

## 一、选择题

- (1) A) 对长度为  $n$  的有序链表进行查找，最坏情况下需要的比较次数为  $n$   
 B) 对长度为  $n$  的有序链表进行对分查找，最坏情况下需要的比较次数为  $(n/2)$   
 C) 对长度为  $n$  的有序链表进行对分查找，最坏情况下需要的比较次数为  $(\log_2 n)$   
 D) 对长度为  $n$  的有序链表进行对分查找，最坏情况下需要的比较次数为  $(\log_2 n)$
- (2) 算法的时间复杂度是指  
 A) 算法的执行时间  
 B) 算法所处理的数据量  
 C) 算法程序中的语句或指令条数  
 D) 算法在执行过程中所需要的基本运算次数
- (3) 软件按功能可以分为：应用软件、系统软件和支撑软件（或工具软件）。下面属于系统软件的是  
 A) 编辑软件  
 B) 操作系统  
 C) 教务管理系统  
 D) 浏览器
- (4) 软件（程序）调试的任务是  
 A) 诊断和改正程序中的错误  
 B) 尽可能多地发现程序中的错误  
 C) 发现并改正程序中的所有错误  
 D) 确定程序中错误的性质
- (5) 数据流程图（DFD图）是  
 A) 软件概要设计的工具  
 B) 软件详细设计的工具  
 C) 结构化方法的需求分析工具  
 D) 面向对象方法的需求分析工具
- (6) 软件生命周期可分为定义阶段，开发阶段和维护阶段。详细设计属于  
 A) 定义阶段  
 B) 开发阶段  
 C) 维护阶段  
 D) 上述三个阶段
- (7) 数据库管理系统中负责数据模式定义的语言是  
 A) 数据定义语言  
 B) 数据管理语言  
 C) 数据操纵语言  
 D) 数据控制语言
- (8) 在学生管理的关系数据库中，存取一个学生信息的数据单位是  
 A) 文件  
 B) 数据库  
 C) 字段  
 D) 记录
- (9) 数据库设计中，用 E - R 图来描述信息结构但不涉及信息在计算机中的表示，它属于数据库设计的  
 A) 需求分析阶段  
 B) 逻辑设计阶段  
 C) 概念设计阶段  
 D) 物理设计阶段
- (10) 有两个关系 R 和 T 如下：

R			T		
A	B	C	A	B	C
a	1	2			
b	2	2	c	3	2
c	3	2	d	3	2
d	3	2			

则由关系 R 得到关系 T 的操作是

- A) 选择  
 B) 投影  
 C) 交  
 D) 并
- (11) 以下叙述正确的是  
 A) C 语言程序是由过程和函数组成的  
 B) C 语言函数可以嵌套调用，例如：fun ( fun ( x ) )  
 C) C 语言函数不可以单独编译  
 D) C 语言中除了 main 函数，其他函数不可作为单独文件形式存在
- (12) 以下关于 C 语言的叙述中正确的是  
 A) C 语言中的注释不可以夹在变量名或关键字的中间  
 B) C 语言中的变量可以在使用之前的任何位置进行定义  
 C) 在 C 语言算术表达式的书写中，运算符两侧的运算数类型必须一致  
 D) C 语言的数值常量中夹带空格不影响常量值的正确表示

(13) 以下C语言用户标识符中，不合法的是

- A) \_1                  B) A a B c                  C) a\_\_b                  D) a - - b

(14) 若有定义：double a = 2.2; int i = 0, k = 18; , 则不符合C语言规定的赋值语句是

- A) a = a + + , i + + ;                  B) i = ( a + k ) < = ( i + k ) ;  
C) i = a % 1 1 ;                  D) i = ! a ;

(15) 有以下程序

```
#include<stdio.h>
```

```
main()
```

```
{char a,b,c,d;
```

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

```
c=getchar();d=getchar;
```

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

当执行程序时，按下列方式输入数据（从第1列开始，< C R >代表回车，注意：回车也是一个字符）

1 2 < C R >

3 4 < C R >

则输出结果是

- A) 1 2 3 4                  B) 1 2                  C) 1 2                  D) 1 2

(16) 以下关于C语言数据类型使用的叙述中错误的是

- A) 若要准确无误差的表示自然数，应使用整数类型  
B) 若要保存带有多位小数的数据，应使用双精度类型  
C) 若要处理如“人员信息”等含有不同类型的相关数据，应自定义结构体类型  
D) 若只处理“真”和“假”两种逻辑值，应使用逻辑类型

(17) 若a是数值类型，则逻辑表达式 ( a == 1 ) || ( a != 1 ) 的值是

- A) 1                          B) 0  
C) 2                          D) 不知道a的值，不能确定

(18) 以下选项中与 if ( a==1) a=b; else a++ ; 语句功能不同的switch 语句是

A) switch ( a )  
{ case 1 : a = b ; break ;  
default : a++;  
}

B) switch ( a==1)  
{ case 0 : a = b ; break ;  
case 1 : a++;  
}

C) switch ( a )  
{ default : a++; break ;  
case 1 : a = b ;  
}

D) switch ( a==1)  
{ case 1 : a = b ; break ;  
case 0 : a++;  
}

( 19) 如下嵌套的 i f 语句

```
if(a<b)
    if(a<c) k=a;
    else k=c;
if(b<c) k=b;
    else k=c;
```

以下选项中与上述 i f 语句等价的语句是

- A ) k=(a<b)?a:b ; k= ( b<c ) ?b:c;
- B ) k=(a<b) ?((b < c ) ? a:b ) ((b > c)?b:c) ;
- C ) k=(a<b) ?((a < c ) ? a:c ) ((b < c)?b:c) ;
- D ) k=(a<b) ?a:b ; k= ( a<c ) ?a:c ;

( 20) 有以下程序

```
#include<stdio.h>
main()
{int i,j,m=1;
 for(i=1;i<3;i++)
 {for(j=3;j>0;j--)
 {if(i+j>3) break;
 m*=i*j;}}
printf( " m=%d\n",m);
} 程序运行后的输出结果是
```

- A ) m = 6
- B ) m = 2
- C ) m = 4
- D ) m = 5

( 21) 有以下程序

```
#include<stdio.h>
main()
{int a=1,b=2;
 for(;a<8;a++) { b+=a;a+=2;}
printf( " %d,%d\n",a,b);
} 程序运行后的输出结果是
```

- A ) 9 , 1 8
- B ) 8 , 1 1
- C ) 7 , 1 1
- D ) 1 0 , 1 4

( 22) 有以下程序，其中 k 的初值为八进制数

```
#include<stdio.h>
main()
{int k=011;
printf( " %d\n",k++);
} 程序运行后的输出结果是
```

- A ) 1 2
- B ) 1 1
- C ) 1 0
- D ) 9

( 23) 下列语句组中，正确的是

- A ) char \*s ; s= " Olympic " ;
- B ) char s[7] ; s= " Olympic " ;
- C ) char \*s ; s= { " Olympic " } ;
- D ) char s[7] ; s={ " Olympic " } ;

( 24) 以下关于 return 语句的叙述中正确的是

- A ) 一个自定义函数中必须有一条 return 语句
- B ) 一个自定义函数中可以根据不同情况设置多条 return 语句
- C ) 定义成 void 类型的函数中可以有带返回值的 return 语句
- D ) 没有 return 语句的自定义函数在执行结束时不能返回到调用处

(25) 下列选项中，能正确定义数组的语句是

- A ) int num[0..2008] ;                      B ) int num[] ;  
C ) int N=2008 ;                              D ) #define N 2008  
int num[N] ;                      int num[N] ;

(26) 有以下程序

```
#include<stdio.h>
void fun(char *c,int d)
{ *c=*c+1;d=d+1;
printf( " %c,%c" ,*c,d);
}
main()
{char b= ' a ',a= ' A ' ;
fun(&b,a);printf( " %c,%c\n" , b,a);
} 程序运行后的输出结果是
```

