（不能用全局变量）

1、一只青蛙一次可以跳上 1 级台阶，也可以跳上2 级。求该青蛙跳上一个n 级的台阶总共有多少种跳法。

设计要求：

（1）、主函数中输入台阶个数，自定义函数解决计算过程，计算结果在主函数中输出。

（2）、分别使用递归方法和常规方法（即循环结构）完成以上设计。

输入输出样例1：

10

89

输入输出样例2：

20

10946

/\*递归方法\*/

#include <stdio.h>

int f(int n)

{

if (n == 0)

return 0;

else if (n == 1)

return 1;

else if (n == 2)

return 2;

else

return (f(n-1)+f(n-2)); //若最后跳1步到达,则为f(n - 1)种；若最后跳2步到达,则为f(n-2)种

}

int main()

{

int n;////定义台阶数

printf("Input a number:");

scanf("%d", &n); //接收台阶数

if (n < 0)

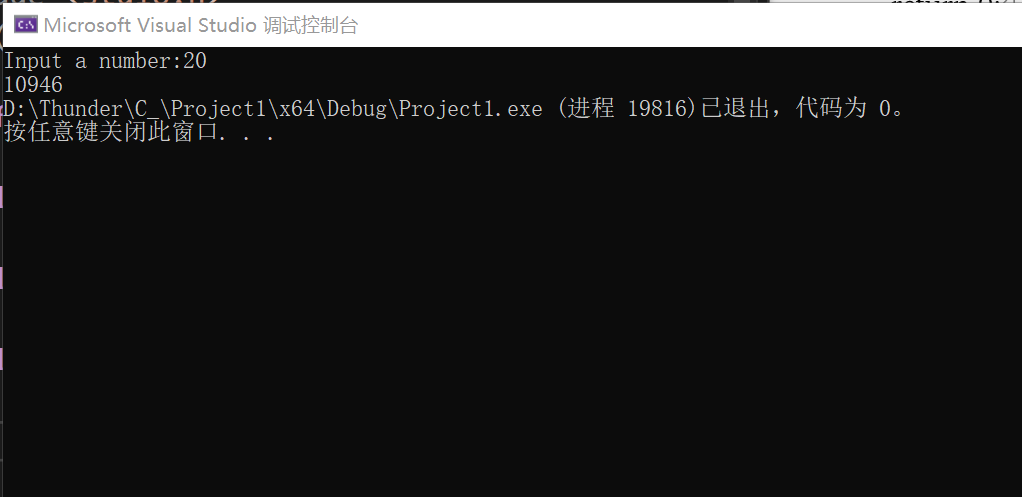
printf("error!\n");

else

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

return 0;

} 



/\*循环结构\*/

#include<stdio.h>

int f(int n)

{

if (n == 0 || n == 1 || n == 2)

return n;

int f1 = 1;

int f2 = 2;

int num = 0;

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

{

num = f1 + f2;

f1 = f2;

f2 = num;

}

return num;

}

int main()

{

int n;////定义台阶数

printf("Input a number:");

scanf("%d", &n); //接收台阶数

if (n < 0)

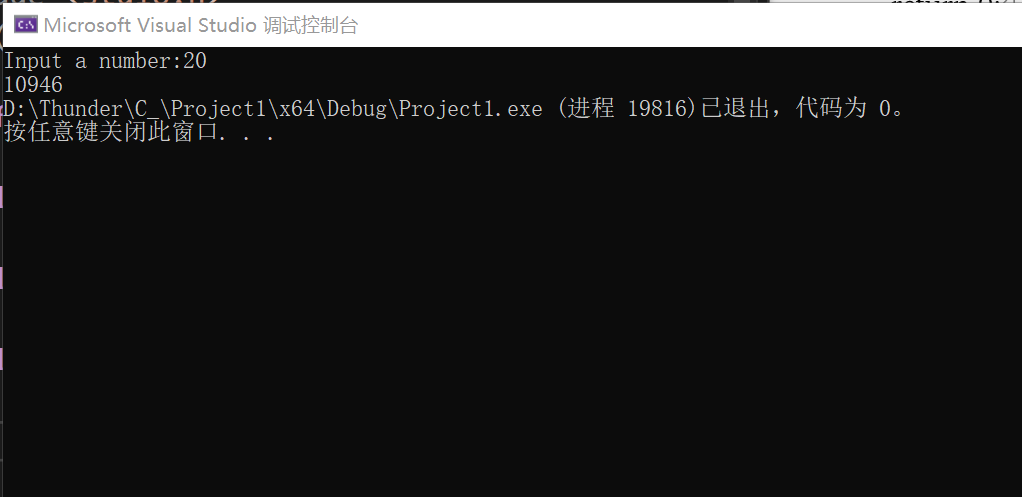
printf("error!\n");

else

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

return 0;

}



2、 自定义一个函数，找出一行英文句子中最长的单词出现的开始位置和最长单词的字符个数，英文句子的输入和最终输出结果要求在主函数中实现。

输入样例：各单词间保留一个空格

I am a student

输出样例：

说明：输出结果表示最长单词student出现在第8个字符的位置，该单词包含7个字符

8  7

#include<stdio.h>

#include<string.h>

void f(char\* s,int \*a)

{

int i = 0;

int n = 0;//记录单词字母个数

int max = 0;//记录最长单词字母个数

for (i = 0; i < 100; i++)

{

if (s[i] == ' ')

{

if (n > max)

max = n;

n = 0;

}

else if (s[i] == '\0')

break;

else

{

n++;

if (n > max)

max = n;

}

}

a[0] = i - max + 1 ;//最长单词的位置

a[1] = max;//最长单词的字母个数

}

int main()

{

char str[100]="\0";

int a[2]={0};//用于输出单词位置和字母个数

gets(str);//接收字符串

f(str, a);

printf("%d %d",a[0],a[1]);

return 0;

}

