# 4编译预处理实验

## 4.1实验目的

掌握文件包含、宏定义、条件编译、assert宏的使用；练习带参数的宏定义、条件编译的使用；学会使用assert；能够使用Turbo C 2.0集成开发环境中的调试功能：单步执行、设置断点、观察变量值。

## 4.2 必做题

### 4.2.1 源程序改错

**【题目】**

下面是用宏来计算平方差、交换两数的源程序，在这个源程序中存在若干语法和逻辑错误。要求在计算机上对这个例子程序进行调试修改，使之能够正确完成指定任务。

1 #include "stdio.h"

2 #define SUM a+b

3 #define DIF a-b

4 #define SWAP(a,b) a=b,b=a

5 void main

6 {

7 int b, t;

8 printf("Input two integers a, b:");

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

10 printf("\nSUM=%d\n the difference between square of a and square of b is:%d",SUM, SUM\*DIF);

11 SWAP(a,b);

12 Printf("\nNow a=%d,b=%d\n",a,b);

13 }

【**错误原因分析及改错方案**】

经过调试和运行，发现这个程序一共存在三处错误,分析如下。

第11行错误，printf函数首字母大写；正确形式应为将printf函数改为全部小写。

第2行错误，第一个#define SUM a+b，a，b未加括号，导致宏展开时出现错误；正确形式应该为#define SUM ((a)+(b))。

第3行错误，第二个#define DIF a-b，a，b未加括号，导致宏展开时出现错误；正确形式应该为#define DIF ((a)-(b))。

第4行错误，SWAP宏定义不能达到交换的效果；正确形式应该为#define SWAP(a,b) t=a,a=b,b=t。

源程序未定义变量a，故需要在之前定义变量a，即改为int a, b, t。

**【修改后代码】**

#include <stdio.h>

#define SUM (a+b)

#define DIF (a-b)

#define SWAP(a,b) t=a,a=b,b=t

int main()

{

int a, b, t;

printf("Input two integers a, b:");

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

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

printf("\nSUM=%d\n the difference between square of a and square of b is:%d",SUM, SUM\*DIF);

SWAP(a,b);

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

return 0;

}

【**测试数据、测试理论结果及测试结果】**

表4-1实验四必做题4.2.1测试结果

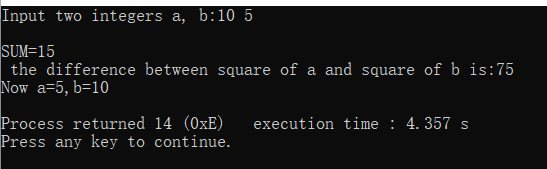


图4-1 实验四必做题4.2.1样例测试结果

### 4.2.2 源程序修改替换

**【题目】**

下面是用函数实现求三个数中最大数、计算两数之和的源程序。在这个源程序中存在若干语法和逻辑错误,要求：

(1) 对这个例子程序进行调试修改，使之能够正确完成指定任务。

(2) 用带参数的宏替换函数max，来实现求最大数的功能。

#include<stdio.h>

int main(void)

{

int a, b, c;

float d, e;

printf("Enter three integers:");

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

printf("\nthe maximum of them is %d\n", max(a, b, c));

printf("Enter two floating point numbers:");

scanf("%f,%f", &d, &e);

printf("\nthe sum of them is %f\n", sum(d, e));

return 0;

}

int max(int x, int y, int z)

{

int t;

if (x>y)

t = x;

else

t = y;

if (t<z)

t = z;

return t;

}

float sum(float x, float y)

{

return x + y;

}

**【错误分析】**

原程序算法是输入三个数并判断最大数之后输出，之后再输入两个数，求两个数之和输出，都是用函数形式来实现。首先，其缺少头文件声明#include<stdio.h>,里面包含了printf()与scanf()函数的声明和定义。 其次，两个自定义函数没有声明，会导致编译器报错。再来因为主函数main已经声明返回值为空，则不需要后面的返回值零，可以去掉return 0;或者将void main()改为int main()。

**【修改替换后代码】**

#include<stdio.h>

#define max(x,y,z) x>y&&x>z?x:y>z&&y>x?y:z

float sum(float x, float y);

int main(void)

{

int a, b, c;

float d, e;

printf("Enter three integers:");

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

printf("\nthe maximum of them is %d\n", max(a, b, c));

printf("Enter two floating point numbers:");

scanf("%f,%f", &d, &e);

printf("\nthe sum of them is %f\n", sum(d, e));

return 0;

}

float sum(float x, float y)

{

return x + y;

}

**【测试数据、测试理论结果及测试结果】**

表4-2 实验四必做题4.2.2 测试结果

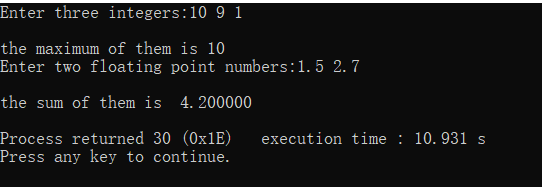


图4-4 实验四必做题4.2.2样例测试结果

### 4.2.3 跟踪调试

**【题目】**

下面程序利用R计算圆的面积s，以及面积s的整数部分。

1）修改程序，使程序编译通过且能运行；

2）单步执行。进入函数decimal\_fraction时watch窗口中x为何值？在返回main时, watch窗口中i为何值？

3）排除错误，使程序能正确输出面积s值的整数部分，不会输出错误信息assertion failed。

#define R

void main(void)