- A ) b , B , b , A                      B ) b , B , B , A  
C ) a , B , B , a                      D ) a , B , a , B

(27) 若有定义 int ( \*pt ) [3] ; , 则下列说法正确的是

- A ) 定义了基类型为 int 的三个指针变量  
B ) 定义了基类型为 int 的具有三个元素的指针数组 pt  
C ) 定义了一个名为 \*pt、具有三个元素的整型数组  
D ) 定义了一个名为 pt 的指针变量，它可以指向每行有三个整数元素的二维数组

(28) 设有定义 double a[10],\*s = a ; , 以下能够代表数组元素 a[3] 的是

- A ) (\*s ) [3]                      B ) \* ( s+3)  
C ) \*s[3]                      D ) \*s+3

(29) 有以下程序

```
#include<stdio.h>
main()
{int a[5]={1,2,3,4,5},b[5]={0,2,1,3,0},i,s=0;
for(i=1;i<3;i++) s=s+a[b[i]];
printf( " %d\n" ,s);
} 程序运行后的输出结果是
```

- A ) 6                      B ) 1 0                      C ) 1 1                      D ) 1 5

(30) 有以下程序

```
#include<stdio.h>
main()
{int b[3][3]={0,1,2,0,1,2,0,1,2},i,j,t=1;
for(i=1;i<3;i++)
for(j=1;j<=1;j++) t+=b[i][b[j]][i];
printf( " %d\n" ,t);
} 程序运行后的输出结果是
```

- A ) 1                      B ) 3                      C ) 4                      D ) 9

(31) 若有以下定义和语句

```
char s1[10] = " abcd! " , s2=" " ;
printf ( " % d % d \ n " , strlen ( s1 ) , strlen ( s2 ) ) ;
则输出结果是
```

- A ) 5 5                      B ) 1 0 5                      C ) 1 0 7                      D ) 5 8

( 32) 有以下程序

```
#include<stdio.h>
#define N 8
void fun(int *x,int i)
{ *x=*x+i;}
main()
{int a[N]={1,2,3,4,5,6,7,8},i;
fun(a,2); for(i=1;i<N/2;i++)
{printf(    “ %d” ,a[i]);}
printf(    “ \n ” );
} 程序运行后的输出结果是
```

A ) 1 3 1 3

B ) 2 2 3 4

C ) 3 2 3 4

D ) 1 2 3 4

( 33) 有以下程序

```
#include<stdio.h>
int f(int t[],int n);
main()
{int a[4]={1,2,3,4},s;
s=f(a,4);printf(    “ %d\n” ,s);}
int f(int t[],int n)
{if (n>0)return t[n-1]+f(t,n-1);
else return 0;} 程序运行后的输出结果是
```

A ) 4

B ) 1 0

C ) 1 4

D ) 6

( 34) 有以下程序

```
#include<stdio.h>
int fun()
{static int x=1;
x*=2;return x;
}
main()
{int i,s=1;
for(i=1;i<=2;i++) s=fun();
printf(    “ %d\n” ,s);
} 程序运行后的输出结果是
```

A ) 0

B ) 1

C ) 4

D ) 8

( 35) 有以下程序

```
#include<stdio.h>
#define SUB(a) (a)-(a)
main()
{int a=2,b=3,c=5,d;
d=SUB(a+b)*c;
printf(    “ %d\n” ,d);
} 程序运行后的输出结果是
```

A ) 0

B ) - 1 2

C ) - 2 0

D ) 1 0

( 36) 设有定义：

```
struct complex
{int real,unreal;} data1={1,8},data2;
```

则以下赋值语句中错误的是

A ) data2 = data1 ;

B ) data2 = ( 2 , 6 ) ;

C ) data2 . real = data1 . real ;

D ) data2 . real = data1 . unreal ;

( 37 ) 有以下程序

```
#include<stdio.h>
#include<string.h>
struct A
{int a;char b[10];double c;};
void f(struct At);
main()
{struct Aa={1001, " ZhangDa",1098.0};f(a);
printf( " %d,%s,%6.1f\n" ,a.a,a.b,a.c);}
void f(struct At)
{t.a=1002;strcpy(t.b, " ChangRong");t.c=1202.0;}
```

输出结果是

A ) 1 0 0 1 , Z h a n g D a , 1 0 9 8 . 0

B ) 1 0 0 2 , C h a n g R o n g , 1 2 0 2 . 0

C ) 1 0 0 1 , C h a n g R o n g , 1 0 9 8 0

D ) 1 0 0 2 , Z h a n g D a , 1 2 0 2 . 0

( 38 ) 有以下定义和语句

```
struct workers
{int num;char name[20];char c;
  struct
  {int day;int month;int year;}s
};
struct workers w,*pw;
pw=&w;
```

能给w中year成员赋1980的语句是

A ) p w . y e a r = 1 9 8 0 ;

B ) w . y e a r = 1 9 8 0 ;

C ) p w - > y e a r = 1 9 8 0 ; D ) w . s . y e a r = 1 9 8 0 ;

( 39 ) 有以下程序

```
#include<stdio.h>
main()
{int a=2,b=a,c=2;
printf( " %d\n" ,a/b&c);
} 程序运行后的输出结果是
```

A ) 0

B ) 1

C ) 2

D ) 3

( 40 ) 有以下程序

```
#include<stdio.h>
main()
{FILE *fp;char str[10];
fp=open( " myfile.dat " , " w " );
fputs( " abc " ,pf);close(pf);
fp=open( " myfile.dat " , " a+ " );
fprintf(pf, " %d" ,28);
rewind(pf);
```

```
fscanf(pf, " %s",str);puts(str);
close(pf);
}程序运行后的输出结果是
```

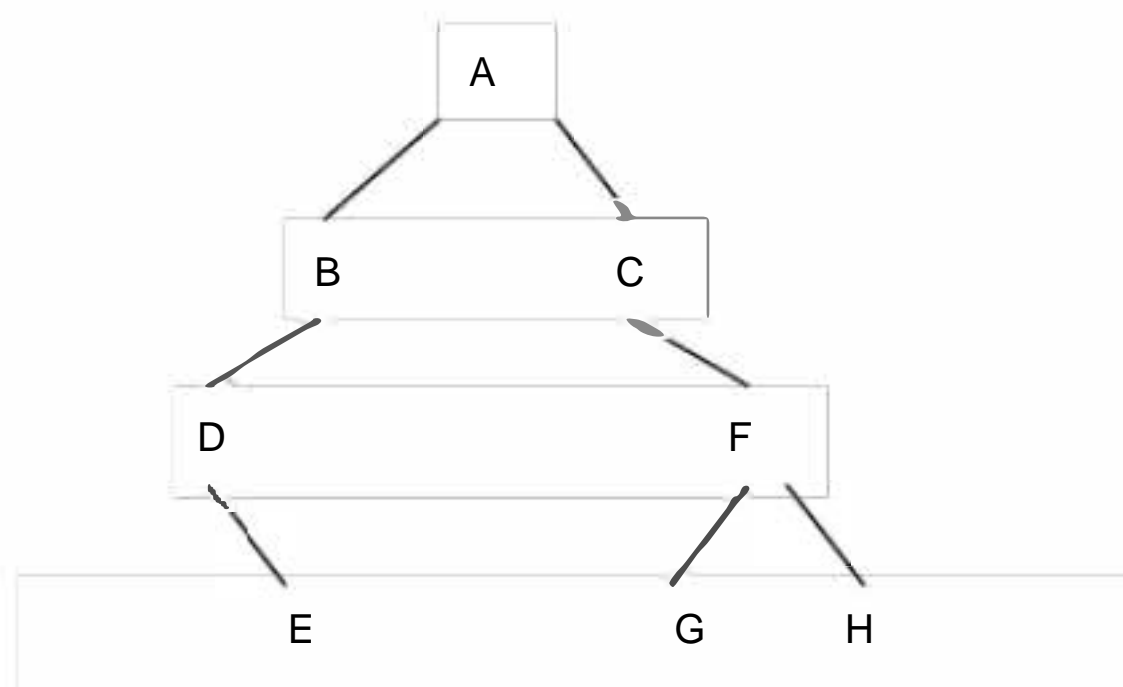
- A ) a b c                      B ) 2 8 c  
C ) a b c 2 8                  D ) 因类型不一致而出错

