# 实验4 编译预处理实验

4.1、实验目的

（1）掌握文件包含、宏定义、条件编译、assert宏的使用；

（2）练习带参数的宏定义、条件编译的使用；

（3）练习assert宏的使用；

（4）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、实验内容

**4.2.1．源程序改错题**

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);

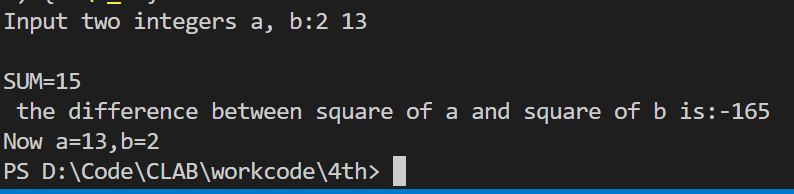
12 Printf("\nNow a=%d,b=%d\n",a,b);

13 }

**解答：**

1. 错误修改：
2. 将宏定义中后面的表达式加上括号，以防止在第10行中出现优先级混乱的问题导致错误。
3. 第四行，修改为#define SWAP(a,b,t) t=a,a=b,b=t。
4. 第五行，修改为int main(void)。
5. 第九行，去掉两%d之间的逗号。
6. 第十一行改为SWAP(a,b,t);

（2）错误修改后的运行和结果：



**图4-1 改错题错误修改后的运行结果**

**4.2.2．源程序修改替换题**

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

1 void main(void)

2 {

3 int a, b, c;

4 float d, e;

5 printf("Enter three integers:");

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

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

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

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

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

11 return 0;

12 }

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

14{

15 int t;

16 if (x>y)

17 t=x;

18 else

19 t=y;

20 if (t<z)

21 t=z;

22 return t;

23 }

24 float sum(float x, float y)

25 {

26 return x+y;

27 }

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。

修改后代码：

#include <stdio.h>

#define MAXC(a, b, c) (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c)

int max(int x, int y, int 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;

}

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;

}

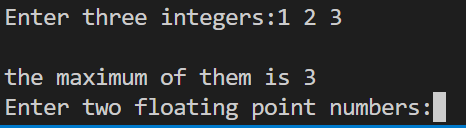
float sum(float x, float y)

{

return x + y;

}

运行结果：



**图4-2 源程序修改替换后的运行结果**

1. 用带参数的宏替换函数max，来实现求最大数的功能。

修改后代码：

#include <stdio.h>

#define MAXC(a, b, c) (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c)

int max(int x, int y, int 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", MAXC(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;

}

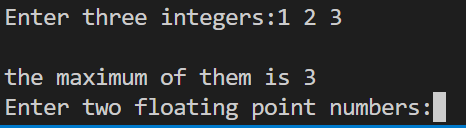
float sum(float x, float y)

{

return x + y;

}

运行结果：



**图4-3 源程序用宏定义修改替换后的运行结果**

**4.2.3．跟踪调试题**

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 #define R

2 void main(void)

3 {

4 float r, s;

5 int s\_integer=0;

6 printf ("input a number: ");

7 scanf("%f",&r);

8 #ifdef R

9 s=3.14159\*r\*r;

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

11 s\_integer= integer\_fraction(s);

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

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

14 #endif

15 }

16 int integer\_fraction(float x)

17 {

18 int i=x;

19 return i;

20 }

1. 修改程序，使程序编译通过且能运行；

修改后代码：

#include<stdio.h>

#include<assert.h>

#define R 1

int integer\_fraction(float x);

int 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

return 0;

}

int integer\_fraction(float x)

{

int i = x;

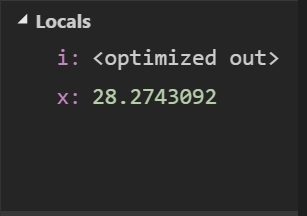
return i;

}

1. 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？

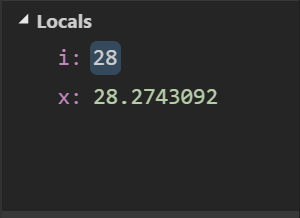
当输入值为3时：

进入函数decimal\_fraction时watch窗口中x为：



**图4-4 跟踪调试测试结果（1）**

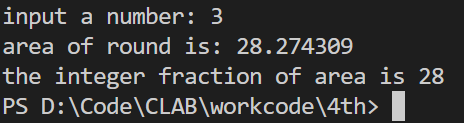
在返回main时, watch窗口中i为：



**图4-5 跟踪调试测试结果（2）**

1. 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。

运行结果：



**图4-6 跟踪调试运行结果**

**4.2.4．编程设计题**

**（1）**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。编写程序，用带参数的宏来计算三角形的面积。

**解答：**

1. 解题思路：定义两个带参数宏S，AREA。进行计算

2）程序清单

#include<stdio.h>

#include<math.h>

#define S(a, b, c) (((a)+(b)+(c))/2)

