**TEST – 12 编译预处理与带参数的主函数**

**测试 - 试题/答案**

【测试时间90分，自行批阅，不计分】

**学习和解题要点**

1. C程序中，以“#”符号开头的命令是在源程序正式编译前进行处理的，称为“编译预处理”命令。注意，这里是“命令”，而不是语句，因此最后不能加分号。一行只能写一个预处理命令行。C程序中，“宏名”一般用大写，并且中间不能有空格。
2. “宏”定义的作用是当程序中出现“宏名”时，就在“宏名”的位置用对应的“一串符号”替换。注意，这里纯粹就是“替换”，相当于用“一串符号”去“涂改” “宏名”。如:

PI

3.1416

#define PI 3.1416 程序中有 a=r\*r\* ; 涂改为 a=r\*r\* ;

1. 带参的“宏”定义也纯粹就是替换。进行宏替换时，先用实参替换“一串符号”中的形参，然后再用被替换过的“一串符号”替换“宏名(形参表)”。
2. 程序中，用双引号括起来的宏不被替换。
3. 对带参的“宏”定义，在替换时不要人为的加“括号”。如：

#define W(r) 3.14\*r\*r 当a=W(a+b); 正确的替换是 a=3.14\*a+b\*a+b; 而不是a=3.14\*(a+b)\*(a+b) 这两个括号是人为加的。当然，如果我们是想求半径为a+b的面积时就发生了错误，这时应该把“宏”定义写成 #define W(r) 3.14\*(r)\*(r) 因此，为了在调用过程中不出现二义性，应该把字符序列中出现的虚参全部用括号括起来。

6．文件包含编译预处理命令有两种格式：”包含文件名” 和 <包含文件名>。前者，系统先在本程序文件所在的磁盘和路径下寻找包含文件；若找不到，再按系统规定的路径搜索包含文件。后者，系统仅按系统规定的路径搜索包含文件。

7．“包含文件”可以是任意的文件名扩展名，可以是 .h 也可以是 .c ，还可以是其他扩展名。文件名前还可以加上盘号和目录，如：#include “d:\user\file1.c”

8．带参数的主函数是用于在用命令行的形式运行 .exe程序文件时接收参数的。argc接收到包括程序文件名在内的字符串个数，argv的每一个指针指向一个参数字符串。其中argv[0]指向程序文件名。

9．预处理命令若有变动，必须对程序重新进行编译和连接。

**练习题**

1. 单项选择题
2. C程序中，以#号带头的行是预编译（ ）

A．命令 B. 语句 C. 函数 D. 字符串

2．下列正确的预编译命令是（ ）。

A．define PI 3.14159 B. #define P(a,b) strcpy(a,b)

C. #include stdio.h D. #define PI 3.14159;

3．下列命令或语句中，正确的是（ ）。

A. #define MYNAME=”ABC” B．#include stdio.h

C．for(i=0;i<10;i++); D．struct int stu{int name};

4．下列命令或语句中，正确的是（ ）。

A．#define PI=3.14159 B．include “stdio.h”

C．for(i=0,i<10,i++)a++; D．static struct {int i;} b={2};

5．下列命令或语句中，错误的是（ ）。

A．#define PI 3.14159 B．#include <math.h>;

C．if(2); D．for(;;) if(1) break;

6. 设有宏定义“ #define AREA(a,b) a\*b ”，则正确的“宏调用”是（ ）。

A．s=AREA(r\*r) B. s=AREA(x\*y) C. s=AREA D.s=c\*AREA((x+3.5),(y+4.1))

7．设有以下宏定义，则执行语句“z=2\*(N+Y(5+1));”后，z的值为（ ）。

#define N 3

#define Y(n) ((N+1)\*n)

A．出错 B. 42 C. 48 D. 54

8. 设有以下宏定义，int x,m=5,n=1时，执行语句“IFABC(m+n,m,x);”后，x的值为（ ）。

#define IFABC(a,b,c) c=a>b?a:b

A．5 B. 6 C. 11 D. 出错

9. 以下程序中的for循环执行的次数是（ ）

#include “stdio.h”

