11月7号作业

1、

#include<math.h>

double sabc(double a, double b, double c)

{

double s, p;

p = (a + b + c) / 2;

s = sqrt(p \* (p - a) \* (p - b) \* (p - c));

return(s);

}

2、

#include<stdio.h>

main()

{

int m, n, q;

double a;

double p(int);

printf("请输入正整数m和n：");

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

q = m - n;

a = p(m) / p(q);

printf("A=%lf", a);

}

double p(int x)

{

double s = 1.0;

int i;

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

s \*= i;

return(s);

}

6、

int Fib(int n)

{

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

else return (Fib(n - 1) + Fib(n - 2));

}

7、

int Ack(int n, int x, int y)

{

if (n == 0) return (x + 1);

else if (y == 0)

{

if (n == 1) return (x);

else if (n == 2) return 0;

else if (n == 3) return 1;

else if (n >= 4) return 2;

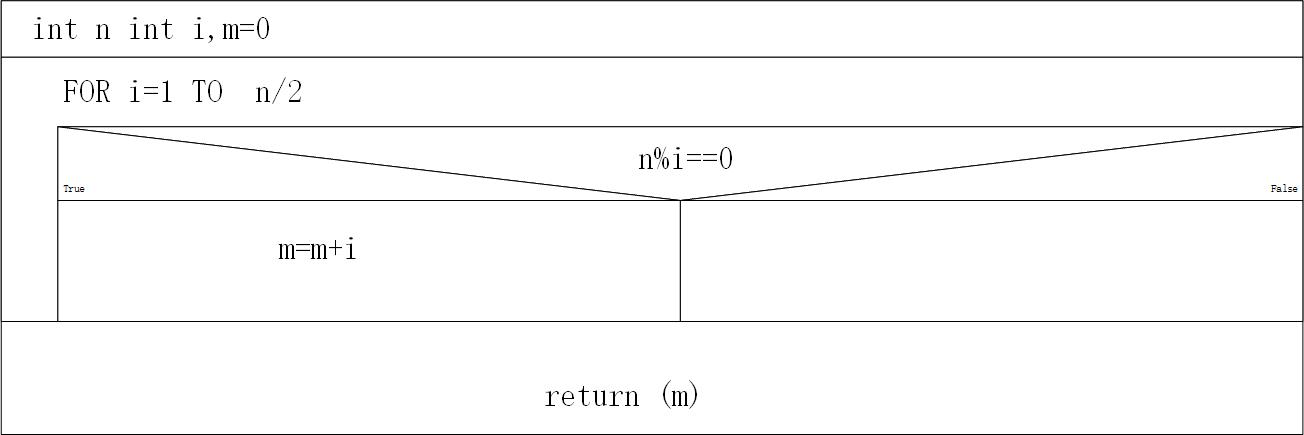
}

else return Ack(n - 1, Ack(n, x, y - 1), x);

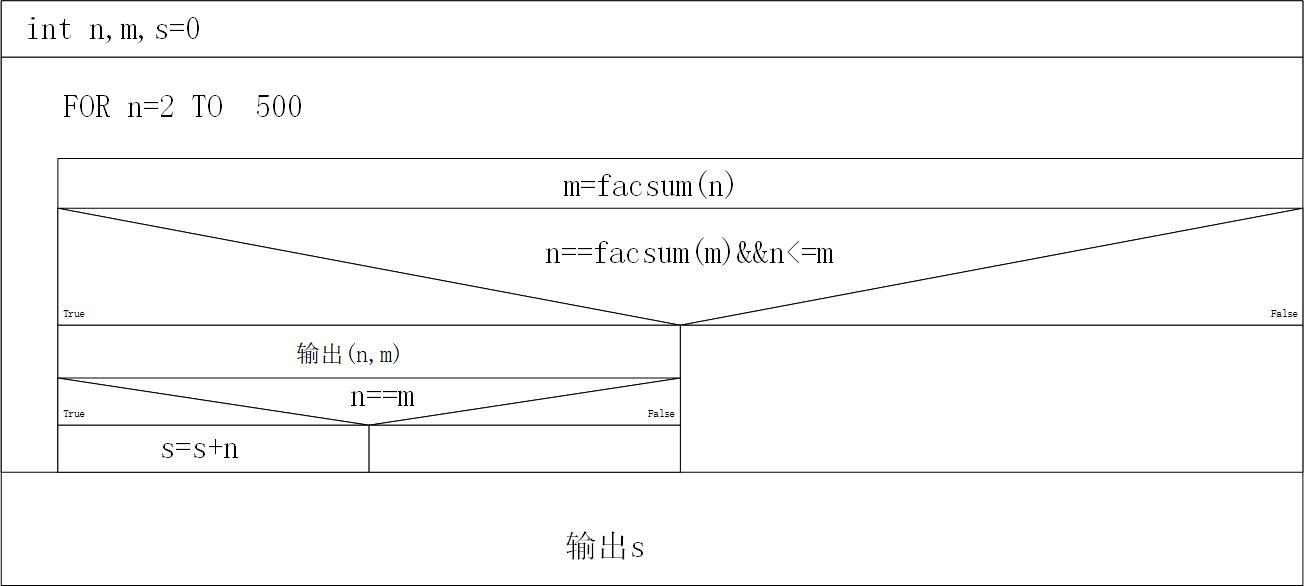
}

10、

facsum函数流程图



main函数流程图



**放在两个不同文件下：**

/\*file1.c\*/

#include<stdio.h>

main()