## 二、填空题

( 1 ) 一个队列的初始状态为空。现将元素 A , B , C , D , E , F , 5 , 4 , 3 , 2 , 1 依次入队 , 然后再依次退队 , 则元素退队的顺序为 【 1 】 。

( 2 ) 设某循环队列的容量为 50 , 如果头指针 front = 45 ( 指向队头元素的前一位置 ) , 尾指针 rear = 10 ( 指向队尾元素 ) , 则该循环队列中共有 【 2 】 个元素。

( 3 ) 设二叉树如下 :



对该二叉树进行后序遍历的结果为 【 3 】 。

( 4 ) 软件是 【 4 】 数据和文档的集合。

( 5 ) 有一个学生选课的关系 , 其中学生的关系模式为 : 学生 ( 学号 , 姓名 , 班级 , 年龄 ) , 课程的关系模式为 : 课程 ( 课号 , 课程名 , 学时 ) , 其中两个关系模式的键分别是学号和课号 , 则关系模式选课可定义为 : 选课 ( 学号 , 【 5 】 , 成绩 ) 。

( 6 ) 设 x 为 int 型变量 , 请写出一个关系表达式 【 6 】 , 用以判断 x 同时为 3 和 7 的倍数时 , 关系表达式的值为真。

( 7 ) 有以下程序

```
#include<stdio.h>
main()
{int a=1,b=2,c=3,d=0;
if(a==1)
if(b!=2)
if(c==3) d=1;
else d=2;
else if(c!=3) d=3;
else d=4;
else d=5;
printf( " %d\n",d);
}程序运行后的输出结果是 【 7 】 。
```

( 8 ) 有以下程序

```
#include<stdio.h>
main()
{int m,n;
```

```
scanf( " %d%d",&m,&n);
while(m!=n)
{while(m>n) m=m-n;
 while(m<n) n=n-m;}
printf( " %d\n",m);
```

程序运行后，当输入14 63 <回车>时，输出结果是 【8】。

( 9 ) 有以下程序

```
#include<stdio.h>
main()
{int i,j,a[3][3]={1,2,3,4,5,6,7,8,9};
 for(i=1;i<3;i++)
 for(j=1;j<3;j++) printf( " %d",a[i][j]);
 printf( " \n " );
```

程序运行后的输出结果是 【9】

( 10 ) 有以下程序

```
#include<stdio.h>
main()
{int a[]={1,2,3,4,5,6},*k[3],i=0;
 while(i<3)
 {k[i]=&a[2*i];
 printf( " %d",*k[i]);
 i++;
 }
```

程序运行后的输出结果是 【10】。

( 11 ) 有以下程序

```
#include<stdio.h>
main()
{int a[3][3]={{1,2,3},{4,5,6},{7,8,9}};
 int b[3]={0},i;
 for(i=1;i<3;i++) b[i]=a[i][2]+a[2][i];
 for(i=1;i<3;i++)printf( " %d",b[i]);
 printf( " \n " );
```

程序运行后的输出结果是 【11】。

( 12 ) 有以下程序

```
#include<stdio.h>
#include<string.h>
void fun(char *str)
{char temp;int n,i;
 n=strlen(str);
 temp=str[n-1];
 for(i=n-1;i>0;i--) str[i]=str[i-1];str[0]=temp;
 }
main()
{char s[50];scanf( " %s",s);fun(s);printf( " %s\n",s);}
```

程序运行后输入：abcdef <回车>，则输出结果是 【12】

( 13 ) 以下程序的功能是：将值为三位正整数的变量x中的数值按照个位、十位、百位的顺序拆分并输出。请填空。



```
#include<stdio.h>
main()
{int x=256;
printf(    “ %d-%d-%d\n”, 【13】 , x/10%10, x/100);
}
```

( 1 4 ) 以下程序用以删除字符串中所有的空格，请填空。

```
#include<stdio.h>
main()
{char[100]={    “ Our teacher teach c language!    ”};int i,j;
for(i=j=0;s[i]!=    ‘ \0 ’ ;i++)
if(s[i]!=    ‘ ’ ) {s[j]=s[i];j++;}
s[j]=    【14】 ;
printf(    “ %s\n” ,s);
}
```

( 1 5 ) 以下程序的功能是：借助指针变量找出数组元素中的最大值及其元素的下标值。请填空。

```
#include<stdio.h>
main()
{int a[10],*p,*s;
for(p=a;p-a<10;p++) scanf(    “ %d” ,p);
for(p=a,s=a;p-a<10;p++) if(*p>*s) s=    【15】 ;
printf(    “ index=%d\n” s-a);
}
```

## 参考答案：

### 一、选择题：

1.C 2.D 3.B 4.A 5.C 6.B 7.A 8.D 9.C 10.A      11.B 12.A 13.D 14.C 15.C 16.D 17.A 18.B 19.C 20.A 21.D 22.D 23.A  
24.B 25.D 26.A 27.D 28.B 29.C 30.C  
31.A 32.C 33.B 34.C 35.B 36.B 37.A 38.D 39.A 40.C

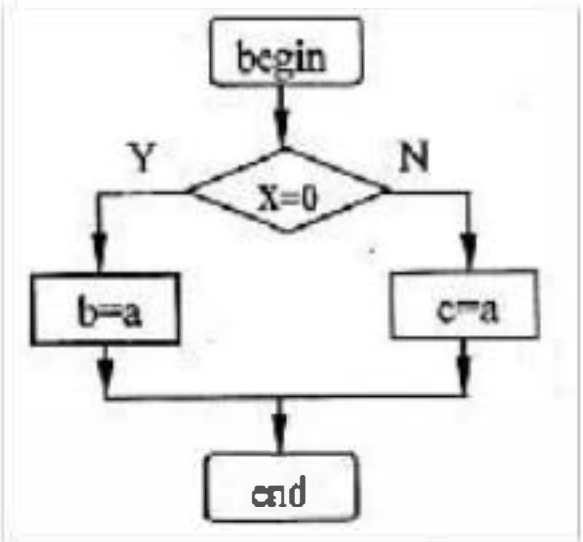
### 二、填空题：

(1)A B C D E F 5 4 3 2 1      (2)15      (3)EDBGHFCA      (4) 程序      (5) 课号  
(6)(x%3==0) && (x%7==0)      (7)4      (8)7      (9)123569      (10)135  
(11)101418      (12)abcde      (13)x%10      (14)'\0'      (15)p

2009年9月全国计算机等级考试二级C笔试试卷

一、选择题

- (1) 下列数据结构中，属于非线性结构的是  
A ) 循环队列 B) 带链队列 C) 二叉树 D ) 带链栈
- (2) 下列数据结果中，能够按照“先进后出”原则存取数据的是  
A) 循环队列 B) 栈 C) 队列 D) 二叉树
- (3) 对于循环队列，下列叙述中正确的是  
A ) 队头指针是固定不变的 B ) 队头指针一定大于队尾指针  
C ) 队头指针一定小于队尾指针 D ) 队头指针可以大于队尾指针，也可以小于队尾指针
- (4) 算法的空间复杂度是指  
A ) 算法在执行过程中所需要的计算机存储空间 B ) 算法所处理的数据量  
C ) 算法程序中的语句或指令条数 D ) 算法在执行过程中所需要的临时工作单元数
- (5) 软件设计中划分模块的一个准则是  
A) 低内聚低耦合 B) 高内聚低耦合 C) 低内聚高耦合 D) 高内聚高耦合
- (6) 下列选项中不属于结构化程序设计原则的是  
A) 可封装 B) 自顶向下 C) 模块化 D) 逐步求精
- (7) 软件详细设计产生的图如下：