#define N 2

#define M N+1

#define NUM (M+1)\*M/2

main()

{ int i,n=0;

for ( i=1;i<=NUM;i++ ) { n++; printf(“%d”,n); }

printf(“\n”);

}

A. 5 B. 6 C. 8 D. 9

10. 设有一程序编译连接成目标程序名为ps1.exe，如果输入如下的命令：

ps1 123 456 AAA ↙(其中的↙表示回车)，则程序运行时，会出错的语句是（ ）。

A. printf(“%d”,argc); B. printf(“%s”,argv[4]);

C. printf(“%s”,argv[3]); D. printf(“%s”,argv[2]);

11．格式1 #include “文件名”

格式2 #include <文件名>

下面4个结论，正确的是（ ）。

1. 格式1中的文件名能带路径，而格式2不能
2. 格式1中的文件名不能带路径，而格式2可以
3. 如果被包含文件在当前目录下，两种格式都能用
4. 如果文件名中不带路径，格式1能搜索当前目录和系统规定的路径中搜索包含文件，而格式2不行。

12．程序ccw1.c中有函数max1()，程序ccw2.c中有调用函数max1()的语句，则当程序ccw.c中有包含命令时，正确的写法是（ ）。

1. #include “ccw2.c” B. #include <ccw1.c>

#include “ccw1.c” #include <ccw2.c>

C. #include “ccw1.c” D. 两个包含命令次序可以任意

#include “ccw2.c”

二、填充题

1. C程序中，以“#”符号开头的命令是在源程序正式 前进行 的，称为

命令。

1. 使用分别编译方法时，在不同的编译单位间能传递的变量和函数是 。

3．设有定义如下：#include F(N) 2\*N ，则表达式f(2+3)的值是 。

4．设有如下的宏定义：#include P(r,a) printf(r,a) 当程序中执行“P(str,3);”语句后会输出 i=3 ,则在程序中应有定义语句 char \*str= ;

5．设有下列程序：

#include <stdio.h>

main(int argc, char \*argv[]))

{ while (--argc>c) printf(“%s”,srgv[argc]);

printf(“\n”);

}

假定上述程序编译连接成目标程序名为p.exe，如果输入如下的命令：

p 123 AAA BBB ↙ (其中的↙表示回车)

则输出结果是 。

三、程序分析题

1. 阅读下面程序，写出运行结果。

#define M 5

#define N M\*3+4

#define MN N\*M

main()

{ printf(“%d, %d\n”, 2\*MN, MN/2);

}

1. 阅读下面程序，写出运行结果。

#define EXCH(a,b) { int t; t=a;a=b;b=t;}

main()

{ int x=5,y=9;

EXCH(x,y)

printf(“x=%d, y=%d\n”,x,y);

}

1. 阅读下面程序，写出运行结果。

#define PR(ar) printf(“%d,”,ar)

main()

{ int j, a[]={1,3,5,7,9,11,13,15}, \*p=a+5;

for(j=3; j; j--)

switch(j)

{ case 1:

case 2: PR(\*p++); break;

case 3: PR(\*(--p));

}

}

1. 设下面的程序经编译连接后生成的目标程序名为：ABC.EXE，写出输入以下命令行后的输出：ABC FILE1 FILE2 ↙ (其中的↙表示回车)

#include “stdio.h”

main(int argc, char \*argv[])

{ int i;

printf(“%d\n”,argc);

for( (i=0;i<argc;i++) printf(“%s”,argv[i]);

printf(“\n”);

}

1. 读程序，写功能。

#include <stdio.h>

main(int argc, char \*argv[])

{ FILE \*f1, \*f2;

char ch;

if (argc<3)

{ printf(“Parameter missing!\n”); exit(0); }

if (((f1=fopen(argv[1],”r”))==NULL ||(f2=fopen(argv[2],”w”))==NULL)

{ printf(“Can not open file!\n”); exit(0); }

while (!feof(f1)) fputc(fgetc(f1),f2);

fclose(f1); fclose(f2);

}

四、程序设计题

area= √s(s-a)(s-b)(s-c)