{

int n, m, s = 0;

int facsum(int);

printf("各对“亲密数”：");

/\*1不属于亲密数，跳过其开始循环\*/

for (n = 2; n <= 500; n++)

{

m = facsum(n);

/\*判断是否符合亲密数条件并且较小数在前\*/

if (n == facsum(m) && n <= m)

{

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

/\*已知是亲密数后再判断是否为完数并做累加\*/

if (n == m)

s += n;

}

}

printf("\n");

printf("“完数”之和：%d\n", s);

}

/\*file2.c\*/

/\*下面编写一个求n所有因子之和的函数\*/

int facsum(int n)

{

int i, m = 0;

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

{

if (n % i == 0) m += i;

}

return (m);

}

**放在同一文件下：**

/\*file1.c\*/

#include<stdio.h>

main()

{

int n, m, s = 0;

int facsum(int);

printf("各对“亲密数”：");

/\*1不属于亲密数，跳过其开始循环\*/

for (n = 2; n <= 500; n++)

{

m = facsum(n);

/\*判断是否符合亲密数条件并且较小数在前\*/

if (n == facsum(m) && n <= m)

{

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

/\*已知是亲密数后再判断是否为完数并做累加\*/

if (n == m)

s += n;

}

}

printf("\n");

printf("“完数”之和：%d\n", s);

}

/\*下面编写一个求n所有因子之和的函数\*/

int facsum(int n)

{

int i, m = 0;

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

{

if (n % i == 0) m += i;

}

return (m);

}

11、

#include<stdio.h>

main()

{

double m0, n0;

long f(long);

printf("请输入m,n：");

scanf("%lf %lf", &m0, &n0);

/\*判断m,n的合理性，包括其为小数的不合理性\*/

if (m0 != (long)m0 || n0 != (long)n0 || n0 < m0 || n0 < 0 || m0 < 0)

printf("输入的m,n错误！");

else

{

long i, m, n, s = 0;

m = (long)m0; n = (long)n0;

if (n != 0) //n==0时不会求和

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

s += f(i);

printf("阶乘累加值为%ld\n", s);

}

}

/\*下面编写计算阶乘的递归函数\*/

long f(long k)

{

if (k < 0)

{

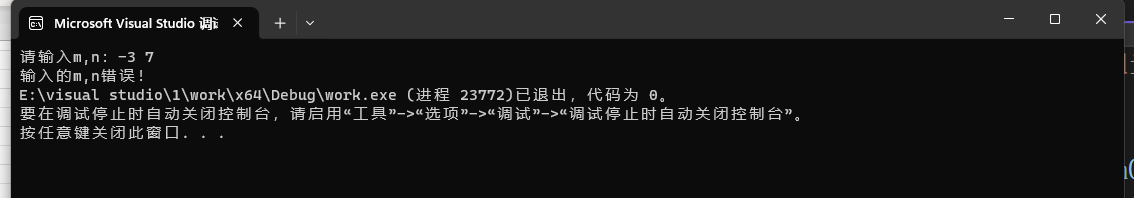
printf("k值小于0不合理，错误！\n");

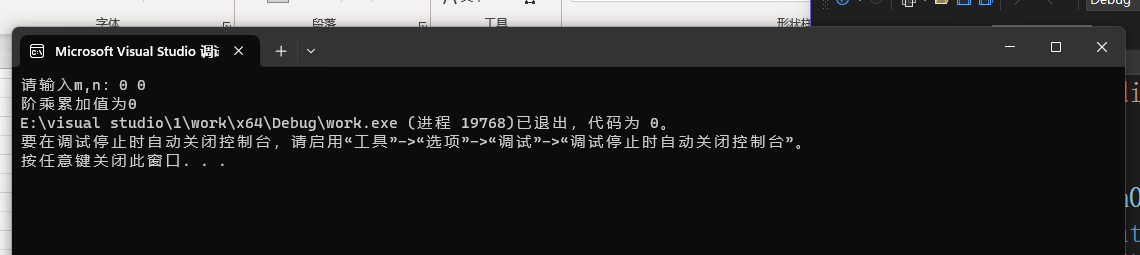
return 0;