- 该图是  
A) N-S 图 B) PAD 图 C) 程序流程图 D) E-R 图
- (8) 数据库管理系统是  
A ) 操作系统的一部分 B) 在操作系统支持下的系统软件  
C) 一种编译系统 D) 一种操作系统
- (9) 在 E-R图中，用来表示实体联系的图形是  
A) 椭圆图 B) 矩形 C) 菱形 D) 三角形
- (10) 有三个关系 R, S和 T如下：

R			S			T		
A	B	C	A	B	C	A	B	C
a	1	2	d	3	2	a	1	2
b	2	1				b	2	1
c	3	1				c	3	1
						d	3	2

- 其中关系 T 由关系 R和 S通过某种操作得到，该操作为  
A) 选择 B) 投影 C) 交 D) 并
- (11) 以下叙述中正确的是  
A) 程序设计的任务就是编写程序代码并上机调试 B ) 程序设计的任务就是确定所用数据结构

C) 程序设计的任务就是确定所用算法

D ) 以上三种说法都不完整

( 12) 以下选项中，能用作用户标识符的是

A ) void      B    ) 8\_8      C    ) \_0\_      D    ) unsigned

( 13) 阅读以下程序

```
#include
main()
{ int case; float printf;
  printf( " 请输入2个数: " );
  scanf( " %d %f" ,&case,&printf);
  printf( " %d %f" ,case,printf);
}
```

该程序编译时产生错误，其出错原因是

A) 定义语句出错，case 是关键字，不能用作用户自定义标识符

B) 定义语句出错，printf 不能用作用户自定义标识符

C) 定义语句无错，scanf 不能作为输入函数使用

D ) 定义语句无错，printf 不能输出 case 的值

( 14) 表达式：(int)((double)9/2)-(9)%2 的值是

A) 0      B    ) 3      C    ) 4      D    ) 5

( 15) 若有定义语句：int x=10; ，则表达式 x-=x+x 的值为

A) -20      B    ) -10      C    ) 0      D) 10

( 16) 有以下程序

```
#include
main()
{ int a=1,b=0;
  printf( " %d," ,b=a+b);
  printf( " %d" ,a=2*b);
}
```

程序运行后的输出结果是

A ) 0,0      B    ) 1,0      C) 3,2      D    ) 1,2

( 17 ) 设有定义：int a=1,b=2,c=3; ，以下语句中执行效果与其它三个不同的是

A) if(a>b) c=a,a=b,b=c;      B            ) if(a>b) {c=a,a=b,b=c;}

C) if(a>b) c=a;a=b;b=c;      D            ) if(a>b) {c=a;a=b;b=c;}

( 18) 有以下程序

```
#include
main()
{ int c=0,k;
  for (k=1;k<3;k++)
  switch (k)
  { default: c+=k
  case 2: c++;break;
  case 4: c+=2;break;
  }
  printf( " %d" ,c);
}
```

程序运行后的输出结果是

A) 3            B) 5      C    ) 7            D) 9

( 19 ) 以下程序段中 , 与语句 :  $k=a>b?(b>c?1:0):0$  ; 功能相同的是

- A)  $\text{if}((a>b)\&\&(b>c))\ k=1;$   
 $\text{else}\ k=0;$
- B)  $\text{if}((a>b)\|\|(b>c))\ k=1;$   
 $\text{else}\ k=0;$
- C)  $\text{if}(a\leq b)\ k=0;$   
 $\text{else}\ \text{if}(b\leq c)\ k=1;$
- D)  $\text{if}(a>b)\ k=1;$   
 $\text{else}\ \text{if}(b>c)\ k=1;$   
 $\text{else}\ k=0;$

(20 ) 有以下程序

```
#include
main()
{ char s[]={ " 012xy " };int i,n=0;
  for(i=0;s[i]!='\0';i++)
    if(s[i]>='a' && s[i]<='z' ) n++;
  printf( " %d\n",n);
}
```

程序运行后的输出结果是

- A) 0      B      ) 2      C      ) 3      D) 5

( 21 ) 有以下程序

```
#include
main()
{ int n=2,k=0;
  while(k++&& n++>2);
  printf( " %d %d\n",k,n);
}
```

程序运行后的输出结果是

- A) 0 2    B    ) 1 3      C) 5 7    D    ) 1 2

( 22 ) 有以下定义语句 , 编译时会出现编译错误的是

- A)  $\text{char}\ a= 'a' ;$     B      )  $\text{char}\ a= 'n' ;$
- C)  $\text{char}\ a= 'aa' ;$     D      )  $\text{char}\ a= 'x2d' ;$

( 23 ) 有以下程序

```
#include
main()
{ char c1,c2;
  c1= 'A' + '8' - '4' ;
  c2= 'A' + '8' - '5' ;
  printf( " %c,%c\n",c1,c2);
}
```

已知字母 A 的 ASCII 码为 65 , 程序运行后的输出结果是

- A) E,68      B      ) D,69      C) E,D      D) 输出无定值

( 24 ) 有以下程序

```
#include
void fun(int p)
{ int d=2;
```

```

    p=d++; printf(    “ %d” ,p);}
main()
{ int a=1;
  fun(a); printf(    “ %d” ,a);}

```

程序运行后的输出结果是

A) 32                  B) 12          C    ) 21          D        ) 22

( 25) 以下函数 findmax 拟实现在数组中查找最大值并作为函数值返回，但程序中有错导致不能实现预定功能

```

#define MIN -2147483647
int findmax (int x[],int n)
{ int i,max;
  for(i=0;i<N;i++)< p="" />
  { max=MIN;
  if(max
    return max;
  }

```

造成错误的原因是

A) 定义语句 int i,max; 中 max未赋初值  
 B) 赋值语句 max=MIN中，不应给 max赋 MIN值  
 C) 语句 if(max  
 D) 赋值语句 max=MIN放错了位置

( 26) 有以下程序

```

#include
main()
{ int m=1,n=2,*p=&m,*q=&n,*r;
  r=p;p=q;q=r;
  printf(    “ %d,%d,%d,%d” ,m,n,*p,*q);
}

```

程序运行后的输出结果是

A) 1,2,1,2                  B) 1,2,2,1          C    ) 2,1,2,1                  D) 2,1,1,2

( 27) 若有定义语句：int a[4][10],\*p,\*q[4]; 且 0 ≤ i<4，则错误的赋值是

A) p=a                  B        ) q[i]=a[i]          C        ) p=a[i]                  D        ) p=&a[2][1]

( 28) 有以下程序

```

#include
#include
main()
{ char str[ ][20]={    “ One*World” ,    “ One*Dream!” },*p=str[1];
  printf(    “ %d,” ,strlen(p));printf(    “ %s” ,p);
}

```

程序运行后的输出结果是

A) 9,One\*World          B        ) 9,One\*Dream  
 C) 10,One\*Dream          D        ) 10,One\*World

( 29) 有以下程序

```

#include
main()
{ int a[ ]={2,3,5,4},i;
  for(i=0;i<4;i++)

```

```

switch(i%2)
{ case 0:switch(a[i]%2)
{case 0:a[i]++;break;
case 1:a[i]--;
}break;
case 1:a[i]=0;
}
for(i=0;i<4;i++) printf(      “ %d” ,a[i]); printf(      “ ” );
}

```

A) 3 3 4 4                      B) 2 0 5 0                      C)                      ) 3 0 4 0                      D) 0 3 0 4

( 30) 有以下程序

```

#include
#include
main()
{ char a[10]= “ abcd ” ;
printf( “ %d,%d ” ,strlen(a),sizeof(a));
}