1. 三角形的面积为

其中 s=0.5(a+b+c)，a、b、c为三角形的三边。定义两个带参的宏，一个用来求s，另一个用来求 area。写程序，在程序中用宏来求三角形的周长和面积。

1. 设计一个复制文本文件的程序，源文件名和目标文件名随程序执行时输入。
2. 设计一个程序，将随程序执行时输入的两个整数的和、差、积、整除商、模运算的结果输出。
3. 设计一个程序，能将随程序执行时输入的任意个整数求和。
4. 设计一个比较字符串的程序，两个字符串随程序执行时输入，输出其中大的字符串。

**【答案】**

**编译预处理与带参数的主函数**

**练习题答案**

1. 单项选择题

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| A | B | C | D | C | D | C | A | C | B |
| 11 | 12 |  |  |  |  |  |  |  |  |
| D | C |  |  |  |  |  |  |  |  |

二、填充题

1. 编译 处理 “编译预处理”

2. 非静态存储类型变量和外部函数

3. 7

4. i=”%d\n”

5. BBB AAA 123 命令行中有4个字符串(包括p),因此,argc=4, argv[0]→p ,

argv[1]→123 , argv[2]→AAA , argv[3]→BBB。

三、程序分析题

1. 程序运行结果为： 50 25

2. 程序运行结果为： x=9, y=5

3. 程序运行结果为：9 9 11

4. 程序输出：3

ABC FILE1 FILE2

5. 该程序的功能是：在用命令行的形式运行本程序时，必须在程序名后加二个参数，这二个参数分别表示磁盘上的二个文件名。程序运行后，能把第一个文件的内容拷贝到第二个文件。如果命令行缺少参数或文件打不开，程序终止运行并显示出错情况。

四、程序设计题

1. 解

#include <stdio.h>

#include <math.h>

#define S(a,b,c) 0.5\*(a+b+c)

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

main()

{ float a,b,c;

printf(“输入三角形的三条边长：a,b,c\n”);

scanf(“%f,%f,%f”,&a,&b,&c);

if((a+b>c)&&(b+c>a)&&(c+a>b))

{ printf(“周长=%f\n”, 2\*S(a,b,c));

printf(“面积=%f\n”, AREA(a,b,c));

}

else printf(“a,b,c 的长度不能构成三角形\n”);

}

2. 解

#include <stdio.h>

main(int argc, char \*argv[])

{ FILE \*f1, \*f2;

char ch;

if (argc<3)

{ printf(“Error! Useage: program\_name source\_file\_name object\_file\_name \n”); exit(0); }

if ((f1=fopen(argv[1],”r”))==NULL)

{ printf(“Can not open source file!\n”); exit(0); }

if ((f2=fopen(argv[2],”w”))==NULL)

{ printf(“Can not open object file!\n”); exit(0); }

while (!feof(f1)) fputc(fgetc(f1),f2);

fclose(f1); fclose(f2);

}

3. 解

#include <stdio.h>

#include <stdlib.h>

main(int argc, char \*argv[])

{ int a,b;

if (argc<3) { printf(“Parameter missing!\n”); exit(0); }

a=atoi(argv[1]);

b=atoi(argv[2]); //在stdlib.h中有库函数atoi，把字符串转换成整数

printf(“%5d + %5d = %5d\n”,a,b,a+b);

printf(“%5d - %5d = %5d\n”,a,b,a-b);

printf(“%5d \* %5d = %5d\n”,a,b,a\*b);

printf(“%5d / %5d = %5d\n”,a,b,a/b);

printf(“%5d %% %5d = %5d\n”,a,b,a%b);

}

4. 解

#include <stdio.h>

#include <stdlib.h>

main(int argc, char \*argv[])

{ int i, sum=0;

for(i=1;i<argc;i++) sum=sum+atoi(argv[i]);

printf(“和 = %d\n”,sum);

}

5. 解

#include <stdio.h>

#include <string.h>

main(int argc, char \*argv[])

{ if (argc<3)

{ printf(“Parameter missing!\n”); exit(0); }

if( strcmp(argv[1],argv[2])>0) puts(argv[1]);

else puts(argv[2]);

}