《C语言程序设计》复习题

**第1章 C语言程序设计概述**

**一、选择题**

**1．下列关于计算机语言的叙述中，正确的是\_\_\_C\_\_\_\_。**

A．在计算机语言中，只有机器语言属于低级语言

B．高级语言的源程序可以被计算机直接执行

C．C语言属于高级语言

D．机器语言是与所有机器无关的语言

**2．源程序TEST.C经编译产生的目标文件和连接后产生的可执行文件是\_\_\_B\_\_\_\_。**

A．TEST.BAK和TEST.OBJ B．TEST.OBJ和TEST.EXE

C．TEST.EXE 和TEST.C D．TEST.BAK和TEST.EXE

**3．一个C程序的执行是从\_\_\_A\_\_\_\_。**

A．本程序的main函数开始，到main函数结束

B．本程序文件的第一个函数开始，到本程序文件的最后一个函数结束

C．本程序的main函数开始，到本程序文件的最后一个函数结束

D．本程序文件的第一个函数开始，到本程序main函数结束

**4．以下叙述正确的是\_\_\_C\_\_\_\_。**

A．在C程序中，main函数必须位于程序的最前面

B．程序的每行中只能写一条语句

C．C语言本身没有输入输出语句

D．在对一个C程序进行编译的过程中，可发现注释中的拼写错误

**5．以下叙述不正确的是\_\_\_C\_\_\_\_。**

A．一个C源程序可由一个或多个函数组成

B．一个C源程序必须包含一个main函数

C．在C程序中，注释说明只能位于一条语句的后面

D．C程序的基本组成单位是函数

**6．一个C语言程序是由\_\_\_B\_\_\_\_。**

A．一个主程序和若干子程序组成

B．函数组成

C．若干过程组成

D．若干子程序组成

**7．关于C语言程序书写规则说法中正确的是\_\_\_D\_\_\_\_。**

A．程序语句必须占一行 B．不区分大小写

C．每行必须有行号 D．每条语句用分号结束

**8．下面关于算法的说法，错误的是\_\_\_B\_\_\_\_。**

A．算法必须有输出，但不一定有输入

B．算法必须在计算机上用某种语言实现

C．算法必须在有限步执行后能结束

D．算法的每一步骤必须有确切的定义

**9．结构化程序设计规定的3种基本控制结构是\_\_\_D\_\_\_\_。**

A．顺序、选择和转向 B．层次、网状和循环

C．模块、选择和循环 D．顺序、选择和循环

**10．结构化程序由三种基本结构组成，三种基本结构组成的算法\_\_\_A\_\_\_\_。**

A．可以完成任何复杂的任务 B．只能完成部分复杂的任务

C．只能完成符合结构化的任务 D．只能完成一些简单的任务

**二、填空题**

1．C源程序的基本单位是**\_\_\_函数\_\_\_**，一个C源程序中必须包括一个**\_\_main()函数\_**。

2．在一个C源程序中，注释部分两侧的分界符分别为**\_\_\_/\*\_\_\_**和**\_\_\_\*/\_\_\_**。

3．C语言源程序文件的后缀是**\_\_\_.C\_\_\_**；经过编译后，生成文件的后缀是**\_\_.OBJ\_\_**；经过连接后，生成的文件的后缀是**\_\_.EXE\_\_**。

4．结构化程序由 **\_顺序结构\_、\_\_选择结构\_、\_\_循环结构\_\_**，3种基本结构组成。

**第2章 数据类型及其运算**

**一、选择题**

**1．下面四个选项中，均是不合法的用户标识符的选项的是\_\_\_C\_\_\_\_。**

A．A p\_o do B．float lao \_A

C．b-a goto int D．\_123 temp INT

**2．以下选项中可以作为C语言中合法整数的是\_\_\_C\_\_\_\_。**

A． 10110B B． 0386 C． 0Xffa D． x2a2

**3．以下选项中合法的实型常数是\_\_\_C\_\_\_\_。**

A．5E2.0 B．E-3 C．2E0 D．1.3E

**4．以下选项中可作为C语言合法常量的是\_\_\_A\_\_\_\_。**

A．-80 B．-080 C．-8e1.0 D．-80.0e

**5．下面四个选项中，均是合法转义字符的选项是\_\_\_B\_\_\_\_。**

A．'\' ' \\' '\n' B．'\'' '\017' '\t'

C．'\018' ' \f' '\xab' D．'\\0' '\101' 'x1f'

**6．下面不正确的字符串常量是\_\_\_A\_\_\_\_。**

A．'abc' B．"12’12" C．"0" D．" "

**7．以下选项中不属于C语言的类型的是\_\_\_D\_\_\_\_。**

A．signed short int B．unsigned long int

C．unsigned int D．long short

**8．在C语言中，要求运算数必须是整型的运算符是\_\_\_D\_\_\_\_。**

A．/ B．++ C．!= D．%

**9．以下运算符中优先级最低的是\_\_\_C\_\_\_，以下运算符中优先级最高的是\_\_\_B\_\_\_\_。**

A．&& B．<= C．|| D．==

**10．若x = 2 , y=3，则x && y的结果是\_\_\_B\_\_\_\_。**

A．0 B．1 C．2 D．3

**11．若x，i，j和k都是int型变量，则计算表达式x = ( i = 4 , j = 16 , k = 32 )后，x的值为\_\_\_C\_\_\_\_。**

A．4 B．16 C．32 D．52

**12．若有定义：int a = 7 ; float x = 2.5 , y = 4.7 ; ，则表达式x + a % 3 \* ( int )( x + y ) % 2 / 4的值是\_\_\_A\_\_\_\_。**

A．2.500000 B．2.750000 C．3.500000 D．0.000000

**13．若有以下类型说明语句：char w ; int x ; float y ; double z ; ，则表达式w \* x + z - y的结果为\_\_\_D\_\_\_\_类型。**

A．float B．char C．int D．double

**14．已有定义：int x = 3 , y = 4 , z = 5 ; 则表达式!( x + y ) + z – 1 && y + z / 2的值是\_\_\_D\_\_\_\_。**

A．6 B．0 C．2 D．1

**15．以下选项中非法的表达式是\_\_\_D\_\_\_\_。**

A．0 <= x < 100 B．i = j == 0 C．( char )( 65 + 3 ) D．x + 1 = x + 1

**16．设a、b、c、d、m、n均为int型变量，且a = 5、b = 6、c = 7、d = 8、m = 2、n = 2，则逻辑表达式( m = a > b ) && ( n = c > d)运算后，n的值为\_\_\_C\_\_\_\_。**

A．0 B．1 C．2 D．3

**17．若变量c为char类型，能正确判断出c为小写字母的表达式是\_\_\_D\_\_\_\_。**

A．'a' <= c <= 'z' B．( c >= 'a' ) || ( c <= 'z' )

C．( 'a' <= c ) and ( 'z' >= c ) D．( c >= 'a' ) && ( c <= 'z' )

**18．若x和y代表整型数，以下表达式中不能正确表示数学关系|x―y|<10的是\_\_\_C\_\_\_\_。**

A．abs( x – y ) < 10 B．x – y > -10 && x – y < 10

C．!( x – y ) < -10 || !( y – x ) > 10 D．( x – y ) \* ( x – y ) < 100

**二、填空题**

1．C语言中的标识符只能由三种字符组成，它们是**\_\_字母\_\_，\_\_数字\_\_**和**\_\_下划线\_\_\_。**

2．在C语言中（以32位PC机为例），一个char型数据在内存中所占的字节数为**\_\_1\_\_\_**，一个int型数据在内存中所占的字节数为**\_\_\_4\_\_\_**，一个float型数据在内存中所占的字节数为**\_\_\_4\_\_\_**，一个double型数据在内存中所占的字节数为**\_\_\_8\_\_\_**。

3．有以下定义int m = 5，y = 2；则计算表达式y += y -= m \*= y后的y值是**\_\_-16\_\_**。

4．若定义：int a = 2，b = 3；float x = 3.5，y = 2.5；则表达式( float )(a + b ) / 2 + ( int ) x % ( int ) y的值是**\_\_3.5\_\_**。

5．x和n均为int型变量，且x的初值为12，n的初值为5，则计算表达式x %= ( n %= 2 )后x的值为**\_\_0\_\_**。

6．设所有变量均为整型，则表达式( a = 2 , b = 5 , a++ , b++ , a + b )的值为**\_\_\_9\_\_\_**。

7．a、b和c均是int型变量，则计算表达式a = ( b = 4 ) + ( c = 2 )后，a值为**\_\_6\_\_\_**，b值为**\_\_\_4\_\_\_**，c值为**\_\_\_2\_\_\_**。

8．a是int型变量，且a的初值为6，则计算表达式a += a -= a \* a后a的值为**\_\_-60\_\_**。

9．设y是int型变量，请写出判断y为奇数的关系表达式**\_\_y%2==1\_\_**。

10．若a，b，c均为整型变量，则执行以下语句a = b = c = 0 ; a++ && ++b || c++ ; 后，a的值为**\_\_\_1\_\_\_**，b的值为**\_\_\_0\_\_\_**，c的值为**\_\_1\_\_**。

**三、程序阅读题**

1．下面程序的运行结果是**\_\_\_b\_\_\_**。

#include<stdio.h>

int main( )

{

char m ;

m = 'B' + 32 ;

printf( "%c\n" , m ) ;

return 0 ;

}

2．下面程序的运行结果是**\_\_\_0\_\_\_\_**。

#include<stdio.h>

int main( )

{

int x = 3 , y = 5 ;

printf( "%d\n" , x / y ) ;

return 0 ;

}

3．下面程序的运行结果是**\_\_\_0\_\_\_\_**。

#include<stdio.h>

int main( )

{

int m = 0xabc , n = 0xabc ;

m -= n ;

printf( "%x\n" , m ) ;

return 0 ;

}

4．下面程序的运行结果是**\_\_10 , 2\_\_**。

#include<stdio.h>

int main( )

{

int x = 10 , y = 3 ;

printf( "%d,%d\n" , x++ , --y ) ;

return 0 ;

}

5．下面程序的运行结果是**\_\_9 , 10 , 9 , 10\_\_**。

#include<stdio.h>

int main( )

{

int i , j ;

i = 8 ;

j=10;

printf( "%d,%d,%d,%d\n" , i , j , ++i , j++ ) ;

return 0 ;

}

6．下面程序的运行结果是**\_\_3 , 1 , 0 , 0\_\_**。

#include<stdio.h>

int main( )

{

int a = 1 , b = 2 , c = 3 ;

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

return 0 ;

}

7．下面程序的运行结果是**\_\_3 , 20 , 30 , 1\_\_**。

#include<stdio.h>

int main( )

{

int a = 2 , b=20 , c=30 , d ;

d = ++a <= 10 || b-- >= 20 || c++ ;

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

return 0 ;

}

**第3章 语句与输入输出**

**一、选择题**

**1．设有定义：long x = 123456L ; ，则以下能够正确输出变量x值的语句是\_\_\_B\_\_\_。**

A．printf( "x=%d\n" , x ) ; B．printf( "x=%ld\n" , x ) ;