```

程序运行后的输出结果是

A) 7,4                      B)                      ) 4,10                      C) 8,8                      D) 10,10

( 31) 下面是有关 C语言字符数组的描述，其中错误的是

A) 不可以用赋值语句给字符数组名赋字符串                      B) 可以用输入语句把字符串整体输入给字符数组  
C) 字符数组中的内容不一定是字符串                      D) 字符数组只能存放字符串

( 32) 下列函数的功能是

```

fun(char * a,char * b)
{ while((*b=*a)!= ‘ 0 ’ ) {a++,b++;} }

```

A) 将 a 所指字符串赋给 b 所指空间                      B) 使指针 b 指向 a 所指字符串  
C) 将 a 所指字符串和 b 所指字符串进行比较                      D) 检查 a 和 b 所指字符串中是否有 ‘ 0 ’

( 33) 设有以下函数

```

void fun(int n,char * s) {,,}

```

则下面对函数指针的定义和赋值均是正确的是

A) void (\*pf)(); pf=fun;                      B) void \*pf(); pf=fun;  
C) void \*pf(); \*pf=fun;                      D) void (\*pf)(int,char);pf=&fun;

( 34) 有以下程序

```

#include
int f(int n);
main()
{ int a=3,s;
s=f(a);s=s+f(a);printf( “ %d” ,s);
}
int f(int n)
{ static int a=1;
n+=a++;
return n;
}

```

程序运行以后的输出结果是

A) 7                      B) 8                      C) 9                      D) 10

( 35) 有以下程序

```
#include
#define f(x) x*x*x
main()
{ int a=3,s,t;
  s=f(a+1);t=f((a+1));
  printf(  “ %d,%d ” ,s,t);
}
```

程序运行后的输出结果是

- A) 10,64                      B) 10,10                      C) 64,10                      D) 64,64

( 36) 下面结构体的定义语句中，错误的是

- A) struct ord {int x;int y;int z;}; struct ord a;  
B) struct ord {int x;int y;int z;} struct ord a;  
C) struct ord {int x;int y;int z;} a;  
D) struct {int x;int y;int z;} a;

( 37) 设有定义：char \*c;，以下选项中能够使字符型指针 c 正确指向一个字符串的是

- A) char str[ ]= “ string ” ;c=str;                      B) scanf( “ %s” ,c);  
C) c=getchar();                      D) \*c= “ string ” ;

( 38) 有以下程序

```
#include
#include
struct A
{ int a; char b[10]; double c;};
struct A f(struct A t);
main()
{ struct A a={1001, “ ZhangDa” ,1098.0};
a=f(a);printf( “ %d,%s,%6.1f ” ,a.a,a.b,a.c);
}
struct A f(struct A t)
( t. a=1002;strcpy(t.b, “ ChangRong” );t.c=1202.0;return t; )
```

程序运行后的输出结果是

- A) 1001,ZhangDa,1098.0                      B) 1001,ZhangDa,1202.0  
C) 1001,ChangRong,1098.0                      D) 1001,ChangRong,1202.0

( 39) 若有以下程序段

```
int r=8;
printf( “ %d ” ,r>>1);
输出结果是
```

- A) 16      B) 8      C) 4      D) 2

( 40) 下列关于 C 语言文件的叙述中正确的是

- A) 文件由一系列数据依次排列组成，只能构成二进制文件  
B) 文件由结构序列组成，可以构成二进制文件或文本文件  
C) 文件由数据序列组成，可以构成二进制文件或文本文件  
D) 文件由字符序列组成，其类型只能是文本文件

## 二、填空题（每空2分，共30分）

(1) 某二叉树有 5 个度为 2 的结点以及 3 个度为 1 的结点，则该二叉树中共有 【1】 个结点。

- (2) 程序流程图中的菱形框表示的是 【2】 。
- (3) 软件开发过程主要分为需求分析、设计、编码与测试四个阶段，其中 【3】 阶段产生“软件需求规格说明书”。
- (4) 在数据库技术中，实体集之间的联系可以是一对一或一对多或多对多的，那么“学生”和“可选课程”的联系为 【4】 。
- (5) 人员基本信息一般包括：身份证号，姓名，性别，年龄等。其中可以作为主关键字的是 【5】 。
- (6) 若有定义语句：int a=5; ，则表达式：a++的值是 【6】 。
- (7) 若有语句 double x=17;int y; ，当执行 y=(int)(x/5)%2; 之后 y 的值为 【7】 。
- (8) 以下程序运行后的输出结果是 【8】 。

```
#include
main()
{ int x=20;
  printf(  “ %d” ,0<X<20);< p="" />
  printf(  “ %d ” ,0<X&&X<="" />
```

- (9) 以下程序运行后的输出结果是 【9】 。

```
#include
main ( )
{ int a=1,b=7;
  do {
    b=b/2;a+=b;
  } while (b>1);
  printf(  “ %d ” ,a);}
```

- (10) 有以下程序

```
#include
main()
{ int f,f1,f2,i;
  f1=0;f2=1;
  printf(  “ %d %d” ,f1,f2);
  for(i=3;i<=5;i++)
    { f=f1+f2; printf(      “ %d” ,f);
    f1=f2; f2=f;
  }
  printf(  “ n ” );
}
```

程序运行后的输出结果是 【10】 。

- (11) 有以下程序

```
#include
int a=5;
void fun(int b)
{ int a=10;
  a+=b;printf(  “ %d” ,a);
}
main()
{ int c=20;
  fun(c);a+= c;printf(  “ %d ” ,a);
}
```

程序运行后的输出结果是 【11】 。

- (12) 设有定义：



```
struct person
{ int ID;char name[12];}p;
```

请将 scanf( “ %d” , 【12】 ); 语句补充完整 , 使其能够为结构体变量 p 的成员 ID 正确读入数据。

( 13) 有以下程序

```
#include
main()
{ char a[20]= “ How are you? ” ,b[20];
  scanf( “ %s” ,b);printf( “ %s %s” ,a,b);
}
```

程序运行时从键盘输入 : How are you?<回车>

则输出结果为 【13】 。

( 14) 有以下程序

```
#include
typedef struct
{ int num;double s}REC;
void fun1( REC x ){x.num=23;x.s=88.5;}
main()
{ REC a={16,90.0 };
  fun1(a);
  printf( “ %d ” ,a.num);
}
```

程序运行后的输出结果是 【14】 。

( 15) 有以下程序

```
#include
fun(int x)
{ if(x/2>0) run(x/2);
  printf( “ %d ” ,x);
}
main()
{ fun(6);printf( “ n ” ); }
```

程序运行后的输出结果是 【15】 。

2009年 9月全国计算机等级考试二级 C语言笔试参考答案

#### 一、 选择题

CBDAB ACBCD DCABB DCAAB DCACD BACCB DAACA BADCC

#### 二、 填空题

( 10) 0 1 123 ( 9) 5 ( 8) 1 0 ( 7) 1 ( 6) 5 ( 5) 身份证号

( 4) 多对多 (3) 需求分析 ( 2) 逻辑条件 ( 1) 14

2009年3月二级 C语言笔试真题及答案

(一) 填空

- (1) 下列叙述中正确的是
- A ) 栈是“先进先出”的线性表
- B ) 队列是“先进先出”的线性表
- C ) 循环队列是非线性结构
- D ) 有序性表既可以采用顺序存储结构，也可以采用链式存储结构
- (2) 支持子程序调用的数据结构是
- A ) 栈    B ) 树    C ) 队列    D ) 二叉树
- (3) 某二叉树有5个度为2的结点，则该二叉树中的叶子结点数是
- A ) 10    B ) 8    C ) 6    D ) 4
- (4) 下列排序方法中，最坏情况下比较次数最少的是
- A ) 冒泡排序    B ) 简单选择排序    C ) 直接插入排序    D ) 堆排序
- (5) 软件按功能可以分为：应用软件、系统软件和支撑软件（或工具软件）。下面属于应用软件的是
- A ) 编译软件    B ) 操作系统    C ) 教务管理系统    D ) 汇编程序
- (6) 下面叙述中错误的是
- A ) 软件测试的目的是发现错误并改正错误
- B ) 对被调试的程序进行“错误定位”是程序调试的必要步骤
- C ) 程序调试通常也称为 Debug
- D ) 软件测试应严格执行测试计划，排除测试的随意性
- (7) 耦合性和内聚性是对模块独立性度量的两个标准。下列叙述中正确的是
- A) 提高耦合性降低内聚性有利于提高模块的独立性
- B) 降低耦合性提高内聚性有利于提高模块的独立性
- C) 耦合性是指一个模块内部各个元素间彼此结合的紧密程度
- D) 内聚性是指模块间互相连接的紧密程度
- (8) 数据库应用系统中的核心问题是
- A) 数据库设计
- B) 数据库系统设计
- C) 数据库维护
- D) 数据库管理员培训
- (9) 有两个关系 R, S如下：