{

float r, s;

int s\_integer=0;

printf ("input a number: ");

scanf("%f",&r);

#ifdef R

s=3.14159\*r\*r;

printf("area of round is: %f\n",s);

s\_integer= integer\_fraction(s);

printf("the integer fraction of area is %d\n", s\_integer);

assert((s-s\_integer)<1.0);

#endif

}

int integer\_fraction(float x)

{

int i=x;

return i;

}

**【错误分析】**

缺少头文件#include<stdio.h>和#include<assert.h>,会导致printf()函数,scanf()函数和assert()函数没有声明和定义，导致错误。

**【修改后代码】**

#include<stdio.h>

#include<assert.h>

int integer\_fraction(float x);

#define R

void main(void)

{

float r, s;

int s\_integer = 0;

printf("input a number: ");

scanf\_s("%f", &r);

#ifdef R

s = 3.14159\*r\*r;

printf("area of round is: %f\n", s);

s\_integer = integer\_fraction(s);

printf("the integer fraction of area is %d\n", s\_integer);

assert((s - s\_integer)<1.0);

#endif

}

int integer\_fraction(float x)

{

int i = x;

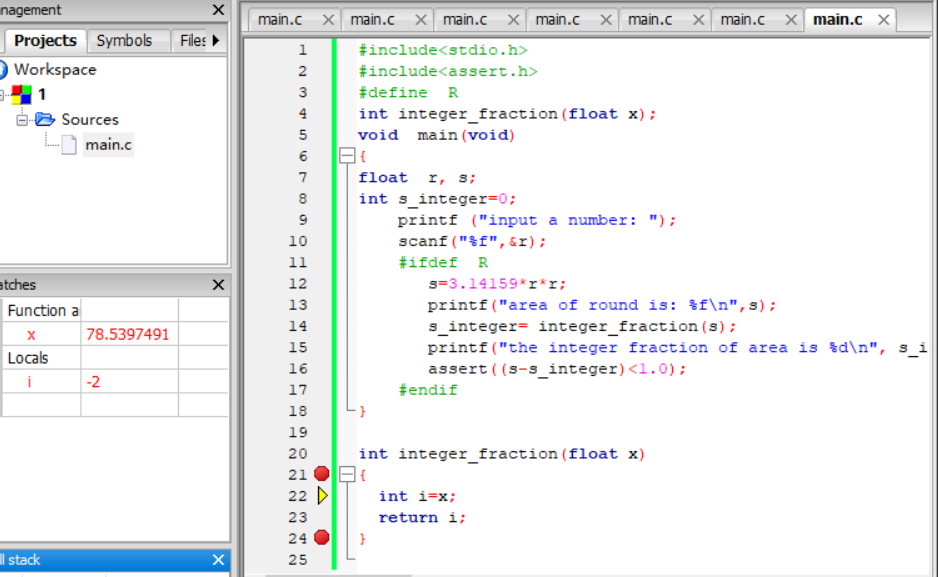
return i;

}

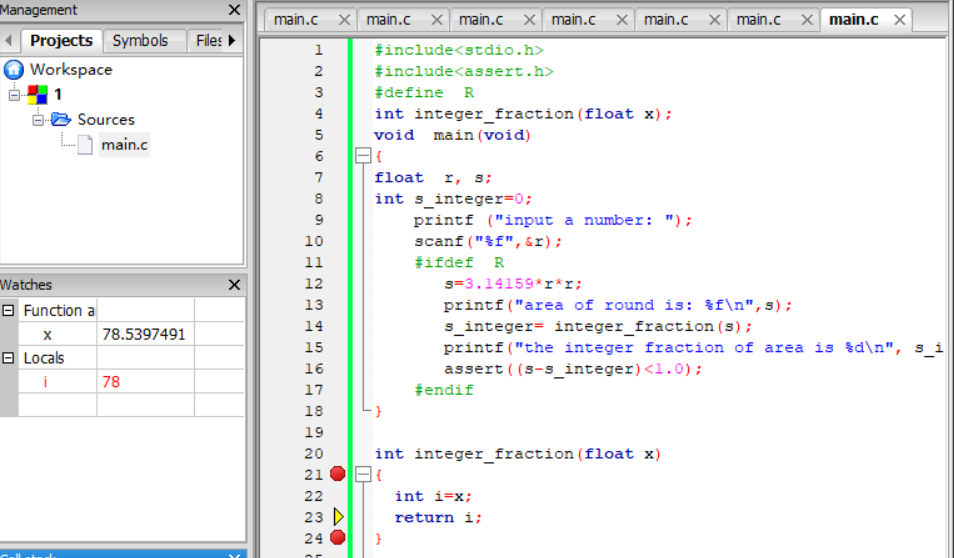
**【调试过程】**

可以直接在函数decimal\_fraction内部设置设置两个断点，分别是刚进入函数decimal\_fraction和返回主函数处，刚进入函数时命中断点，如图4-7，根据局部变量窗口，此时x的值为78.5397491。在返回main时，根据局部变量窗口，i的值变为78，如图4-8