C．printf( "x=%8dL\n" , x ) ; D．printf( "x=%LD\n" , x ) ;

**2．若有以下程序段**

int m = 0xabc , n = 0xabc ;

m -= n ;

printf( "%X\n" , m ) ;

**执行后输出结果是\_\_\_C\_\_\_\_。**

A．0X0 B．0x0 C．0 D．0XABC

**3．已知char a = '\103' ；则语句printf( "%3d" , a ) ; 执行后的输出结果为\_\_\_C\_\_\_\_。**

A．67 B．'C' C． 67 D．103

**4．已知a、b、c为int类型，执行语句：scanf( "a=%d, b=%d, c=%d" , &a , &b , &c ) ; ，若要使得a为1，b为2，c为3。则以下选项中正确的输入形式是\_\_\_C\_\_\_\_。**

A． a=1 B．1, 2, 3 C．a=1, b=2, c=3 D．1 2 3

b=2

c=3

**5．已知i、j、k为int型变量，若从键盘输入：1,2,3↙，使i的值为1、j的值为2、k的值为3，以下选项中正确的输入语句是\_\_\_C\_\_\_。**

A．scanf( "%2d%2d%2d" , &i , &j , &k ) ; B．scanf( "%d %d %d" , &i , &j , &k ) ;

C．scanf( "%d,%d,%d" , &i , &j , &k ) ; D．scanf( "i=%d,j=%d,k=%d" , &i , &j , &k ) ;

**6．x、y、z被定义为int型变量，若从键盘给x、y、z输入数据，正确的输入语句是\_\_\_B\_\_\_\_。**

A．INPUT x、y、z;

B．scanf( "%d,%d,%d" , &x , &y , &z ) ;

C．scanf( "%d,%d,%d" , x , y , z ) ;

D．read( "%d,%d,%d" , &x , &y , &z ) ;

**7．若有int a ; float b ; scanf( "%d,%f" , &a , &b ) ; 使a、b的值分别为30和5.5，则正确的输入是\_\_\_B\_\_\_\_。**

A．30 5.5 B．30, 5.5 C．a=30, b=5.5 D．a=30 b=5.5

**8．根据以下定义语句和数据的输入方式，scanf语句的正确形式应为\_\_\_B\_\_\_\_。**

已有定义：float f1 , f2 ;

数据的输入方式：4.52↙

3.5↙

A．scanf( "%f, %f" , &f1 , &f2 ) ;

B．scanf( "%f%f" , &f1 , &f2 ) ;

C．scanf( "%3.2f %2.1f" , &f1 , &f2 ) ;

D．scanf( "%3.2f%2.1f" , &f1, &f2 ) ;

**9．以下针对scanf()函数的叙述中，正确的是\_\_\_D\_\_\_\_。**

A．输入项可以为一实型常量，如scanf( "%f" , 3.5 ) ;

B．只有格式控制，没有输入项，也能进行正确输入，如scanf( "a=%d, b=%d" ) ;

C．当输入一个实型数据时，格式控制部分应规定小数点后的位数，如：scanf( "%4.2f" , &f ) ;

D．当输入数据时，必须指明变量的地址，如scanf( "%f" , &f ) ;

**10．有以下程序**

#include<stdio.h>

int main( )

{

char c1 = '1' , c2 = '2' ;

c1 = getchar( ) ; c2 = getchar( ) ;

putchar( c1 ) ; putchar( c2 ) ;

}

**当运行时输入a↙后，以下叙述正确的是\_\_\_A\_\_\_\_。**

A．变量c1被赋予字符a,c2被赋予回车符

B．程序将等待用户输入2个字符

C．变量c1被赋予字符a,c2中仍是原有字符2

D．变量c1被赋予字符a,c2中将无确定值

**二、填空题**

1．已知float f = 123.467 ; ，则执行语句printf( "%.2f\n" , f ) ; ，输出结果是**\_\_123.47\_\_**。

2．已知字符'A'的ASCII值为十进制65，变量c为字符型，则执行语句c = 'A' + '6' - '3' ; printf( "%c\n " , c ) ; ，输出结果是**\_\_\_D\_\_\_**。

3．scanf()函数在输入数据时默认的分隔符有**\_\_回车\_\_**、Tab、空格。

4．已知：int i = 10 , j = 1 ; 执行语句 printf( "%d,%d" , i++ , ++j ) ; 后输出结果**\_10 , 2\_**。

**三、程序阅读题**

**1．下面程序的运行结果是\_\_\_261\_\_\_\_。**

#include<stdio.h>

int main( )

{

int m = 177 ;

printf( "%o\n" , m ) ;

return 0 ;

}

**2．下面程序的运行结果是\_\_20\_\_\_。**

#include<stdio.h>

int main( )

{

int n = 0 ;

n += ( n = 10 ) ;

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

return 0 ;

}

**3．下面程序的运行结果是\_\_201 , 10\_\_。**

#include<stdio.h>

int main( )

{

int a = 201 , b = 012 ;

printf( "%2d,%2d\n" , a , b ) ;

return 0 ;

}

**y=4630**

**4．下面程序的运行结果是\_\_\_ y= 4630\_\_\_\_。**

#include<stdio.h>

int main( )

{

int y = 2456 ;

printf( "y=%3o\n" , y ) ;

printf( "y=%8o\n" , y ) ;

return 0 ;

}

**5．下面程序的运行结果是\_\_\*3.140000 ，3.142\*\_\_\_。**

#include <stdio.h>

int main( )

{

printf( "\*%f,%4.3f\*\n" , 3.14 , 3.1415 ) ;

return 0 ;

}

**6．下面程序的运行结果是\_\_ c:dec=120 ，oct=170 ，hex=78 ，ASCII=x \_\_\_。**

#include<stdio.h>

int main( )

{

char c = 'x' ;

printf( "c:dec=%d,oct=%o,hex=%x,ASCII=%c\n" , c , c , c , c ) ;

return 0 ;

}

**x=1 y=2 \*sum\*=3**

**7．下面程序的运行结果是\_\_10 squared is : 100\_\_\_。**

#include<stdio.h>

int main( )

{

int x = 1 , y = 2 ;

printf( "x=%d y=%d \*sum\*=%d\n" , x , y , x + y ) ;

printf( "10 squared is : %d\n" , 10 \* 10 ) ;

return 0 ;

}

**8．下面程序的运行结果是\_\_2 48 20.0 20.0\_\_\_。**

#include<stdio.h>

int main( )

{

int n = 100 ;

char c ;

float f = 10.0 ;

double x ;

x = f \*= n /= ( c = 48 ) ;

printf( "%d %d %3.1f %3.1f\n" , n , c , f , x ) ;

return 0 ;

}

**9．下面程序的运行结果是\_\_ x+y+z=48\_\_。**

运行时从键盘输入：25 13 10↙。

#include<stdio.h>

int main( )

{

int x , y , z ;

scanf( "%d%d%d" , &x , &y , &z ) ;

printf( "x+y+z=%d\n" , x + y + z ) ;

return 0 ;

}

**10．下面程序的运行结果是\_\_55 ，空格 ，A \_\_\_\_\_。**

运行时从键盘输入：55 A B↙。

#include <stdio.h>

int main( )

{

int k = 0 ; char c1 = 'a' , c2 = 'b' ;

scanf("%d%c%c" , &k , &c1 , &c2 ) ;

printf( "%d,%c,%c\n" , k , c1 , c2 ) ;

return 0 ;

}

**第4章 选择结构程序设计**

**一、选择题**

**1．if语句的选择条件是\_\_\_D\_\_\_\_。**

A．只能用关系表达式 B．只能用关系表达式或逻辑表达式

C．只能用逻辑表达式 D．可以用任何表达式

**2．已知int x = 2 , y = -1 , z = 3 ;，执行下面语句后，z的值是\_\_\_C\_\_\_\_。**

if( x < y ) if( y < 0 ) z = 1 ; else z++ ;

A．1 B．2 C．3 D．4

**3．以下程序段的输出结果是\_\_\_C\_\_\_\_。**

int a = 2 , b = 1 , c = 2 ;

if( a < b ) if( b < 0 ) c = 0 ; else c += 1 ;

printf( "%d\n" , c ) ;

A．0 B．1 C．2 D．3

**4．当a = 1，b = 3，c = 5，d = 4时，执行下面一段程序后，x的值为\_\_B\_\_\_\_\_。**

if( a < b )

if( c < d ) x = 1 ;

else

if( a < c )

if( b < d ) x = 2 ;

else x = 3 ;

else x = 6 ;

else x = 7 ;

A．1 B．2 C．3 D．6

**5．为了避免嵌套的if-else的二义性，C语言规定，else与\_\_\_B\_\_\_\_配对。**

A．与最外层的if B．其之前最近的不带else的if

C．其之后最近的if D．与最近的{ }之前的if

**6．若有定义：float w ; int a , b ; 则合法的switch语句是\_\_\_D\_\_\_\_。**

A． switch( w ) B． switch( a )

{ {

case 1.0 : printf( "\*\n" ) ; case 1 printf( "\*\n" ) ;

case 2.0 : printf( "\*\*\n" ) ; case 2 printf( "\*\*\n" ) ;

} }

C． switch( b ) D． switch(a+b);

{ {

case 1 : printf( "\*\n" ) ; case 1 : printf( "\*\n" ) ;

default : printf( "\n" ) ; case 2 : printf( "\*\*\n" ) ;

case 1 + 2 : printf( "\*\*\n" ) ; default : printf( "\n" ) ;

} }

**二、程序阅读题**

**1．下面程序的运行结果是\_\_\_10\_\_\_\_。**

运行时从键盘输入：9↙。

#include<stdio.h>

int main( )

{

int n ;

scanf( "%d" , &n ) ;

if( n++ < 10 ) printf( "%d\n" , n ) ;

else printf( "%d\n" , n-- ) ;

return 0 ;

}

**2．下面程序的运行结果是\_\_2 ，2 ，2\_\_。**

#include<stdio.h>

int main( )

{

int i = 1 , j = 1 , k = 2 ;

if( ( j++ || k++ ) && i++ ) printf( "%d,%d,%d\n" , i , j , k ) ;

return 0 ;

}

**3．下面程序的运行结果是\_\_\_6\_\_\_。**

#include<stdio.h>

int main( )

{

int m = 5 ;

if(m++ > 5) printf( "%d\n" , m ) ;

else printf( "%d\n" , m-- ) ;

return 0 ;

}

**4．下面程序的运行结果是\_\_\_97 ，b\_\_\_\_。**

#include<stdio.h>

int main( )

{

char c1 = 97 ;

if(c1 >= 'a' && c1 <= 'z' )

printf( "%d,%c" , c1 , c1 + 1 ) ;

else

printf( "%c" , c1 ) ;

return 0 ;

}

**5．下面程序的运行结果是\_\_C=-1\_\_。**

#include<stdio.h>

int main( )

{

int a = 1 , b = 2 , c = 3 ;

if( a > b ) c = 1 ;

else

if( a == b ) c = 0 ;

else c = -1 ;

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

return 0 ;