R

A	B	C
a	3	2
b	0	1
c	2	1

S

A	B
a	3
b	0
c	2

由关系 R 通过运算得到关系 S，则所使用的运算为

A) 选择 B) 投影 C) 插入 D) 连接

(10) 将 E-R 图转换为关系模式时，实体和联系都可以表示为

A) 属性 B) 键 C) 关系 D) 域

(11) 以下选项中合法的标识符是

A) 1-1 B) 1—1 C) -11 D) 1--

(12) 若函数中有定义语句：int k；，则

A) 系统将自动给 k 赋初值0

B) 这时 k 中值无定义

C) 系统将自动给 k 赋初值-1

D) 这时 k 中无任何值

(13) 以下选项中，能用作数据常量的是

A) o115 B) 0118 C) 1.5e1.5 D) 115L

(14) 设有定义：int x=2；，以下表达式中，值不为6的是

A) x\*=x+1

B) x++,2\*x

C) x\*= ( 1+x)

D) 2\*x,x+=2

(15) 程序段：int x=12;

double y=3.141593; printf ( “ %d%8.6f” ,x,y ) ; 的输出结果是

A) 123.141593

B) 12

3.141593

C) 12 , 3.141593

D) 123.141593

(16) 若有定义语句：double x,y,\*px,\*py, 执行了 px=&x, py=&y; 之后，正确的输入语句是

A) scanf ( “ %f%f” ,x,y );

B) scanf ( “ %f%f” ,&x,&y );

C) scanf ( “ %lf%le ” ,px,py );

D) scanf ( “ %lf%lf ” ,x,y );

(17) 以下是 if 语句的基本形式：

if ( 表达式 )

语句

其中“表达式”

A) 必须是逻辑表达式

B) 必须是关系表达式

C) 必须是逻辑表达式或关系表达式

D) 可以是任意合法的表达式

(18) 有以下程序

```
#include <stdio.h>
```

```
main ( )
```

```
{int x;
```

```
scanf ( “ %d” ,&x );
```

```
if ( x<=3 ) ; else
```

```
if ( x!=10 )
```

```
printf ( “ %dn ” ,x );  
}
```

程序运行时，输入的值在哪个范围才会有输出结果

- A) 不等于10的整数
- B) 大于3且不等于10的整数
- C) 大于3或等于10的整数
- D) 小于3的整数

( 19) 有以下程序

```
#include<stdio.h>  
Main ( )  
{  
int a=1,b=2,c=3,d=0;  
if ( a= =1 &&b++= =2)  
if ( b!=2 || c--!=3 )  
printf ( “ %d,%d,%d ” ,a,b,c );  
else printf ( “ %d,%d,%d ” ,a,b,c );  
else printf ( “ %d,%d,%d ” ,a,b,c );  
}
```

程序运行后的输出结果是

- A) 1,2,3
- B) 1,3,2
- C) 1,3,3
- D) 3,2,1

( 20) 以下程序中的变量已正确定义

```
for ( i=0;i<4;i++,i++  
for ( k=1;k<3;k++ );printf ( “ * ” );
```

程序段的输出结果是

- A) \*\*\*\*\*
- B) \*\*\*\*
- C) \*\*
- D) \*

( 21) 有以下程序

```
#include<stdio.h>  
main ( )  
{char  
*s= ( “ ABC” );  
do  
{printf ( “ %d” ,*s%10 );s++;  
}while ( *s );  
}
```

注意，字母 A 的 ASCII 码值为65。程序运行后的输出结果是

- A) 5670
- B) 656667
- C) 567
- D) ABC

( 22) 设变量已正确定义，以下不能统计出一行中输入字符个数（不包含回车符）的程序段是

- A) n=0;while ( ( ch=getchar ( ) ) != ' n ' ) n++;  
B) n=0;while ( getchar ( ) != ' n ' ) n++;  
C) for ( n=0; getchar ( ) != ' n ' ;n++ );  
D) n=0;for ( ch=getchar ( ) ;ch!= ' n ' ;n++ );

( 23 ) 有以下程序

```
#include<stdio.h>
main ( )
{ int a1,a2;char c1,c2;
scanf ( " %d%c%d%c",&a1,&c1,&a2,&c2 );
printf ( " %d,%c,%d,%c",&a1,c1,a2,c2 );
}
```

若想通过键盘输入，使得 a1的值为12,a2的值为34，c1的值为字符 a,c2的值为字符 b, 程序输出结果是：12,a,34,b, 则正确的输入格式是 ( 以下

代表空格，<CR代表回车 )

- A) 12a34b<CR>  
B) 12a34b<CR>  
C) 12,a,34,b<CR>  
D) 12a34b<CR>

( 24 ) 有以下程序

```
#include<stdio.h>
int f ( int x,int y )
{return ( ) y-x ) *x );}
main ( )
{int a=3,b=4,c=5,d;
d=f ( f ( a,b ) ,f ( a,c ) );
printf ( " %dn",d );
}
```

程序运行后的输出结果是

- A) 10  
B) 9  
C) 8  
D) 7

( 25 ) 有以下程序

```
#include<stdio.h>
void fun ( char
*s )
{while ( *s )
{ if ( *s%2==0)
printf ( " %c",*s );
s++;
}
}
main ( )
{ char
a[]={ " good" };
}
```

```
fun ( a ) ;printf ( “ n ” );  
}
```

注意：字母 a 的 ASC 码值为97，程序运行后的输出结果是

- A) d
- B) go
- C) god
- D) good

( 26 ) 有以下程序

```
#include <stdio.h>  
void fun ( int *a,int *b )  
{int *c;  
c=a;a=b;b=c;  
}  
main ( )  
{int x=3,y=5,*p=&x,*q=&y;  
fun ( p,q );printf ( “ %d,%d,” ,*p,*q )  
fun ( &x,&y );printf ( “ %d,%d ” ,*p,*q );  
}
```

程序运行后的输出结果是

- A) 3,5,5,3
- B) 3,5,3,5
- C) 5,3,3,5
- D) 5,3,5,3

( 27 ) 有以下程序

```
#include <stdio.h>  
void f ( int *p,int *q ) ;  
main ( )  
{ int m=1,n=2,*r=&m;  
f ( r,&n );printf ( “ %d,%d’ ,m,n );  
}  
void f ( int *p,int *q )  
{p=p+1;*q=*q+1;}
```

程序运行后输出的结果是

- A) 1,3
- B) 2,3
- C) 1,4
- D) 1,2

( 28 ) 以下函数按每行8个输出数组中的数据

```
void fun ( int *w,int n )  
{ int i;  
for ( i=0;i<n;i++ )  
{  
printf ( “ %d” ,w );  
}  
printf ( “ n ” );  
}
```