#define AREA(S, a, b, c) sqrt((S)\*((S)-a)\*((S)-b)\*((S)-c))

int main(void){

double s, a, b, c, area;

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

s = S(a, b, c);

area = AREA(s, a, b, c);

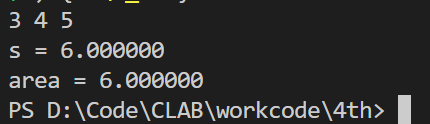
printf("s = %lf\narea = %lf", s, area);

return 0;

}

3）测试

对应测试的运行结果如图4-8所示。

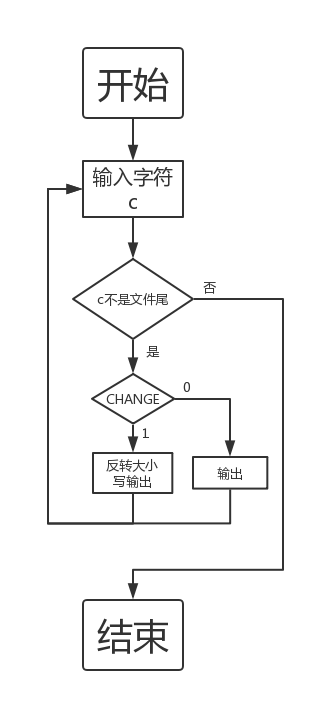


**图4-8 编程题1的测试运行结果**

**（2）**用条件编译方法来编写程序。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。

**解答：**

1) 解题思路：



**图4-9 编程题2的算法流程图**

2）程序清单

#include<stdio.h>

#define CHANGE 1

int main(void){

char c;

while((c = getchar()) != EOF){

#if CHANGE

if(c >= 'A' && c <= 'Z')

printf("%c", c + 32);

else if(c >= 'a' && c <= 'z')

printf("%c", c - 32);

else

putchar(c);

#else

printf("%c", c);

#endif

}

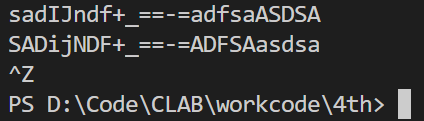
return 0;

}

3）测试

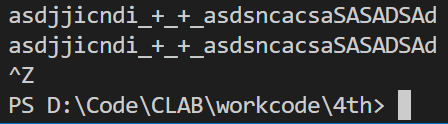
对应测试的运行结果如图4-10、4-11所示。

**#define CHANGE 1：**



**图4-10 编程题2的测试运行结果（1）**

**#define CHANGE 0：**



**图4-11 编程题2的测试运行结果（2）**

## 4.3 实验小结

第四次C语言上机实验，主要掌握并熟悉了宏定义和条件编译的使用并练习了assert宏的使用，同时温故了上节课学习的调试并学会了单步执行、设置断点、观察变量值等方法。

**（1）改错实验**

改错题首先继续让我认识到了宏定义的不当使用，可能会造成一些奇怪的错误。这些错误进一步告诉我们，在使用宏定义这一技术时，应注意到的一系列问题。加深了我对预处理器与编译器之间的关系，也领会到了宏定义通过预处理器进行替换的方式，受益匪浅。而且，通过比对带参数宏以及函数的差别，认识到了二者在运行效率以及所需程序运行空间上的差别，大致上学会了如何选择是使用带参数宏还是函数来完成任务。

1. **程序修改替换实验**

在此题中继续领会到了带参数宏的使用上的便捷与方便，与需要更多行的函数相比较更加省时省力。但同时，需要保证在宏定义中所使用的表达式的正确性。

1. **跟踪调试题**

实际体会到了条件编译的使用方法，在实际的工程开发中，条件编译常用于避免头文件重复，或使得代码在不同环境下都可以正常编译，有更好的跨平台性，对于初学者来说可学习最基本的条件编译。同时，复习了使用调试器的方法。

**（4）程序设计实验**

在三角形面积的题目中，我将题目所需要的两个函数（半周长和三角形面积）都用宏定义的形式构造。仅仅两行简单的代码，就成功替换了多行的函数。同时，助教检查时提醒我，以后最好注意如果输入的值是负数等不合理的情况，要返回给用户其输入错误的信息。这样的良好习惯，在我们以后工作时开发软件，可以给用户带来更好的体验。

在电文转换的实验题中，按照题目要求对其进行了编写。使用条件编译，可以在类似情况下，省去对暂时不需要的代码进行编译的过程，节省编译时间与程序占用空间，同时提高效率。