}

**6．下面程序的运行结果是\_\_8888\_\_。**

运行时从键盘输入：88↙。

#include<stdio.h>

int main( )

{

int a ;

scanf( "%d" , &a ) ;

if( a > 90 ) printf( "%d" , a ) ;

if( a > 80 ) printf( "%d" , a ) ;

if( a > 70 ) printf( "%d" , a ) ;

return 0 ;

}

**7．下面程序的运行结果是\_\_20 ，0\_\_。**

#include<stdio.h>

int main( )

{

int x = 10 , y = 20 , t = 0 ;

if( x == y ) t = x ; x = y ; y = t ;

printf( "%d,%d\n" , x , y ) ;

return 0 ;

}

**8．下面程序的运行结果是\_\_2 ，1\_\_。**

#include<stdio.h>

int main( )

{

int x = 1 , a = 0 , b = 0 ;

switch( x )

{

case 0 : b++ ;

case 1 : a++ ;

case 2 : a++ ; b++ ;

}

printf( "%d,%d" , a , b ) ;

return 0 ;

}

**9．下面程序的运行结果是\_\_\_\_\_\_\_。**

**1 ，1**

**2 ，2**

**2 ，1**

**-2 ，2**

运行时从键盘输入：1↙。

#include stdio.h>

int main( )

{

int k ;

scanf( "%d" , &k ) ;

switch( k )

{

case 1: printf( "%d,%d\n" , k++ , k ) ;

case 2: printf( "%d,%d\n" , k , k-- ) ;

case 3: printf( "%d,%d\n" , ++k , k ) ;

case 4: printf( "%d,%d\n" , -k , k++ ) ; break ;

default: printf( "full!\n" ) ;

}

return 0 ;

}

**10．下面程序的运行结果是\_\_a=1 ，b=3\_\_。**

#include<stdio.h>

int main( )

{

int x = 1 , y = 1 , a = 0 , b = 0 ;

switch( x )

{

case 1: switch( y )

{

case 0: a++ ;

case 1: b++ ;

case 2: b++ ; break ;

}

case 2: a++ ; b++ ; break ;

case 3: a++ ; b++ ;

}

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

return 0 ;

}

**三、程序完善题**

**1．下面程序用于将输入的三个数从小到大排序输出。**

#include<stdio.h>

int main( )

{ int a , b , c , x , y ;

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

if**(\_\_a>b\_\_)** { x = a ; y = b ; }

else { x = b ; y = a ; }

if**(\_\_c>x\_\_)** x = c ;

else if( y > c ) y = c ;

printf( "%d,%d,%d" , y , a + b + c – x – y , x ) ;

return 0 ;

}

**第5章 循环结构程序设计**

**一、选择题**

**1．有以下程序段**

int k = 10 ;

while( k = 0 ) k = k - 1 ;

则下面描述中正确的是\_\_\_C\_\_\_\_。

A．while循环执行10次 B．循环是无限循环

C．循环体语句一句也不执行 D．循环体语句执行一次

**2．有以下程序段**

int x = 0 , s = 0 ;

while( !x != 0 ) s += ++x ;

printf( "%d" , s ) ;

**则**\_\_\_B\_\_\_\_**。**

A．运行程序段输出0 B．运行程序段输出1

C．程序段中的控制表达式是非法的 D．程序段执行无限次

**3．下面程序的功能是将从键盘输入的一对数，由小到大排序输出。当输入一对相等数时结束循环，请选择填空。**

#include<stdio.h>

int main( )

{

int a , b , t ;

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

while(\_\_\_B\_\_\_\_)

{

if( a > b )

{

t = a ;

a = b ;

b = t ;

}

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

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

}

return 0 ;

}

A．!a = b B．a != b C．a == b D．a = b

**4．下面程序的功能是在输入的一批正整数中求出最大者，输入0结束循环，请选择填空。**

#include<stdio.h>

int main( )

{

int a , max = 0 ;

scanf( "%d" , &a ) ;

while(\_\_\_B\_\_\_\_)

{

if( max < a )

max = a ;

scanf( "%d" , &a ) ;

}

printf( "%d" , max ) ;

return 0 ;

}

A．a == 0 B．a C．!a == 1 D．!a

**5．C语言中while和do－while循环的主要区别是**\_\_\_A\_\_\_\_**。**

A．do－while的循环体至少无条件执行一次

B．while的循环控制条件比do－while的循环控制条件严格

C．do－while允许从外部转到循环体内

D．do－while的循环体不能是复合语句

**6．下面程序的运行结果是**\_\_\_B\_\_\_\_**。**

#include<stdio.h>

int main( )

{

int a = 1, b = 10 ;

do

{

b -= a ;

a++ ;

}

while( b-- < 0 ) ;

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

return 0 ;

}

A．a=3, b=11 B．a=2, b=8 C．a=1, b=-1 D．a=4,b=9

**7．若 i为整型变量，则以下循环执行次数是**\_\_\_B\_\_\_\_**。**

for( i = 2 ; i == 0 ; ) printf( "%d" , i-- ) ;

A．无限次 B．0次 C．1次 D．2次

**8．下面程序的功能是计算1到10之间的奇数之和及偶数之和。请选择填空。**

#include<stdio.h>

int main( )

{

int a , b , c , i ;

a = c = 0 ;

for( i = 0 ; i <= 10 ; i += 2 )

{

a += i ;

\_\_\_B\_\_\_\_ (1) ;

c += b ;

}

printf( "sum of the even = %d\n" , a ) ;

printf( "sum of the odd = %d\n" , \_\_\_C\_\_\_\_ (2) ) ;

return 0 ;

}

(1) A．b = i-- B．b = i + 1 C．b = i++ D．b = i - 1

(2) A．c - 10 B．c C．c - 11 D．c - b

**9．有以下程序**

#include<stdio.h>

int main( )

{

int i ;

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

printf( "%d\n" , i ) ;

return 0 ;

}

**则下面描述中正确的是**\_\_\_D\_\_\_\_**。**

A．输出1 B．输出2 C．输出3 D．死循环

**10．for循环语句：for( 表达式1 ; 表达式2 ; 表达式3 )语句，以下叙述正确的是**\_\_\_C\_\_\_\_**。**

A．for语句中的3个表达式一个都不能少

B．for语句中的循环体至少要执行一次

C．for语句中的循环体可以是一个复合语句

D．for语句只能用于循环次数已经确定的情况

**11．关于break和continue，以下说法正确的是**\_\_\_B\_\_\_\_**。**

A．break语句只应用在循环体中

B．continue语句只应用在循环体中

C．break是无条件跳转语句，continue不是

D．break和continue语句的跳转范围不够明确，容易产生错误

**二、程序阅读题**

**1．下面程序的运行结果是**\_\_1 ，2 ，0\_\_**。**

#include<stdio.h>

#include<math.h>

int main( )

{

int a = 1 , b = 2 , c = 2 , t ;

while( a < b < c )

{

t = a ;

a = b ;

b = t ;

c -- ;

}

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

return 0 ;

}

**2．有以下程序**

#include<stdio.h>

int main( )

{

int a , b , m , n ;

m = n = 1 ;

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

do

{

if( a > 0 )

{

m = 2 \* n ;

b++ ;

}

else

{

n = m + n ;

a += 2 ;

b++ ;

}

}

while( a == b ) ;

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

return 0 ;

}

**若输入输入－1 0↙。程序的运行结果是**\_\_m=4 n=2\_\_**。**

**3．下面程序的运行结果是**\_\_A2C4E6\_\_**。**

#include<stdio.h>

int main( )

{

char c1 , c2 ;

int a ;

c1 = '1' ;

c2 = 'A' ;

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

{

if( a % 2 )

putchar( c1 + a ) ;

else

putchar( c2 + a ) ;

}

return 0 ;

}

**4．下面程序的运行结果是**\_\_132\_\_**。**

#include<stdio.h>

int main( )

{

int i , m = 0 , n = 0 , k = 0 ;

for( i = 9 ; i <= 11 ; i++ )

{

switch( i / 10 )

{

case 0 :

m++ ;

n++ ;

break ;

case 10 :

n++ ;

break ;

default :

k++ ;

n++ ;

}

}

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

return 0 ;

}

**5．下面程序的运行结果是**\_\_\_4\_\_\_\_**。**

#include<stdio.h>

int main( )

{

int a = 1 , b ;

for( b = 1 ; b <= 10 ; b++ )

{

if( a >= 8 )

break ;

if( a % 2 == 1 )

{

a += 5 ;

continue ;

}

a = a - 3 ;

}

printf( "%d\n" , b ) ;

return 0 ;

}

**6．下面程序的运行结果是**\_\_\_ k=0,m=5\_\_\_\_**。**

#include<stdio.h>

int main( )

{

int m = 0 , k = 0 , i , j ;

for( i = 0 ; i < 2 ; i++ )

{

for( j = 0 ; j < 3 ; j++ )

k++ ;

k = k - j ;

}

m = i + j ;

printf( "k=%d,m=%d\n" , k , m ) ;

return 0 ;

}

**7．下面程序的运行结果是**\_\_\_x=8\_\_\_\_**。**

#include<stdio.h>

int main( )

{

int i , j , x = 0 ;

for( i = 0 ; i < 2 ; i++ )

{

x++ ;

for( j = 0 ; j <= 3 ; j++ )

{

if( j % 2 )

continue ;

x++ ;

}

x++ ;

}

printf( "x=%d\n" , x ) ;

return 0 ;

}

**8．有以下程序**

#include<stdio.h>

#include<math.h>

int main( )

{

float x , y , z ;

scanf( "%f,%f" , &x , &y ) ;

z = x / y ;

while( 1 )

{

if( fabs( z ) > 1.0 )

/\* fabs()是计算绝对值的函数，包含在math.h文件中，此时计算|Z| \*/

{

x = y ;

y = z ;

z = x / y ;

}

else

break ;

}

printf( "%3.1f\n" , y ) ;

return 0 ;

}

**若输入数据3.6,2.4↙。程序的运行结果是**\_\_\_1.6\_\_\_\_**。**

**9．有以下程序**

#include<stdio.h>

int main( )

{

char c ;

c = getchar( );

while ( c != '\n' )

{

switch ( c - '2' )

{

case 0 :

case 1: putchar( c + 4 ) ;

case 2: putchar( c + 4 ) ;

break;

case 3: putchar( c + 3 ) ;

case 4: putchar( c + 2 ) ;

break;

}

c = getchar( );

}

printf( "\n" ) ;

return 0 ;

}

**若输入数据7654↙。程序的运行结果是**\_\_\_8878\_\_\_\_**。**

**三、程序完善题**

**1．下面程序的功能是：按规律将电文变成密码，即将字母A变成字母E，a变成e，即变成其后的第4个字母，W变成A，X变成B，Y变成C，Z变成D；非字母字符保持原状不变。如“boy”转换为“fsc”。从键盘输入一行字符，用换行符结束输入，输出其相应的密码。请填空。**

#include<stdio.h>

int main( )

{

char ch ;

printf( " please enter:\n" ) ;

ch = getchar( ) ;

while( ch != '\n' )