下划线处应填入的语句是

- A) if ( i/8==0 ) print ( “ n ” ) ;
- B) if ( i/8==0 ) continue;
- C) if ( i%8==0) print ( “ n ” ) ;
- D) if ( i%8==0) continue;

( 29 ) 若有以下定义

```
int x[10],*pt=x;
```

则对 x 数组元素的正确应用是

- A) \*&x[10]
- B) \* ( x+3 )
- C) \* ( pt+10 )
- D) pt+3

( 30 ) 设有定义：char s[81];int i=10;; 以下不能将一行（不超过 80 个字符）带有空格的字符串真确读入的语句或语句组是

- A) gets ( s )
- B) while ( ( s[i++]=getchar ( ) ) != “ n ” ;s= “ 0 ” ;
- C) scanf ( “ %s” ,s );
- D) do{scanf ( “ %c” ,&s );}while ( s[i++]!= “ n ” ) ;s= “ 0 ” ;

( 31 ) 有以下程序

```
#include <stdio.h>
```

```
main ( )
```

```
{ char *a[ ]={ “ abcd ” , “ ef ” , “ gh ” , “ ijk ” };int l;
```

```
for ( i=0;i<4;i++ )
```

```
printf ( “ %c” ,*a );
```

```
}
```

程序运行后输出的结果是

- A) aegi
- B) dfhk
- C) abcd
- D) abcdefghijk

( 32 ) 以下选项中正确的语句组是

- A) char s[];s= “ BOOK!” ;
- B) char \*s;s={ “ BOOK!” };
- C) char s[10];s= “ BOOK!” ;
- D) char \*s;s= “ BOOK!” ;

( 33 ) 有以下程序

```
#include <stdio.h>
```

```
int fun{int x,int y}
```

```
{ if ( x==y) return ( x );
```

```
else returen ( ( x+y ) /2 )
```

```
}
```

```
main ( )
```

```
{ int a=4,b=5,c=6;
```

```
printf ( “ %dn ” ,fun ( 2*a,fun ( b,c ) ) )
```

```
}
```

程序运行后的输出结果是

- A) 3      B    ) 6      C    ) 8      D    ) 12

(34) 设函数中有整型变量 n , 为保证其在未赋值的情况下初值为 0 , 应选择的存储类别是

- A) auto
- B) register
- C) static
- D) auto 或 register

(35) 有以下程序

```
#include <stdio.h>
int b=2;
int fun ( int *k )
{ b=*k+b;return ( b ) ;}
main ( )
{ int a[10]={1,2,3,4,5,6,7,8},l;
for ( i=2;i<4;i++ ) {b=fun ( &a ) +b;printf ( “ %d” ,b );}
printf ( “ n” );
}
```

程序运行后输出的结果是

- A) 10 12
- B) 8 10
- C) 10 28
- D) 10 16

(36) 有以下程序

```
#include <stdio.h>
#define PT 3.5;
#define S ( x ) PT*x*x;
main ( )
{ int a=1, b=2; printf ( “ %4.1f n” ,S ( a+b ) );}
程序运行后输出的结果是
```

- A) 14.0
- B) 31.5
- C) 7.5
- D) 程序有错无输出结果

(37) 有以下程序

```
#include <stdio.h>
struct ord
{ int x,y; } dt[2]={1,2,3,4};
main ( )
{ struct ord *p=dt;
printf ( “ %d,” ,++p->x ); printf ( “ %d\n” ,++p->y );
}
```

程序的运行结果是

- A) 1,2
- B) 2,3
- C) 3,4
- D) 4,1

(38) 设有宏定义 : #include

IsDIV ( k,n ) ( ( k%n==1 ) ? 1:0 且变量 m 已正确定义并赋值 , 则宏调用 : IsDIV ( m,5 ) && IsDIV ( m,7 ) 为真时所表达的是



- A) 判断 m 是否能被5或者7整除
- B) 判断 m 是否能被5和7整除
- C) 判断 m被5或者7整除是否余1
- D) 判断 m被5和7整除是否余1

( 39) 有以下程序

```
#include <stdio.h>
main ( )
{ int a=5,b=1,t;
t= ( a<<2|b ); printf ( “ %dn ” ,t )
}
```

程序运行后的输出结果是

- A) 21
- B) 11
- C) 6
- D) 1

( 40) 有以下程序

```
#include <stdio.h>
main ( )
{ FILE *f;
f=fopen ( “ filea.txt ” , “ w ” );
fprintf ( f, “ abc ” );
fclose ( f );
}
```

若文本文件 filea.txt 中原有内容为：hello ，则运行以上程序后，文件 filea.txt 中的内容为

- A) helloabc
- B) abclo
- C) abc
- D) abchello

二、填空题（每空2分，共30分）

( 1) 假设一个长度为50的数组（数组元素的下标从 0到49）作为栈的存储空间，栈底指针 bottom 指向栈底元素，栈顶指针 top 指向栈顶元素，如果 bottom=49, top=30（数组下标），则栈中具有【1】个元素。

( 2) 软件测试可分为白盒测试和黑盒测试。基本路径测试属于 【2】测试。

( 3) 符合结构化原则的三种基本控制结构是：选择结构、循环结构和【3】。

( 4) 数据库系统的核心是 【4】

( 5) 在 E-R图中，图形包括矩形框、菱形框、椭圆框。其中表示实体联系的是【5】框。

( 6) 表达式 ( int ) ( ( double ) ( 5/2 ) +2.5 ) 的值是【6】

( 7) 若变量 x、y 已定义为 int 类型且 x 的值为99，y 的值为9，请将输出语句 printf ( 【7】 , x/y ); 补充完整，使其输出的计算结果形式为：x/y=11

(8) 有以下程序

```
#include <stdio.h>
main ( )
{
char c1,c2;
scanf( “ &c ” ,&c1);
while(c1<65||c1>90)
scanf( “ &c ” ,&c1);
```

```

c2=c1+32;
printf(    “ &c, &c n ” ,c1,c2);
}

```

程序运行输入65回车后，能否输出结果、结束运行（请回答能或不能）【8】。

（9）以下程序运行后的输出结果是【9】

```

#include <stdio.h>
main( )
{int k=1    , s=0 ;
do{
if{  ( ( k&2) !=0) continue ;
s+=k; k++;
}while ( k ) 10 ) ;
printf  ( “ s=%d/n ” ,s );
}

```

(10) 下列程序运行时，若输入 labced12df<回车>输出结果为【10】

```

#include <stdio.h>
main( )
{char
a =0,ch;
while((ch=getchar())!=    ‘n’ )
{if(a&2!=0&&(ch>    ‘a’ &&ch<=‘z’ )) ch=ch ‘a’ + ‘A’ ;
a++;putchar(ch);
}
printf(    “ n ” );
}

```

(11) 有以下程序，程序执行后，输出结果是【11】

```

#include <stdio.h>
void fun (int *a)
{a[0]=a[1];}
main()
{int a[10]={10,9,8,7,6,5,4,3,2,1};i;
for(i=2;i>=0;i--) fun{&a};
for(i=0;i<10;i++) printf(          “ &d” ,a);
printf(    “ n ” );
}

```

(12) 请将以下程序中的函数声明语句补充完整

```

#include <stdio.h>
int 【12】
main( )
{int x,y,(*p)();
p=max;
printf(    “ &dn ” ,&x,&y);
}
Int max(int a,int b)
{return (a>b/a:b);}

```

(13) 以下程序用来判断指定文件是否能正常打开，请填空

```
#include <stdio.h>
main( )
{FILE *fp;
if (((fp=fopen(    “ test.txt    ” , ” r ” ))== 1【3】 ))
printf ( “ 未能打开文件！ n ” ) ;
```

```
else
printf( “ 文件打开成功！ n ” );
```