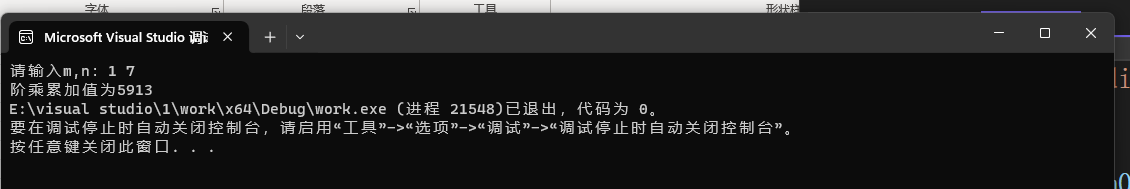
}

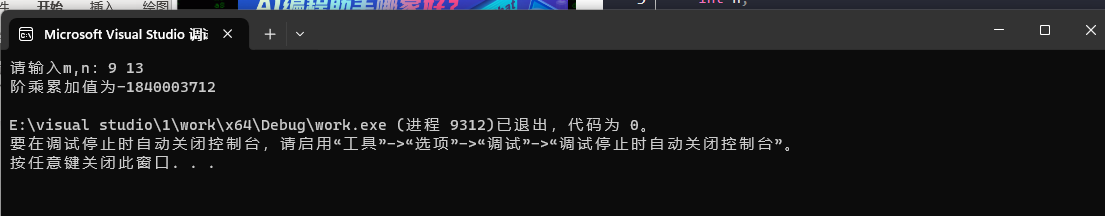
else if (k == 0) return 1;

else return (k \* f(k - 1));

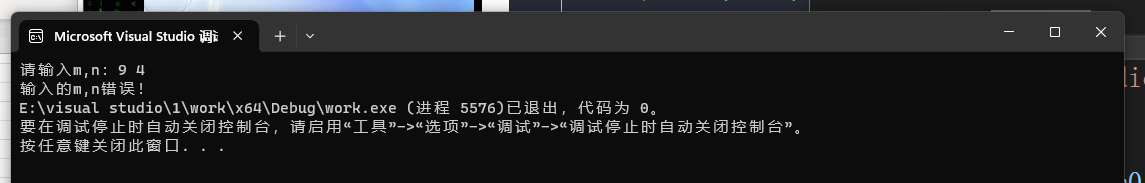
}







（数值溢出）



11月9号作业

1、

#include<stdio.h>

main()

{

int a[10];

int i;

printf("请输入10个整数：");

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

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

/\*下面循环逆序输出元素\*/

printf("逆序输出为：");

for (i = 9; i >= 0; i--)

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

printf("\n");

/\*下面通过比较各元素大小确定最大元素\*/

int max, n;

max = a[0]; n = 0;

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

{

if (a[i] > max)

{

max = a[i];

n = i;

}

}

printf("最大元素的下标是：%d\n", n);

}

4、

#include<stdio.h>

main()

{

int a[4][6];

int i, k;

printf("请输入数组的各元素：");

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

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

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

/\*下面比较各元素大小确定最小元素\*/

int min, m=0, n=0;

min = a[0][0];

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

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

{

if (a[i][k] < min)

{

min = a[i][k];

m = i; n = k;

}

}

printf("最小元素的行下标和列下标是：%d %d\n", m, n);

}

5、

#include<stdio.h>

main()

{

int a[5][5], b[5];

int i, k;

int min(int [][5],int);

printf("请输入数组的各元素：");

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

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

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

/\*下面选出每行中的最小值\*/

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

b[i] = min(a, i);//最小值储存在一维函数中

/\*下面输出各元素和最小值\*/

printf("数组a[5][5]如下：\n");

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

{

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

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

printf(" min=%d\n", b[i]);

}

}

/\*下面编制一个计算一行最小值的函数\*/

int min(int a[5][5], int i)

{

int m = a[i][0];

int p;

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

{

if (a[i][p] < m)

m = a[i][p];

}

return (m);

}

10、

#include<stdio.h>

#include<string.h>

main()

{

char a[51];

int flag=1, i, m = 0, n = 0;

printf("请输入50个字符:");

gets(a);

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

{

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

m++;

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

n++;

else if (a[i] == '\0') flag = 0;

}

printf("英文字母有%d个\n", m);

printf("数字字符有%d个\n", n);

}

12、

#include<stdio.h>

main()

{

int year, month, day, k;

printf("请输入今天是几年几月几日：");

scanf("%d %d %d", &year, &month, &day);

int r(int); int t(int, int, int);

k = t(year, month, day);

if (k % 5 >= 1 && k % 5 <= 3) printf("今天打鱼");

else printf("今天晒网");

}

/\*下面编写一个判断闰年的函数\*/

int r(int year)

{

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

return 1;

else return 0;

}

/\*下面编写一个计算总时间的函数\*/

int t(int year, int month, int day)

{

/\*定义一个存放月数的数组\*/

int p[13] = { 0,31,28,31,30,31,30,31,31,30,31,30,31 };

int y, m, t = 0;

/\*求出指定前一年的总天数\*/

for (y = 2000; y < year; y++)

{

if (r(year))

t += 366;

else

t += 365;

}

/\*判断最后一年是不是闰年来判断二月\*/

if (r(year))

p[2]++;

/\*最后算出总天数\*/

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

t += p[m];

t += day;

return (t);

}