{

if( ( ch >= 'a' && ch <= 'z' ) || ( ch >= 'A' && ch <= 'Z' ) )

{

ch = ch + 4 ;

if\_\_\_( ch > 'Z' && ch <= 'Z' + 4 ) ||(ch > 'z' ) \_\_\_\_

ch = \_\_ch-26\_\_;

}

printf( "%c" , ch ) ;

ch = getchar( ) ;

}

printf( "\n" ) ;

return 0 ;

}

**2．下面程序的功能是：计算100至1000之间有多少个数其各位数字之和是5。请填空。**

#include<stdio.h>

int main( )

{

int i , s , k , count = 0 ;

for( i = 100 ; i <= 1000 ; i++ )

{

s = 0 ;

k = i ;

while(\_\_\_k\_\_\_\_)

{

s = s + k % 10 ;

k =\_\_k/10\_\_\_\_\_ ;

}

if( s != 5 )

\_\_continue\_\_\_\_\_ ;

else

count++;

}

printf( "%d" , count ) ;

return 0 ;

}

**3．下面程序的功能是计算：s=1+12+123+1234+12345。请填空。**

**#include<stdio.h>**

int main( )

{

int t = 0 , s = 0 , i ;

for( i = 1 ; i <= 5 ; i++ )

{

t =\_\_t\*10+i\_\_\_;

\_s=s+t\_;

}

printf( "s=%d\n" , s ) ;

return 0 ;

}

**4．下面程序的功能是：用公式****求的近似值，直到发现某一项的绝对值小于为止（该项不累加）。请填空。**

#include<stdio.h>

#include<math.h>

int main( )

{

int f = 1 ;

float pi = 0.0 , n = 1 , t = 1 ;

while(\_\_fabs( t ) >= 1e-6\_\_)

{

pi = pi + t ;

n = n + 2 ;

\_\_f=-f\_\_\_\_\_;

t = f / n ;

}

pi = pi \* 4 ;

printf( "pi=%8.6f\n" , pi ) ;

return 0 ;

}

**5．下面程序的功能是：计算100以内能被3整除，且个位数为4的所有整数。请填空。**

#include<stdio.h>

int main( )

{

int i , j ;

for( i = 0 ; \_\_\_\_i<10\_\_\_; i++ )

{

j = i \* 10 + 4 ;

if(\_\_j%3!=0\_\_)

continue;

printf( "%d\n" , j ) ;

}

return 0 ;

}

**第6章 数组**

**一、选择题**

**1．在C语言中，引用数组元素时，其数组下标的数据类型不允许是\_\_\_D\_\_\_\_。**

A．整型常量 B．整型表达式

C．整型常量或整型表达式 D．任何类型的表达式

**2．以下对一维整型数组a的定义，正确的是\_\_\_D\_\_\_\_。**

A．int a(10) ; B．int n = 10 , a[n] ;

C．int n ; D．int a[10] ;

scanf( "%d" , &n ) ;

int a[n] ;

**3．若有定义：int a[10] ;，则对a数组元素的正确引用是\_\_\_D\_\_\_\_。**

A．a[10] B．a[3.5] C．a(5) D．a[10-10]

**4．对定义 int a[10] = {6 , 7 , 8 , 9 , 10} ; 的正确理解是\_\_\_B\_\_\_\_。**

A．将5个初值依次赋给a[1]--a[5]

B．将5个初值依次赋给a[0]--a[4]

C．将5个初值依次赋给a[6]--a[10]

D．因为数组长度与初值个数不相同，所以此语句不正确

**5．以下对二维数组a的正确说明是\_\_\_C?\_\_\_\_。**

A．int a[3][ ]; B．float a(3)(4);

C．double a[ ][4] ; D．float a(3,4);

**6．若有定义： int a[3][4]; , 则对a数组元素的正确引用是\_\_\_C\_\_\_\_。**

A．a[3][4] B．a[1,3] C．a[1+1][0] D．a(2)(1)

**7．以下对二维数组a初始化正确的语句是\_\_\_B\_\_\_\_。**

A．int a[2][ ]={{0 , 1 , 2}, {3 , 4 , 5}};

B．int a[ ][3]={{0, 1, 2}, {3, 4, 5}};

C．int a[2][4]={{0, 1 , 2}, {3 , 4}, {5}};

D．int a[ ][3]={{0, 1, 2}, { }, {3, 4}};

**8．对二维数组a进行如下初始化**

int a[ ][3]={0 , 1 , 2 , 3 , 4 , 5};

则a[1][1]的值是**\_\_\_C\_\_\_\_**。

A．0 B．3 C．4 D． 1

**9．下面程序段的运行结果是\_\_\_\_C\_\_\_。**

#include<stdio.h>

int main( )

{

int i , x[3][3] = {1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9} ;

for( i = 0 ; i < 3 ; i++ )

printf( "%2d" , x[i][2-i] ) ;

return 0 ;

}

A．1 5 9 B．1 4 7 C．3 5 7 D．3 6 9

**10．以下对数组s的初始化，错误的是\_\_\_D\_\_\_\_。**

A．char s[5] = { "abc" } ; B．char s[5] = { 'a' , 'b' , 'c' } ;

C．char s[5] = "" ; D．char s[5] = "abcde" ;

**11．对两个数组a和b进行如下初始化，**

char a[ ] = "ABCDEF" ;

char b[ ] = { 'A' , 'B' , 'C' , 'D' , 'E' , 'F' } ;

则以下叙述正确的是**\_\_\_D\_\_\_\_**。

A．a和b数组完全相同 B．a与b长度相同

C．a的长度比b短 D．a的长度比b长

**12．有两个字符数组a，b，则以下正确的输入语句是\_\_\_B\_\_\_\_。**

A．gets( a , b ) ; B．scanf( "%s,%s" , a , b ) ;

C．scanf( "%s%s" , &a , &b ) ; D．gets( "a" ) , gets( "b" ) ;

**13．下面程序段的运行结果是\_\_\_D\_\_\_\_。**

char a[7] = "abcdef" ;

char b[4] = "ABC" ;

strcpy( a , b ) ;

printf( "%c\n" , a[5] ) ;

A．□(□表示一个空格) B．\0 C．e D．f

**14．判断字符串s1是否大于字符串s2，应当使用\_\_\_D\_\_\_\_。**

A．if( s1>s2 ) B．if( strcmp( s1 , s2 ) )

C．if( strcpy( s1 , s2 ) ) D．if( strcmp( s1 , s2 )>0 )

**15．若有定义：char s[100] = "hello" ;，下列函数调用语句中，不正确的是\_\_\_C\_\_\_\_。**

A．strlen( strcpy( s , "well" ) ) ; B．strcat( s , "!" ) ;

C．puts( puts( "hello" ) ) ; D．!strcmp( "" , s ) ;

**16．下面程序的运行结果是\_\_\_D\_\_\_\_。**

#include<stdio.h>

#include<string.h>

int main( )

{

char a[80] = "AB" , b[80] = "LMNP" ;

int i = 0 ;

strcat( a , b ) ;

while( a[i++] != '\0' )

b[i] = a[i] ;

puts( b ) ;

return 0 ;

}

A．LB B．ABLMNP C．AB D．LBLMNP

**二、程序阅读题**

**1．下面程序的运行结果是\_\_\_8\_\_\_\_。**

#include<stdio.h>

int main( )

{

int i , k , a[10] , p[3] ;

k = 5 ;

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

a[i] = i ;

for( i = 0 ; i < 3 ; i++ )

p[i] = a[i \* i] ;

for( i = 0 ; i < 3 ; i++ )

k = p[i] \* 2 ;

printf( "%d\n" , k ) ;

return 0 ;

}

**2．下面程序的运行结果是\_\_\_4\_\_\_\_。**

#include<stdio.h>

int main( )

{

int i ,x[10] = {2 , 3 , 0 , 3 , 0 , 2 , 3 , 2 , 1 , 3} ,y[4] = {0} ;

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

y[x[i]]++ ;

printf( "%d\n" , y[3] ) ;

return 0 ;

}

**3．下面程序的运行结果是\_\_\_0 ，2\_\_\_\_。**

#include<stdio.h>

int main( )

{

int i , j , a[3][3] = {0} ;

for( i = 0 ; i < 3 ; i++ )

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

a[i][j] = i \* j ;

printf( "%d,%d\n" , a[1][2] , a[2][1] ) ;

return 0 ;

}

**4．下面程序的运行结果是\_\_\_12\_\_\_\_。**

#include<stdio.h>

int main( )

{

int a[3][3] = {0 , 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8} ;

int i , s = 0 ;

for( i= 0 ; i < 3 ; i++ )

s += a[i][1] ;

printf( "%d\n" , s ) ;

return 0 ;

}

**5．下面程序的运行结果是\_\_\_t\*M\_\_\_。**

#include<stdio.h>

int main( )

{

int i ;

char a[ ] = "Time" , b[ ] = "Tom" ;

for( i = 0 ; a[i] != '\0' && b[i] != '\0' ; i++ )

if( a[i] == b[i] )

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

printf( "%c" , a[i] - 32 ) ;

else printf( "%c" , a[i] + 32 );

else printf( "\*" ) ;

return 0 ;

}

**6．下面程序的运行结果是\_\_\_mo\_\_\_\_。**

#include<stdio.h>

int main( )

{

char a[ ]= "morning" , t ;

int i , j = 0 ;

for( i = 1 ; i < 7 ; i++ )

if( a[j] < a[i] )

j = i ;

t = a[j] ;

a[j] = a[7] ;

a[7] = a[j] ;

puts( a ) ;

return 0 ;

}

**三、程序完善题**

**1．下面程序的功能是输入5个整数，找出最大数和最小数所在的位置，并把二者对调，然后输出调整后的5个数。请填空。**

#include<stdio.h>

int main( )

{

int a[5] , max , min , i , j = 0 , k = 0 ;

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

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

min = a[0] ;

for( i = 1 ; i < 5 ; i++ )

if( a[i] < min )

{

min = a[i] ;

**\_\_k=i\_\_;**

}

max = a[0] ;

for( i = 1 ; i < 5 ; i++ )

if( a[i] > max )

{

max = a[i] ;

**\_\_j=i\_\_;**

}

**\_\_a[k]=max\_\_;**

**\_\_a[j]=min\_\_;**

printf( "\nThe position of min is:%3d\n" , k ) ;

printf( "\nThe position of max is:%3d\n" , j ) ;

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

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

return 0 ;

}

**2．下面程序的功能是：从键盘上输入10个学生的成绩，统计计算出平均成绩，并输出低于平均分的学生成绩。请填空。**

#include<stdio.h>

int main( )

{

int score[10] , n = 0 , i ;

float sum = 0.0 , avg ;

printf( "pleasr enter :\n" ) ;

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

{

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

**\_\_sum=sum+score[i]\_\_\_;**

}

avg = sum / 10;

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

if**(\_\_score[i]<avg\_\_)**

printf( "%3d" , score[i] ) ;

return 0 ;

}

**3．下面程序的功能是：将数组a[3][4]中的元素以3行4列的格式输出，然后按行求和并且存储到数组s中。请填空。**

#include<stdio.h>

int main( )