图4-7 实验四必做题4.2.3调试过程



**4.7**



**4.8**

图4-8 实验四必做题4.2.3调试过程

### 4.2.4 计算三角形面积

**【题目】**

已知三角形的面积是，其中，a,b,c为三角形的三边。定义两个带参数的宏，一个用来求s，另一个用来求area,试编写一程序，用带参数的宏来计算三角形的面积。

**【算法流程图】**

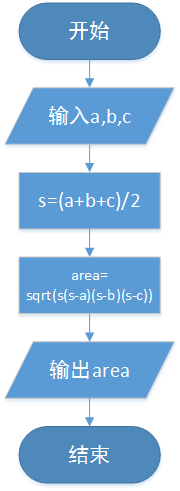


图4-9 实验四必做题4.2.4 算法流程图

**【程序清单】**

#include<stdio.h>

#include<math.h>

#define sum(a,b,c) (a+b+c)/2.0

#define area(a,b,c) sqrt(s\*(s-a)\*(s-b)\*(s-c))

int main()

{

float a,b,c,s,area;

printf("Please input three integer a,b,c:");

scanf("%f,%f,%f",&a,&b,&c);

s=sum(a,b,c);

area=area(a,b,c);

printf("%f",area);

return 0;

}

**【测试数据、测试理论结果及测试结果】**

表4-3 实验四必做题4.2.4测试结果

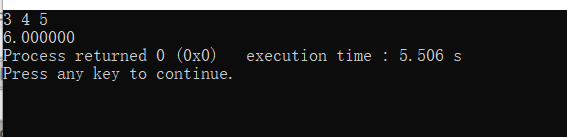


图4-10 实验四必做题4.2.4样例测试结果

### 4.2.5 条件编译电报

**【题目】**

使用条件编译方法编写一程序,其功能要求是,输入一行电报文字，可以任选两种输出：一为原文输出；二为变换字母的大小写（如小写‘a’变成大写‘A’，大写‘D’变成小写‘d’），其他字符不变。用#define命令控制是否变换字母的大小写。例如，#define CHANGE 1 则输出变换后的文字，若#define CHANGE 0则原文输出。

**【算法流程图】**

该程序采用单个输入字符遇到\n停止的方法，当define CHANGE不等于0的时候，执行if语句，由于大小写字母在内存中仅第五位相反，所以利用异或转换将大小写字母进行相互转化。程序框图如图

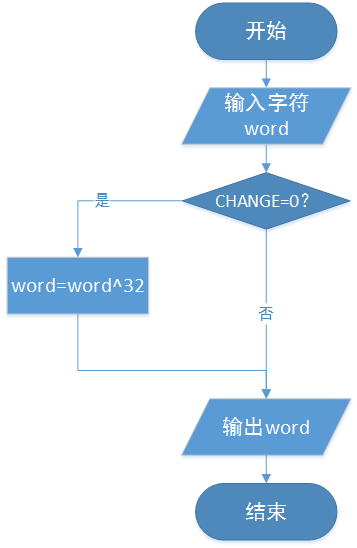


图4-13 实验四必做题4.2.5算法流程图

**【程序清单】**

#include<stdio.h>

#define CHANGE 1

int main()

{

int Word;

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

{

#if CHANGE

printf("%c", Word ^ 32);

#else

printf("%c", Word);

#endif

}

printf("\n");

return 0;

}

**【测试数据、测试理论结果】**

本程序分为两种情况：CHANGE为0和1。根据这两种情况各设计一组测试数据如下：

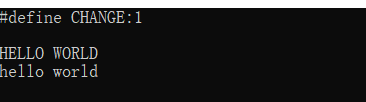
1、当定义CHANGE为0时，输入HelloWorld，应输出原文HelloWorld；测试结果如图4-14。

2、当应以CHANGE为1时，输入helloworld，应输出大小写相互转化后的结果HELLOWORLD；测试结果如图4-15。

**【测试结果】**



图4-14 实验四必做题4.2.5 样例1测试结果



**图4-15 实验四必做题4.2.5 样例2测试结果**

## 4.3小结

通过此次实验，我进一步的了解了预编译的原理和作用，以及头文件的使用。还有关于带参宏定义的使用。同时发现宏定义表达式带上括号的重要性，这样可以避免优先级不同产生一些逻辑错误。在程序修改替换实验中，学会了用宏定义代替某些简单函数的方法。跟踪调试中，对静态断言assert函数的了解有了进一步加深，也对数据的强制转换类型进行了深刻的理解。