(score, n);

break;

case 10: PrintScore(num, name, score, n);

break;

case 0: printf("End of program!");

exit(0);

default: printf("Input error!\n");

}

}

return 0;

}

/\* 函数功能：显示菜单并获得用户键盘输入的选项 \*/

int Menu(void)

{

int itemSelected;

printf("Management for Students' scores\n");

printf("1.Input record\n");

printf("2.Caculate total and average score of course\n");

printf("3.Sort in descending order by score\n");

printf("4.Sort in ascending order by score\n");

printf("5.Sort in ascending order by number\n");

printf("6.Sort in dictionary order by name\n");

printf("7.Search by number\n");

printf("8.Search by name\n");

printf("9.Statistic analysis\n");

printf("10.List record\n");

printf("0.Exit\n");

printf("Please Input your choice:\n");

scanf("%d", &itemSelected); /\* 读入用户输入 \*/

return itemSelected;

}

/\* 函数功能：输入 n 个学生的某门课成绩 \*/

void ReadScore(long num[], char name[][MAX\_LEN], float score[], int n)

{

int i;

printf("Input student's ID, name and score:\n");

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

{

scanf("%ld%s%f", &num[i], name[i], &score[i]);

}

}

/\* 函数功能：计算全班总分和平均分 \*/

void AverSumofScore(float score[], int n)

{

int

i;

float sum = 0;

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

{

sum = sum + score[i];

}

printf("sum=%.0f,aver=%.2f\n", sum, n>0 ? sum/n : 0);

}

/\* 函数功能：按选择法将数组 score 的元素值排序 \*/

void SortbyScore(long num[], char name[][MAX\_LEN], float score[], int n,

int (\*compare)(float a, float b))

{

int i, j, k;

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

{

k = i;

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

{

if ((\*compare)(score[j], score[k])) k = j;

}

if (k != i)

{

SwapFloat(&score[k], &score[i]); /\* 交换成绩 \*/

SwapLong(&num[k], &num[i]);

SwapChar(name[k], name[i]);

/\* 交换学号 \*/

/\* 交换姓名 \*/

}

}

}

/\* 使数据按升序排序 \*/

int Ascending(float a, float b)

{

return a < b;

/\* 这样比较决定了按升序排序，如果 a<b，则交换 \*/

}

/\* 使数据按降序排序 \*/

int Descending(float a, float b)

{

return a > b;

/\* 这样比较决定了按降序排序，如果 a>b，则交换 \*/

}

/\* 交换两个单精度浮点型数据 \*/

void SwapFloat(float \*x, float \*y)

{

float temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* 交换两个长整型数据 \*/

void SwapLong(long \*x, long \*y)

{

long temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

/\* 交换两个字符串 \*/

void SwapChar(char x[], char y[])

{

char temp[MAX\_LEN];

strcpy(temp, x);

strcpy(x, y);

strcpy(y, temp);

}

/\* 函数功能：按选择法将数组 num 的元素值按从低到高排序 \*/

void AsSortbyNum(long num[], char name[][MAX\_LEN], float score[], int n)

{

int

i, j, k;

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

{

k = i;

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

{

if (num[j] < num[k]) k = j;

}

if (k != i)

{

SwapFloat(&score[k], &score[i]); /\* 交换成绩 \*/

SwapLong(&num[k], &num[i]);

SwapChar(name[k], name[i]);

/\* 交换学号 \*/

/\* 交换姓名 \*/

}

}

}

/\* 函数功能：交换法实现字符串按字典顺序排序 \*/

void SortbyName(long num[], char name[][MAX\_LEN], float score[], int n)

{

int

i, j;

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

{

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

{

if (strcmp(name[j], name[i]) < 0)

{

SwapFloat(&score[i], &score[j]); /\* 交换成绩 \*/

SwapLong(&num[i], &num[j]);

SwapChar(name[i], name[j]);

/\* 交换学号 \*/

/\* 交换姓名 \*/

}

}

}

}

/\* 函数功能：按学号查找学生成绩并显示查找结果 \*/

void SearchbyNum(long num[], char name[][MAX\_LEN], float score[], int n)

{

long number;

int

i;

printf("Input the number you want to search:\n");

scanf("%ld", &number);

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

{

if (num[i] == number)

{

printf("%ld\t%s\t%.0f\n", num[i], name[i], score[i]);

return;

}

}

printf("Not found!\n");

}

/\* 函数功能：按姓名的字典顺序排出成绩表 \*/

void SearchbyName(long num[], char name[][MAX\_LEN], float score[], int n)

{

char x[MAX\_LEN];

int i;

printf("Input the name you want to search:\n");

scanf("%s", x);

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

{

if (strcmp(name[i], x) == 0)

{

printf("%ld\t%s\t%.0f\n", num[i], name[i], score[i]);

return;

}

}

printf("Not found!\n");

}

/\* 函数功能：统计各分数段的学生人数及所占的百分比 \*/

void StatisticAnalysis(float score[], int n)

{

int i, total, t[6] = {0,0,0,0,0,0};

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

{

if (score[i]>=0 && score[i]<60)

else if (score[i]<70)

t[0]++;

t[1]++;

t[2]++;

t[3]++;

t[4]++;

t[5]++;

else if (score[i]<80)

else if (score[i]<90)

else if (score[i]<100)

else if (score[i] == 100)

}

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

{

total = total + t[i];

}

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

{

if (i == 0) printf("<60\t%d\t%.2f%%\n",t[i],(float)t[i]/n\*100);

else if (i == 5) printf("%d\t%d\t%.2f%%\n", (i+5)\*10,t[i],(float)t[i]/n\*100);

else printf("%d-%d\t%d\t%.2f%%\n", (i+5)\*10, (i+5)\*10+9, t[i], (float)t[i]/n\*100);

}

}

/\* 函数功能： 打印学生成绩 \*/

void PrintScore(long num[], char name[][MAX\_LEN], float score[], int n)

{

int i;

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

{

printf("%ld\t%s\t%.0f\n", num[i], name[i], score[i]);

}

}

**3.** 单词接龙

#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;

}

**4.** 分数比较

#include <stdio.h>

#include <stdlib.h>

int zxgb(int a,int b);

int 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);

}

return 0;

}

int zxgb(int a,int b)

{

long 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);

}