{

int a[3][4] = {0 , 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 , 11} ;

int s[3] = {0} ;

int i , j ;

for( i= 0 ; i < 3 ; i++ )

{

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

{

printf( "%4d" , a[i][j] ) ;

**\_\_s[i]=s[i]+a[i][j]\_\_;**

}

**\_\_printf(“\n”)\_\_\_\_\_;**

}

for( i= 0 ; i < 3 ; i++ )

printf( "%d\n" , s[i] ) ;

return 0 ;

}

**4．下面程序的功能是：将字符串str的内容倒过来存放。请填空。**

#include<stdio.h>

#include<string.h>

int main( )

{

char str[ ] = "abcdefg" ;

int i , j , k ;

**\_j=strlen(str)-1\_;**

for( i= 0 ; i < j ; i++ , j-- )

{

k = str[i] ;

str[i] = str[j] ;

**\_\_\_str[j]=k\_\_\_;**

}

printf( "%s\n" , str ) ;

return 0 ;

}

**5．下面程序的功能是：从终端输入字符，直到输入#为止，统计输入的字符中每个大写字母的个数，存放在num数组中，其中num[0]表示字母A的个数，num[1]表示字母B的个数，以此类推。请填空。**

#include<stdio.h>

int main( )

{

int num[26] = {0} , i ;

char c ;

while**(\_\_( c=getchar() )!=’#’\_\_)**

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

**\_\_\_num[c-‘A’]=num[c-‘A’]+1\_\_\_\_**；

for( i = 0 ; i < 26 ; i++ )

if( num[i] )

printf( "%c:%d\n" , i + 'A' , num[i] ) ;

return 0 ;

}

**第7章 函数**

**一、选择题**

**1．以下正确的说法是\_\_\_B\_\_\_\_。**

A．用户若需调用标准库函数，调用前必须重新定义。

B．用户可以重新定义标准库函数，若如此，该函数将失去原有含义。

C．系统根本不允许用户重新定义标准库函数。

D．用户若需调用标准库函数，调用前不必使用预编译命令将该函数所在文件包括到用户源文件中，由系统自动去调用。

**2．以下正确的函数形式是\_\_\_D\_\_\_\_。**

A．double fun( int x , int y )

{

z = x + y ;

return z ;

}

B．fun( int x , y )

{

int z ;

return z ;

}

C．fun( x , y )

{

int x , y ;

double z ;

z = x + y ;

return z ;

}

D．double fun( int x , int y )

{

double z ;

z = x + y ;

return z ;

}

**3．如果一个函数没有返回值，那么该函数的类型是\_\_\_D\_\_\_\_。**

A．int B．char C．float D．void

**4．C语言允许函数值类型缺省定义，此时该函数值隐含的类型是\_\_\_A\_\_\_\_。**

A．int B．char C．float D．void

**5．以下说法不正确的是\_\_\_B\_\_\_\_。**

A．实参可以是常量、变量或表达式

B．形参可以是常量、变量或表达式

C．形参可以为任意类型

D．形参应与其对应的实参类型一致

**6．一个函数形参的作用域是\_\_\_B\_\_\_\_。**

A．main函数 B．形参所在函数体

C．从定义处到文件尾 D．整个程序

**7．以下关于形参和实参的说明中，错误的是\_\_\_\_C\_\_\_。**

A．实参和形参占用不同的内存单元，即使同名也相互不影响

B．实参在进行函数调用时，它们都必须有确定的值，以便把这些值传给形参

C．实参对形参的数据传送是双向的，可以把实参的值传给形参，也可以把形参的值反向传给实参

D．形参变量只有在被调用时才分配内存单元

**8．C语言规定，简单变量做实参时，它和对应形参之间的数据传递方式是\_\_\_B\_\_\_\_。**

A．地址传递

B．单向值传递

C．由实参传递给形参，再由形参传回给实参

D．由用户指定传递方式

**9．关于函数调用的形式，以下错误的描述是\_\_\_\_D\_\_\_\_。**

A．可以出现在执行语句中　　 B．可以出现在一个表达式中

C．可以作为一个函数的实参 D．可以作为一个函数的形参

**10．C语言规定，函数返回值的类型是由\_\_\_\_D\_\_\_\_。**

A．return语句中的表达式类型所决定

B．调用该函数时的主调函数类型所决定

C．调用该函数时系统临时决定

D．在定义该函数所指定的函数类型所决定

**11．以下函数原型声明语句正确的是\_\_\_\_A\_\_\_\_。**

A．void f( int x ) ; B．void f( x ) ;

C．void f( int x ) D．void( int x ) ;

**12．若用一维数组名作为函数调用的实参，传递给形参的是\_\_\_\_A\_\_\_\_。**

A．数组首元素的地址 B．数组第一个元素的值

C．数组中全部元素的值 D．数组元素的个数

**14．以下不正确的说法是\_\_\_D\_\_\_\_。**

A．在不同函数中可以使用相同名字的变量

B．形式参数是局部变量

C．在函数内定义的变量只在本函数范围内有效

D．在函数内的复合语句中定义的变量在本函数范围内有效

**15．凡在函数中未指定存储类型的局部变量，其隐含的存储类型为\_\_\_A\_\_\_。**

A．自动( auto ) B．静态( static )

C．外部( extern ) D．寄存器(register)

**二、程序阅读题**

**1．下面程序的运行结果是\_\_\_\_max is 2\_\_\_\_。**

#include<stdio.h>

int main( )

{

int max( int x , int y ) ;

int a = 1 , b = 2 , c ;

c = max( a , b ) ;

printf( "max is %d\n" , c ) ;

return 0 ;

}

int max( int x , int y )

{

int z ;

z = ( x > y ) ? x : y ;

return z ;

}

**2．下面程序的运行结果是\_\_\_\_ a=1,b=2\_\_\_\_。**

#include<stdio.h>

int main( )

{

void f( int x , int y ) ;

int a = 1 , b = 2 ;

f( a , b ) ;

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

return 0 ;

}

void f( int x , int y )

{

x = 100 ;

y = 200 ;

}

**3．下面程序的运行结果是\_\_\_\_ 1 1 1\_\_\_\_。**

#include<stdio.h>

int main( )

{

void increment( ) ;

increment( ) ;

increment( ) ;

increment( ) ;

return 0 ;

}

void increment( )

{

int x = 0 ;

x += 1 ;

printf( "%d\t" , x ) ;

}

**4．下面程序的运行结果是\_\_\_\_ a=11,b=12,c=2\_\_\_\_。**

#include<stdio.h>

int main( )

{

int f1( int x , int y ) ;

int a = 11 , b = 12 ,c ;

c = f1( a , b ) ;

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

return 0 ;

}

int f1( int x , int y )

{

int f2( int , int ) ;

int c ;

x = x \* 2 ;

y = y \* 2 ;

c = f2( x , y ) ;

return c\*2 ;

}

int f2( int a , int b )

{

int c ;

c = ( a + b ) % 3 ;

return c ;

}

**6．下面程序的运行结果是\_\_\_\_7 8 9\_\_\_\_。**

#include<stdio.h>

int main( )

{

int f( int a ) ;

int a = 2 , i ;

for( i = 0 ; i < 3 ; i++ )

printf( "%4d" , f( a ) ) ;

return 0 ;

}

int f( int a )

{

int b = 0 ;

static int c = 3 ;

b++ ;

c++ ;

return ( a + b + c ) ;

}

**7．下面程序的运行结果是\_\_\_\_8 17\_\_\_\_。**

#include<stdio.h>

int main( )

{

int func( int a , int b ) ;

int k = 4 , m = 1 , p ;

p = func( k , m ) ;

printf( "%d\t" , p );

p = func( k , m ) ;

printf( "%d\n" , p );

return 0 ;

}

int func( int a , int b )

{

static int m = 0 , i = 2 ;

i += m + 1 ;

m = i + a + b ;

return m;

}

**8．下面程序的运行结果是\_\_\_\_0 1 2 0 1 2\_\_\_\_。**

#include<stdio.h>

int a[3] ;

int main( )

{

void f1( ) ;

void f2( ) ;

void print( ) ;

f1( ) ;

print( ) ;

f2( ) ;

print( ) ;

return 0 ;

}

void f1( )

{

int i ;

for ( i = 0 ; i < 3 ; i++ )

a[i] = i ;

}

void f2( )

{

int i ,a[3] ;

for ( i = 0 ; i < 3 ; i++ )

a[i] = i ;

}

void print( )

{

int i ;

for ( i = 0 ; i < 3 ; i++ )

printf( "%3d" , a[i] ) ;

}

**三、程序完善题**

**1．下面程序的功能是：area函数计算圆的面积，main函数中调用area函数，计算给定半径的圆的面积，输出时显示两位小数。请填空。**

#include<stdio.h>

#define PI 3.14

int main( )

{

**\_\_\_ float area ( float r )\_\_\_;**

float x = 2.4 , y ;

y = area( x ) ;

printf( "%6.2f\n" , y ) ;

return 0 ;

}

float area ( float r )

{

float s ;

s= PI \* r \* r ;

**\_\_\_return s\_\_\_\_\_;**

}

**2．下面程序的功能是：fun函数实现的计算，main函数中调用fun函数，计算的值。请填空。**

#include<stdio.h>

int main( )

{

double fun( double x , int y ) ;

int y = 3 ;

double x = 4 , z ;

**\_\_\_z=fun(x,y)\_\_\_;**

printf( "z=%f\n" , z ) ;

return 0 ;

}

double fun( double x , int y )

{

int i ;

double z = 1 ;

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

**\_\_z=z\*x;\_\_\_;**

return z ;

}

**3．下面程序的功能：有一个一维数组score，内放10个学生成绩，统计不及格的学生人数。请填空。**

#include <stdio.h>

int main( )

{

int fun( float array[10] ) ;

float score[10] ;

int i , count = 0 ;

printf( "input 10 scores:\n" ) ;

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

scanf( "%f" , &score[i] );

**\_\_\_count=fun(score)\_\_\_;**

printf( "%d\n" , count ) ;

return 0 ;

}

int fun( float array[10] )

{

int i , count = 0 ;

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

if( array[i] <60 )

**\_\_\_count++\_\_\_;**

return count ;

}

**第8章 编译预处理**

**一、选择题**

**1．编译预处理包括\_\_\_A\_\_\_\_。**

A．文件包含，宏定义和条件编译

B．构造工程文件

C．语句注释

D．编译源程序

**2．在宏定义#define PI 3.1415926中，用宏名PI代替一个\_\_\_D\_\_\_。**

A．常量 B．单精度数 C．双精度数 D．字符串

**3．以下有关宏替换的叙述不正确的是\_\_\_D\_\_\_\_。**

A．宏替换不占用运行时间 B．宏名无类型

C．宏替换只是字符替换 D．宏名必须用大写字母表示

**4．若程序中有#include<文件名>，则意味着\_\_\_A\_\_\_\_。**

A．将所指文件的全部内容，在此命令行出现的这一点上，插入源程序

B．指定标准输入输出

C．宏定义一个函数

D．条件编译说明

**5．在文件包含预处理语句的使用形式中，当#include后面的文件名用""（双引号）括起时，寻找被包含文件的方式是\_\_\_B\_\_\_\_。**

A．直接按系统设定的标准方式搜索目录

B．先在源程序所在目录搜索，再按系统设定的标准方式搜索

C．仅仅搜索源程序所在目录

D．仅仅搜索当前目录

**6．在文件包含预处理语句的使用形式中，当#include后面的文件名用<>（双引号）括起时，寻找被包含文件的方式是\_\_\_A\_\_\_\_。**

A．直接按系统设定的标准方式搜索目录

B．先在源程序所在目录搜索，再按系统设定的标准方式搜索

C．仅仅搜索源程序所在目录

D．仅仅搜索当前目录

**二、程序阅读题**

**1．下面程序的运行结果是\_\_\_6 ，18\_\_\_。**

#include<stdio.h>

#define M 3

#define N M + 1

#define NN N \* N / 2

int main( )

{

printf( "%d," , NN ) ;

printf( "%d" , 5 \* NN ) ;

return 0 ;

}

**第9章 指针**

**一、选择题**

**1．若已定义 int a = 5 ; 下面对 ①、② 两个语句的正确解释是**\_\_\_D\_\_\_**。**

① int \*p = &a ; ② \*p = a ;

A．语句①和②中的\*p含义相同，都表示给指针变量p赋值。

B．①和②语句的执行结果，都是把变量a的地址值赋给指针变量p。

C．①在对p进行说明的同时进行初始化，使p指向a；②将变量a的值赋给指针变量p。

D．①在对p进行说明的同时进行初始化，使p指向a；②将变量a的值赋为\*p。

**2．若需要建立如下图9-16所示的存储结构，且已有说明float \*p , m = 3.14 ; 则正确的赋值语句是**\_\_\_B\_\_\_\_**。**



图9-16

A．p = m ; B．p = &m ; C．\*p = m ; D．\*p = &m ;

**3．有如下语句 int a = 10 , b = 20 , \*p1 , \*p2 ; p1 = &a ; p2 = &b ; 如图9-17所示；若要实现图9-18所示的存储结构，可选用的赋值语句是**\_\_\_B\_\_\_\_**。**

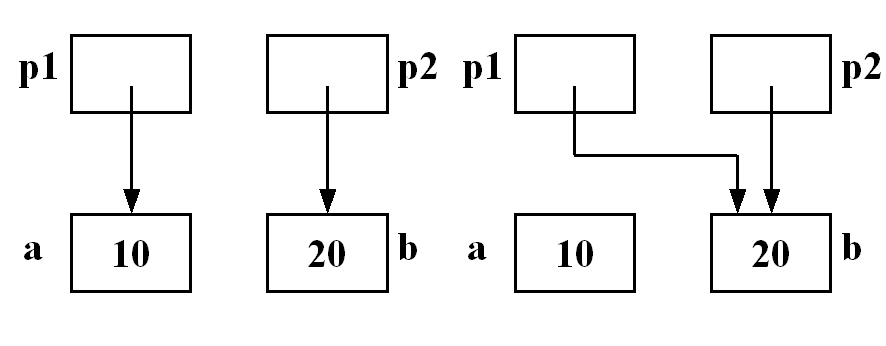


图9-17 图9-18

A．\*p1 = \*p2 ; B．p1 = p2 ; C．p1 = \*p2 ; D．\*p1 = p2 ;

**4．若有说明：int \*p , n ; 以不正确的程序段是**\_\_\_C\_\_\_\_**。**A?

A．scanf( "%d" , &n ) ; B． scanf( "%d" , &n ) ;

\*p = n ; p = &n ;

C．p = &n ; D． p = &n ;

scanf( "%d", &p ) ; scanf( "%d" , p ) ;

**5．下列选项中正确的语句组是**\_\_\_D\_\_\_\_**。**

A．char s[8] ; s = { "Beijing" } ; B．char \*s ; s = { "Beijing" } ;

C．char s[8] ; s = "Beijing" ; D．char \*s ; s = "Beijing" ;

**6．设char \*s = "\ta\017bc" ; 则指针变量s指向字符串所占的字节数是**\_\_C\_\_**。**

A．9 B．5 C．6 D．7

**7．下面程序段的运行结果是**\_\_\_C\_\_\_\_**。**

char a[ ] = "language" , \*p ;

p = a ;

while ( \*p != 'u' ) { printf( "%c" , \*p – 32 ) ; p++ ; }

A．LANGUAGE B．language C．LANG D．langUAGE

**8．若有定义：int a[5] ; 则a数组中首元素的地址可以表示为**\_\_\_C\_\_\_\_**。**

A．&a B．a + 1 C．a D．&a[1]

**9．有以下程序段**

int a[10] = {1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 } , \*p = &a[3] , b ;

b = p[5] ;

b中的值是\_\_\_D\_\_\_\_。

A．5 B．6 C．8 D．9

**12．阅读以下函数**

int fun( char \*sl , char \*s2 )

{

int i = 0;

while( sl[i] == s2[i] && s2[i] != '\0' ) i++ ;

return( sl[i] == '\0' && s2[i] == '\0' ) ;

}

此函数的功能是\_\_\_C\_\_\_\_。

A．将s2所指字符串赋给s1

B．比较s1和s2所指字符串的大小，若s1比s2的大，函数值为1，否则函数值为0

C．比较s1和s2所指字符串是否相等，若相等，函数值为1，否则函数值为0

D．比较s1和s2所指字符串的长度，若s1比s2的长，函数值为1，否则函数值为0

**二、填空题**

1．在C程序中，指针变量能够赋\_\_\_地址值\_\_\_值或\_\_0或NULL\_\_。

2．请填空：建立如下图9-19所示存储结构所需的说明语句是\_\_\_char a,\*p;\_\_\_。建立如下图9-19所示为变量a输入数据的输入语句是\_\_\_ scanf("%c", &a);\_\_\_\_。建立如下图9-19所示存储结构所需的赋值语句是\_\_\_p=&a\_\_\_。



图9-19

3．以下程序段通过移动指针变量m，将如下图9-20所示连续动态存储单元的值，从第一个元素起，输出到终端屏幕。请填空。（假设程序段中的所有变量均已正确说明）



图9-20

for( m = q ; m – q < 10 ; m++ ) printf( "%d, " , \_\_\_\*m\_\_\_\_) ;

printf( "\n" ) ;

4．以下程序段通过指针变量q，给如下图9-21所示连续动态存储单元赋值（在此过程中不能移动q）。请填空。（假设程序段中的所有变量均已正确说明）



图9-21

\_\_\_\_ for( k=0; k<10; k++ )\_\_\_ scanf( "%d" , q + k );

5．若有以下定义和语句，在程序中引用数组元素x[i]的4种形式是：\_\*(x+i)\_、\_p[i]\_、\_\*(p+i)\_和x[i]。（假设i已正确说明并赋值。）

int x[10] , \*p ;

p = x;

6．str\_len函数的功能是计算str所指字符串的长度，并作为函数值返回，请填空。

int str\_len( char \*str )

{

int i ;

for( i = 0 ; \_\_\_str[i]!=’\0’\_\_\_\_ != '\0' ; i++ ) ;

return(\_\_\_i\_\_\_\_ ) ;

}

**三、程序阅读题**

**1．下面程序的运行结果是**\_\_\_gae\_\_\_\_**。**

#include<stdio.h>

int main( )

{

char a[ ] = "Language" , b[ ] = "Programe" ;

char \*p1 , \*p2 ;

int k ;

p1 = a ;

p2 = b ;

for( k = 0 ; k <= 7 ; k++ )

if( \*( p1 + k ) == \*( p2 + k ) )

printf("%c" , \*( p1 + k ) ) ;

return 0 ;

}

**2．以下程序的运行结果是**\_\_\_ bcdABCD \_\_\_\_**。**

#include<stdio.h>

#include<string.h>

int main( )

{

char s[20] = "abcd" ;

char \*sp = s ;

sp++ ;

puts( strcat( sp , "ABCD" ) ) ;

return 0 ;

}

**4．下面程序的运行结果是**\_\_\_8 4\_\_\_**。**

#include<stdio.h>

void f( int y , int \*x )

{

y = y + \*x ;

\*x = \*x + y ;

}

int main( )

{

int x = 2 , y = 4 ;

f( y , &x ) ;

printf( "%d %d\n" , x , y ) ;

return 0 ;

}

3 1

**5．下面程序的运行结果是**\_\_\_4 2\_\_\_**。**

#include<stdio.h>

void fun( int \*a , int \*b )

{

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

\*a = 2 ; \*b = 4 ;

}

int main( )

{

int x = 1 , y = 3 ;

fun( &y , &x ) ;

printf( "%d %d\n" , x , y ) ;

return 0 ;

}

**四、程序完善题**

**2．下面程序的功能是:将一个整数字符串转换为一个整数，例如将字符串“-1234”转换为数值-1234。请填空。**

#include<stdio.h>

#include<string.h>

int chnum( char \*p )

{

int num = 0 , k , len , j ;

len = strlen( p ) ;

for( ; \_\_\_\*p!=’\0’\_\_\_; p++ )

{

k = \_\_\_\*p-‘0’\_\_\_ ;

j = ( --len ) ;

while( j-- > 0 ) { k = k \* 10 ; }

num = num + k ;

}

return ( num );

}

int main( )

{

char s[6]; int n ;

gets( s ) ;

if( \*s == '-' ) n = -chnum( s + 1 ) ;

else n = chnum( s ) ;

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

return 0 ;

}

**3．下面程序的功能是：找出数组中的最大值和该值所在的元素下标，数组元素从键盘输入。请填空。**

#include<stdio.h>

int main( )

{

int x[10] , \*p1 , \*p2 , k ;

for( k = 0 ; k < 10 ; k++ ) scanf( "%d" , x + k ) ;

for( p1 = x , p2 = x ; p1 - x < 10 ; p1++ )

if( \*p1 > \*p2 )

p2 = \_\_\_p1\_\_\_\_ ;

printf( "MAX = %d , INDEX = %d\n" , \*p2 , \_\_\_p2-x\_\_\_) ;

return 0 ;

}

**第10章 结构体与共用体**

**一、选择题**

**1．下面对结构数据类型的错误叙述是**\_\_\_C\_\_\_\_**。**

A．结构变量可以在说明结构类型后定义，也可在说明结构时定义。

B．结构可由若干个成员组成，各成员的数据类型可以不同。

C．定义一个结构类型后，编译程序要为结构的各成员分配存储空间。

D．结构变量的各成员可通过结构变量名和指向结构变量的指针引用。

**2．在VC6.0下，以下程序运行的结果是**\_\_\_B\_\_\_\_**。**

#include<stdio.h>

int main( )

{

union U

{

char st[4] ;

int i ;

long l ;

} ;

struct A

{

int c ;

union U u ;

} a ;

printf( "%d\n" , sizeof( struct A ) ) ;

return 0 ;

}

A．6 B．8 C．10 D．12

**3．说明语句 enum money { Fen , Jiao ,Yuan = 100 , Tenyuan } ; 每个枚举元素所对应的实际值为**\_\_\_A\_\_\_\_**。**

A．0 ，1 ，100 ，101

B．1 ，2 ，100 ，101

C．"Fen"或1 ，"Jiao"或2 ，"Yuan"或100 ，"Tenyuan"或101

D．"Fen" ，"Jiao" ，"Yuan"或100 ，" Tenyuan"

**4．当说明一个共用体变量时系统分配给它的内存是**\_\_\_C\_\_\_\_**。**

A．各成员所需内存量的总和

B．结构中第一个成员所需内存量

C．成员中占内存量最大者所需的容量

D．结构中最后一个成员所需内存量

**5．设有以下说明语句**

struct stu

{

int a ;

float b ;

} stutype ;

**则下面的叙述不正确的是**\_\_\_C\_\_\_\_**。**

A．struct是结构体类型的关键字。

B．structs stu是用户定义的结构体类型。

C．stutype是用户定义的结构体类型名。

D．a和b都是结构体成员名。

**6．以下scanf函数调用语句中对结构体变量成员的不正确引用是**\_\_\_D\_\_\_\_**。**

struct pupil

{

char name[20] ;

int age ;

int sex ;

} pup[5] , \*p = pup ;

A．scanf( "%s" , pup[0].name ) ;

B．scanf( "%d" , &pup[0].age ) ;

C．scanf( "%d" , &( p－>sex ) ) ;

D．scanf( "%d" , ( p－>age ) ) ;

**7.以下对结构体变量的定义错误的是**\_\_\_D\_\_\_\_**。**



**8．根据下面的定义，能打印出字母z的语句是**\_\_\_D\_\_\_\_**。**

struct student

{

char name[10] ;

int age ;

} ;

struct student stu[3] = { "li xiang" , 17 , "zhang san" , 18 ,"wang wu" , 18 } ;

A．printf( "%c\n" , stu[2].name ) ; B．printf( "%c\n" , stu[2].name[1] ) ;

C．printf( "%c\n" , stu[1].name[1] ) ; D．printf( "%c\n" , stu[1].name[0] ) ;

**9．设有如下说明**

typedef struct ST

{

long a ;

int b ;

char c[2] ;

} NEW ;

**则下面叙述中正确的是**\_\_\_C\_\_\_\_**。**

A．以上的说明形式非法 B．ST是一个结构体类型

C．NEW是一个结构体类型 D．NEW是一个结构体变量

**11．typedef unsigned long LONG ;的作用是**\_\_\_D\_\_\_\_**。**

A．定义了一种新的数据类型 B．定义了一个整型变量

C．定义了一个长整形变量 D．定义了一个新的数据类型标识符

**二、填空题**

1．在VC6.0中，若有如下定义：

struct data

{

char ch ;

short int i ;

double f ;

} b ;

则结构变量b占用内存的字节数是\_\_\_16\_\_\_。

**三、程序阅读题**

**1．下面程序的运行结果是**\_\_\_51 ，60 ，21\_\_\_**。**

#include<stdio.h>

struct stu

{

int x ;

int \*y ;

} \*p ;

int dt[4] = {10 , 20 , 30 , 40 } ;

struct stu a[4] = { 50 , &dt[0] , 60 , &dt[1] , 70 , &dt[2] , 80 , &dt[3] } ;

int main( )

{

p = a ;

printf( "%d," , ++p->x ) ;

printf( "%d," , ( ++p )->x ) ;

printf( "%d\n" , ++( \*p->y ) ) ;

return 0 ;

}

**2．下面程序的运行结果是**\_\_\_16\_\_\_**。**

#include<stdio.h>

typedef struct

{

int num ;

double s ;

}REC;

void fun1( REC x )

{

x.num = 23 ; x.s = 88.5 ;

}

int main( )

{

REC a = { 16 , 90.0 } ;

fun1( a ) ;

printf( "%d\n" , a.num ) ;

return 0 ;

}

**3．下面程序的运行结果是**\_\_\_1001 ，ChangRong ，1098.0\_\_\_**。**

#include<stdio.h>

#include<string.h>

struct A

{

int a ; char b[10] ; double c ;

};

void f( struct A \*t ) ;

int main( )

{

struct A a = {1001 , "zhangDa" , 1098.0 } ;

f( &a ) ;

printf( "%d,%s,%6.1f\n" , a.a , a.b , a.c ) ;

return 0 ;

}

void f( struct A \*t )

{

strcpy( t->b , "ChangRong" ) ;

}

**四、程序完善题**

**1．下面程序的功能是用以输出结构体类型struct ps所占内存单元的字节数。请填空。**

#include<stdio.h>

struct ps

{

double i ;

char arr[20] ;

} ;

int main( )

{

struct ps bt ;

printf( "bt size : %d\n" , \_\_\_sizeof(struct ps)\_\_\_) ;

return 0 ;

}

**3．下面程序中，函数 fun 的功能是：统计 person 所指结构体数组中所有性别(sex)为 M 的记录的个数，存入变量 n中，并做为函数值返回。请填空。**

#include<stdio.h>

#define N 3

typedef struct

{

int num ;

char nam[ 10 ] ;

char sex ;

}SS ;

int fun( SS person[ ] )

{

int i , n = 0 ;

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

if(\_\_\_person[i].sex\_\_\_\_ == 'M' )

n++ ;

return n ;

}

int main( )

{

SS W[N] = { {1 , "AA" , 'F' } , {2 , "BB" , 'M'} , { 3 , "CC" , 'M' } };

int n ;

n = fun( W ) ;

printf( "n=%d\n" , n ) ;

return 0 ;

}

**第12章 文件操作**

**一、选择题**

**1．下列关于C语言数据文件的叙述中正确的是**\_\_\_D\_\_\_**。**

A．文件由ASCII码字符序列组成，C语言只能读写文本文件

B．文件由二进制数据序列组成，C语言只能读写二进制文件

C．文件由记录序列组成，可按数据的存放形式分为二进制文件和文本文件

D．文件由数据流形式组成，可按数据的存放形式分为二进制文件和文本文件

**2．以下叙述中错误的是**\_\_\_C\_\_\_\_**。**

A．C语言中对二进制文件的访问速度比文本文件快

B．C语言中，随机文件以二进制代码形式存储数据

C．语句FILE fp ;定义了一个名为fp的文件指针

D．C语言中的文本文件以ASCII码形式存储数据

**3．若要指定打开c盘上子目录myfile下的二进制文件test.bin，在调用函数fopen时，第一个参数的正确格式是**\_\_\_C\_\_\_\_**。**

A．"c:myfile\test.bin" B．"c:\myfile\\test.bin"

C．"c:\\myfile\\test.bin" D．"c:\myfile\test.bin"

**4．若执行fopen函数时发生错误，则函数的返回值是**\_\_\_B\_\_\_**。**

A．地址值 B．0 C．1 D．EOF

**8．正常执行文件关闭操作时，fclose函数的返回值是**\_\_\_C\_\_\_**。**

A．-1 B．TRUE C．0 D．1

**9．fscanf函数的正确调用形式是**\_\_\_D\_\_\_**。**

A．fscanf( fp , 格式字符串 , 输入表列 ) ;

B．fscanf( 格式字符串 , 输入表列 , fp ) ;

C．fscanf( 格式字符串 , 文件指针 , 输入表列 ) ;

D．fscanf( 文件指针 , 格式字符串 , 输入表列 ) ;

**10．C语言库函数fgets( str , n , fp )的功能是**\_\_\_B\_\_\_**。**

A．从文件指针fp指向的文件中读取长度为n的字符串存入str指向的内存

B．从文件指针fp指向的文件中读取长度不超过n-1的字符串存入str指向的内存

C．从文件指针fp指向的文件中读取n个字符串存入str指向的内存

D．从str读取至多n个字符到文件指针fp指向的文件中

**11．读取二进制文件的函数调用形式为：fread( buffer , size , count , fp ) ;其中参数buffer代表的是**\_\_\_C\_\_\_**。**

A．一个文件指针，指向待读取的文件

B．一个整形变量，代表待读取的数据的字节数

C．一个内存块的首地址，代表读入数据的存放地址

D．一个内存块的字节数

**12．阅读以下程序及对程序功能的描述，其中正确的描述是**\_\_\_C\_\_\_**。**

#include<stdio.h>

**#include<stdlib.h>**

int main( )

{

FILE \*in , \*out ;

char ch , infile[10] , outfile[10] ;

printf( "Enter the infile name: \n" ) ;

scanf( "%s" , infile ) ;

printf( "Enter the outfile name : \n" ) ;

scanf( "%s" , outfile ) ;

if( ( in = fopen( infile , "r" ) ) == NULL )

{

printf( "Cannot open infile! \n" ) ;

exit( 0 ) ;

}

if( ( out = fopen( outfile , "w" ) ) == NULL )

{

printf( "Cannot open outfile! \n" ) ;

exit( 0 ) ;

}

while( !feof( in ) )

fputc( fgetc( in ) , out ) ;

fclose( in ) ;

fclose( out ) ;

return 0 ;

}

A．将磁盘文件的信息在屏幕上显示

B．将两个磁盘文件合二为一

C．将一个磁盘的文件复制到另一个磁盘文件中

D．将两个磁盘文件合并，且在屏幕上输出

**13．设fp是指向某个文件的指针，且已读到文件末尾，则库函数feof( fp )的返回值为**\_\_\_C\_\_\_**。**

A．EOF B．-1 C．非零值 D．NULL

**二、填空题**

1．在C语言中，根据数据的组织形式，可分为\_文本文件\_文件和\_二进制文件\_文件。

2．若要在C源程序中定义fp为**指向文件的**指针变量，则应使用的定义形式是

\_\_\_FILE \*fp; \_\_\_；在该程序的开头需要加的命令行是\_\_\_# include <stdio.h>\_\_\_。

3．函数调用语句：fgets( buf , n , fp ) ;从fp指向的文件中读入\_\_\_n-1\_\_\_\_个字符放到buf字符数组中，函数返回值为\_\_\_buf\_\_\_\_。

4．若fp是指向某二进制文件的指针，且未指到文件末尾，则!feof( fp )的值是\_\_\_1\_\_\_\_。

**三、程序阅读题**

**2．下面程序的运行结果后，文件t1.dat中的内容是**\_\_\_end\_\_\_\_**。**

#include<stdio.h>

void WriteStr ( char \*fn , char \*str )

{

FILE \*fp ;

fp = fopen( fn , "w" ) ;

fputs( str , fp ) ;

fclose( fp ) ;

}

int main( )

{

WriteStr( "t1.dat" , "start" ) ;

WriteStr( "t1.dat" , "end" ) ;

return 0 ;

}

**4．下面程序的运行结果后，文件test.dat中的内容是**\_\_\_hello\_\_\_**。**

#include<stdio.h>

# include <string.h>

void fun( char \*fname , char \*st )

{

FILE \*myf ;

int i ;

myf = fopen( fname , "w" ) ;

for( i = 0 ; i < strlen( st ) ; i++ )

fputc( st[i] , myf ) ;

fclose( myf ) ;

}

int main( )

{

fun( "test.dat" , "new world" ) ;

fun( "test.dat" , "hello," ) ;

return 0 ;

}

**四、程序完善题**

**1．下面程序的功能是统计文件letter.txt中小写字母’c’的个数。请填空。**

#include<stdio.h>

# include <stdlib.h>

int main( )

{

char ch ;

long n = 0 ;

FILE \*fp ;

if( ( fp = fopen( "letter.txt" , "r" ) ) == NULL )

{

printf( "Cannot open file! \n" ) ;

exit( 0 ) ;

}

while(\_\_\_! feof (fp)\_\_\_\_)

{

ch =\_\_\_ fgetc (fp)\_\_\_ ;

if( ch == 'c' )

n++ ;

}

printf( "n=%ld\n" , n ) ;

fclose( fp ) ;

return 0 ;

}

**2．下面程序的功能是统计并输出文件num.dat中正整数、零和负数的个数（设文件num.dat中存放了一组整数）。请填空。**

#include<stdio.h>

int main( )

{

int p = 0 , n = 0 , z = 0 , temp ;

FILE \*fp ;

fp =\_\_\_ fopen ( ”num.dat” , ”r”)\_\_\_\_ ;

if( fp == NULL )

printf ( "Cannot open file! \n" ) ; **//???**

else

{

while ( !feof( fp ) )

{

fscanf(\_\_\_fp , ”%d” , &temp\_\_\_) ;

if( temp > 0 ) p++ ;

else if( temp < 0 ) n++ ;

else \_\_\_z++\_\_\_\_ ;

}

}

fclose( fp ) ;

printf( "positive=%d, negative=%d, zero=%d\n" , p , n , z ) ;

return 0 ;

}

**3．下面程序的功能是：由终端键盘输入字符，存放到文件fname中，用“!”结束输入，请填空。**

#include<stdio.h>

# include <stdlib.h>

int main( )

{

FILE \*fp ;

char ch , fname[10] ;

printf( "Input name of file: \n" ) ;

gets( fname ) ;

if( ( fp = fopen( fname, "w" ) ) == NULL )

{

printf( "Cannot open file! \n" ) ;

exit( 0 ) ;

}

printf( "Enter data : \n" ) ;

while(\_\_\_(ch=getchar())\_\_\_\_ != '!' )

fputc(\_\_\_ch , fp\_\_\_\_) ; //🡪

fclose( fp ) ;

}

**程序设计题**

**《C语言实验与等级考试指导》**

**1、第四章P8 3、 4**

**2、第五章P9 1 、2、 5、 7**

**3、第六章P11 1、 2、 11、 12**

**4、第七章P14 3、 4、 5**

**5、第九章P20 6**

**第四章P8**

**第3题 有如下分段函数，输入x的值，计算y的值。编写程序实现。（分段函数计算）**

**3x+5 ( x < 2 )**

**y= x2  (2<=x<12)**

**4x-7 (x>=12)**

# include<stdio.h>

void main( )

{

int x,y;

printf("Input x : ");

scanf("%d",&x);

if ( x < 2 ) y=3\*x+5;

else if ( x >= 2 && x < 12 ) y=x\*x;

else y=4\*x-7;

printf("x=%d y=%d \n", x,y );

}

**第四章P8**

**第4题 从键盘输入一个字符，如果是小写字母，则变换成大写字母输出；其他字符不变换，原样输出。编写程序实现。（英文大小写转换）**

# include <stdio.h>

void main( )

{

char ch;

ch=getchar();

if ( ch >='a' && ch <='z' ) ch=ch-32;

putchar(ch);

}

**第五章P9**

**第1题 若一个三位数等于其各位上数字的立方和，则称这个三位数为水仙花数。例如153是一上水仙花数，因为153=1\*1\*1+5\*5\*5+3\*3\*3。编写一个程序，输出所有的水仙花数。**

# include <stdio.h>

void main( )

{

int i,gw,sw,bw;

for ( i=100 ; i<1000 ; i++ )

{

gw=i%10;

sw=i/10%10;

bw=i/100;

if ( i==gw\*gw\*gw+sw\*sw\*sw+bw\*bw\*bw ) printf("%d \n",i);

}

}

**第五章**

**P9 第2题** 编写一个程序，统计500~1000之间**素数**的个数。

# include <stdio.h>

# include <math.h>

void main( )

{

int m,k,i,count=0;

for ( m=501 ; m<1000 ; m=m+2 )

{

k=sqrt(m);

for ( i=2 ; i<=k ; i++ )

if ( m%i==0 ) break;

if ( i>k ) count++;

}

printf("500~100之间素数的个数是: %d \n" , count );

}

**第五章P9**

**第5题** 编写一个程序，计算s=1 - 1/2 + 1/3 – 1/4 + 1/5 + … + 1/m，其中m由输入决定。**（多项式求和）**

# include <stdio.h>

void main( )

{

int i,m,f=1;

float s=0;

printf("输入 m : ");

scanf("%d", &m);

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

{

s = s + f\*1.0/i;

f = -f;

}

printf("s=%f \n", s);

}

**第五章P9**

**第7题** 编写一个程序，输出1000以内的所有**“完数”**。一个数和如果恰好等于它的因子之和，这个数就是“完数”。例如，6的因子为1、2、3，而6=1+2+3，因此6是“完数”。

# include <stdio.h>

void main( )

{

int i,j,s; //s用于统计每个数的因子和

for ( i=2 ; i<=1000 ; i++ )

{

s=0; //每个数的因子和都要初始化

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

if ( i%j==0 ) s=s+j;

if ( i==s ) printf( "%4d 是完数 \n" , i );

}

}**第六章P11**

**第1题** 从键盘输入5个整数到数组中，编写一个程序，用**冒泡排序算法**对该数组升序排列。

# include <stdio.h>

# define N 5

void main( )

{

int a[N],i,j,t;

printf("输入5个整数 : ");

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

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

for ( i=0 ; i<=N-2 ; i++ )

for ( j=0 ; j<=N-i-2 ; j++ )

if ( a[j]>a[j+1] ) { t=a[j]; a[j]=a[j+1]; a[j+1]=t; }

printf("升序排序后的数组为 : ");

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

printf("%4d",a[i]);

printf("\n");

}

**第六章P11**

第2题 从键盘输入5个整数到数组中，编写一个程序，用选择排序算法对该数组升序排列。

# include <stdio.h>

# define N 5

void main( )

{

int a[N],i,j,t,k;

printf("输入5个整数 : ");

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

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

for ( i=0 ; i<=N-2 ; i++ )

{

k=i;

for ( j=i+1 ; j<=N-1 ; j++ )

if ( a[k]>a[j] ) k=j;

if ( i!=k ) { t=a[i]; a[i]=a[k]; a[k]=t; }

}

printf("升序排序后的数组为 : ");

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

printf("%4d",a[i]);

printf("\n");

}

**第六章P11**

**第11题 编写一个程序，将两个字符串拼接起来。不能使用strcat函数。**

# include <stdio.h>

void main( )

{

char str1[20],str2[20];

int i,j;

printf("输入两个字符串 : \n");

gets(str1);

gets(str2);

i=0;

while ( str1[i]!='\0' ) i++; //找到串1 '\0' 位置

j=0;

while ( str2[j]!='\0' ) //串2 拼接到串1 后

{

str1[i]=str2[j];

i++;

j++;

}

str1[i]='\0'; //最后加上 '\0'

printf("%s \n" , str1); //输出拼接后的串

}

**第六章P11**

**第12题** 编写一个程序，计算**字符串的长度**。不能使用strlen函数。

# include <stdio.h>

void main( )

{

char str[20];

int i;

printf("输入一个字符串 : \n");

gets(str);

i=0;

while ( str[i]!='\0' ) i++; //找到串1 '\0' 位置，串长及'\0'的下标

printf("串 %s 的长度是 : %d \n" , str, i );

}**第七章P14**

**第3题 编写两个函数，分别计算两个自然数的最大公约数和最小公倍数。在主函数中调用这两个函数并输出结果。两个自然数从键盘输入。**

# include <stdio.h>

int gys(int x,int y)

{

int min,i;

min= x<y ? x : y;

for ( i=min ; i>=1 ; i-- )

if ( x%i==0 && y%i==0 ) break;

return i;

}

int gbs(int x,int y)

{

return x\*y/gys(x,y);

}

void main( )

{

int m,n;

printf("输入两个正整数 : ");

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

printf("%d 和 %d 的最大公约数和最小公倍数分别是 : %d %d \n" ,

m,n,gys(m,n), gbs(m,n) );

}**第七章P14**

**第4题** 编写一个函数，使用**选择排序算法**对数组元素进行升序排序。在主函数中调用这个函数并输出结果。数组元素从键盘输入。

# include <stdio.h>

void sort(int b[10],int n)

{

int i,j,t,k;

for ( i=0 ; i<=n-2 ; i++ )

{

k=i;

for ( j=i+1 ; j<=n-1 ; j++ )

if ( b[k]>b[j] ) k=j;

if ( k!=i ) { t=b[k]; b[k]=b[i]; b[i]=t; }

}

}

void main( )

{

int a[10],i;

printf("输入10个整数 : ");

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

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

sort(a,10);

printf("升序排序的数组为 : ");

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

printf("%4d",a[i]);

printf("\n");

}**第七章P14**

**第5题** 编写两个函数，分别寻找N个整数中的**最大数和最小数**。在主函数中调用这两个函数并输出结果。N个整数从键盘输入。

# include <stdio.h>

# define N 10

int max(int b[N],int n)

{

int i,m;

m=b[0]; //假定b[0]最大

for ( i=1 ; i<=n-1 ; i++ )

if ( b[i]>m ) m=b[i];

return m;

}

int min(int b[N],int n)

{

int i,m;

m=b[0]; //假定b[0]最小

for ( i=1 ; i<=n-1 ; i++ )

if ( b[i]<m ) m=b[i];

return m;

}

void main( )

{

int a[N],i;

printf("输入 %d 个整数 : ",N);

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

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

printf("最大数=%d 最小数=%d \n" , max(a,N),min(a,N));

}**第九章P20**

**第6题** 编写一个函数，用**指针的方法**判断字符串是否**回文**，（正读和反读都相同的字符序列为回文，如”abcba”，“123321”是回文）。在主函数中调用这个函数并输出结果，字符串从键盘输入。

# include <stdio.h>

# include <string.h>

int hw(char s[10])

{

char \*p1,\*p2;

p1=s; //p1记录字符串第1个字符的地址

p2=s+strlen(s)-1; //p2记录字符串最后1个字符的地址

while ( p1<p2 )

{

if ( \*p1==\*p2 ) { p1++; p2--; }

else break;

}

if ( p1<p2 ) return 0;

else return 1;

}

void main( )

{

char str[10];

printf("输入一个字符串 : ");

gets(str);

if ( hw(str)==1 ) printf("字符串 %s 是回文 \n",str);

else printf("字符串 %s 不是回文 \n",str);

}