(14) 下列程序的运行结果为【14】

```
#include <stdio.h>
#include <string.h>
struct A
{int a;char b[10];double
c;};
void
f (struct A *t);
main()
{struct A a=(1001,    ” ZhangDa” ,1098,0);
f(&a);printf(    “ &d,&s,&6,if n ” ,a.a,a.b,a.c);
}
void f(struct A *t)
{strcpy(t->b, ” ChangRong” );
}
```

(15) 以下程序把三个 NODETYPE 型的变量链接成一个简单的链表，并在 while 循环中输出链表结点数据域中的数据，请填空

```
#include <stdio.h>
struct node
{int data; struct node *next;};
typedef struct node NODETYPE;
main()
{NODETYPE a,b,c,*h,*p;
a.data=10;b.data=20;c.data=30;h=&a;
b.next=&b;b.next=&c;c.next=    ’ 0 ’ ;
p=h;
while(p){printf(    “ &d” p->data); 【15】 ; }
}
```

**答案：**

选择题：

1~10: DACDCABABC 11~20: BDDACDBCB 21~30: CDABABACBC 31~40: DBCCCB DAC

填空题：

- ( 1 ) 19 ( 2 ) 白盒 ( 3 ) 顺序结构 ( 4 ) 数据库管理系统 ( DBMS ) ( 5 ) 菱形  
( 6 ) 4 ( 7 ) "x/y=%d" ( 8 ) 能 ( 9 ) s=0 ( 10 ) 1AbCeDf2dF  
( 11 ) 7777654321 ( 12 ) max(int a,int b) ( 13 ) NULL  
( 14 ) 1001,ChangRong,1098.0 ( 15 ) p=p—>next

## 2008年9月二级C语言笔试真题及答案

### 一、选择题

- (1) 一个栈的初始状态为空。现将元素1、2、3、4、5、A、B、C、D、E依次入栈，然后再依次出栈，则元素出栈的顺序是( )。
- A) 12345ABCDE B) EDCBA54321 C) ABCDE12345 D) 54321EDCBA
- (2) 下列叙述中正确的是( )。
- A) 循环队列有队头和队尾两个指针，因此，循环队列是非线性结构  
B) 在循环队列中，只需要队头指针就能反映队列中元素的动态变化情况  
C) 在循环队列中，只需要队尾指针就能反映队列中元素的动态变化情况  
D) 循环队列中元素的个数是由队头指针和队尾指针共同决定
- (3) 在长度为n的有序线性表中进行二分查找，最坏情况下需要比较的次数是( )。
- A)  $O(n)$  B)  $O(n^2)$  C)  $O(\log_2 n)$  D)  $O(n \log_2 n)$
- (4) 下列叙述中正确的是( )。
- A) 顺序存储结构的存储一定是连续的，链式存储结构的存储空间不一定是连续的  
B) 顺序存储结构只针对线性结构，链式存储结构只针对非线性结构  
C) 顺序存储结构能存储有序表，链式存储结构不能存储有序表  
D) 链式存储结构比顺序存储结构节省存储空间
- (5) 数据流图中带有箭头的线段表示的是( )。
- A) 控制流 B) 事件驱动 C) 模块调用 D) 数据流
- (6) 在软件开发中，需求分析阶段可以使用的工具是( )。
- A) N-S图 B) DFD图 C) PAD图 D) 程序流程图
- (7) 在面向对象方法中，不属于“对象”基本特点的是( )。
- A) 一致性 B) 分类性 C) 多态性 D) 标识唯一性
- (8) 一间宿舍可住多个学生，则实体宿舍和学生之间的联系是( )。
- A) 一对一 B) 一对多 C) 多对一 D) 多对多
- (9) 在数据管理技术发展的三个阶段中，数据共享最好的是( )。
- A) 人工管理阶段 B) 文件系统阶段 C) 数据库系统阶段 D) 三个阶段相同
- (10) 有三个关系R、S和T如下：
- | R             | S | T |
|---------------|---|---|
| A B B C A B C |   |   |
| m 1 1 3 m 1 3 |   |   |
| n 2 3 5       |   |   |
- 由关系R和S通过运算得到关系T，则所使用的运算为( )。
- A) 笛卡尔积 B) 交 C) 并 D) 自然连接
- (11) 以下叙述中正确的是( )。
- A) C程序的基本组成单位是语句 B) C程序中的每一行只能写一条语句  
C) 简单C语句必须以分号结束 D) C语句必须在一行内写完
- (12) 计算机能直接执行的程序是( )。
- A) 源程序 B) 目标程序 C) 汇编程序 D) 可执行程序
- (13) 以下选项中不能作为C语言合法常量的是( )。
- A) 'cd' B) 0.1e+6 C) "\a" D) '\011'
- (14) 以下选项中正确的定义语句是( )。
- A) double a; b; B) double a=b=7; C) double a=7, b=7; D) double, a, b;
- (15) 以下不能正确表示代数式 $2abcd$ 的C语言表达式是( )。
- A)  $2*a*b/c/d$  B)  $a*b/c/d*2$  C)  $a/c/d*b*2$  D)  $2*a*b/c*d$

( 16) C 源程序中不能表示的数制是 ( )。

A) 二进制 B) 八进制 C) 十进制 D) 十六进制

( 17) 若有表达式 (w)?(--x):(++y) , 则其中与 w 等价的表达式是 ( )。

A) w==1 B) w==0 C) w!=1 D) w!=0

( 18) 执行以下程序段后 , w 的值为 ( )。

```
int w='A', x=14, y=15;
```

```
w=((x || y)&&(w<'a'));
```

A) -1 B) NULL C) 1 D) 0

( 19) 若变量已正确定义为 int 型 , 要通过语句 scanf("%d, %d, %d", &a, &b, &c) ; 给 a 赋值

1、给 b 赋值2、给 c 赋值3, 以下输入形式中错误的是 ( ) ( ) 代表一个空格符 \ )。

A) 0 0 0 1,2,3< 回车> 1 0 2 0 3< 回车>

C) 1, 0 0 0 2, 0 0 0>3D 回车>2,3< 回车>

( 20) 有以下程序段

```
int a, b, c;
```

```
a=10; b=50; c=30;
```

```
if (a>b) a=b, b=c; c=a;
```

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

程序的输出结果是 ( )。

A) a=10 b=50 c=10 B) a=10 b=50 c=30 C) a=10 b=30 c=10 D) a=50 b=30 c=50

( 21) 若有定义语句 : int m[]={5,4,3,2,1}, i=4; , 则下面对 m 数组元素的引用中错误的是 ( )。

A) m[--i] B) m[2\*2] C) m[m[0]] D) m[m[i]]

( 22) 下面的函数调用语句中 func 函数的实参个数是 ( )。

```
func (f2(v1, v2), (v3, v4, v5), (v6, max(v7, v8)));
```

A) 3 B) 4 C) 5 D) 8

( 23) 若有定义语句 : double x[5]={1.0,2.0,3.0,4.0,5.0}, \*p=x ; 则错误引用 x 数组元素的是 ( )。

A) \*p B) x[5] C) \*(p+1) D) \*x

( 24) 若有定义语句 : char s[10]="1234567\0\0"; , 则 strlen(s) 的值是 ( )。

A) 7 B) 8 C) 9 D) 10

( 25) 以下叙述中错误的是 ( )。

A) 用户定义的函数中可以没有 return 语句

B) 用户定义的函数中可以有多条 return 语句 , 以便可以调用一次返回多个函数值

C) 用户定义的函数中若没有 return 语句 , 则应当定义函数为 void 类型

D) 函数的 return 语句中可以没有表达式

( 26) 以下关于宏的叙述中正确的是 ( )。

A) 宏名必须用大写字母表示

B) 宏定义必须位于源程序中所有语句之前

C) 宏替换没有数据类型限制

D) 宏调用比函数调用耗费时间

( 27) 有以下程序

```
#include<stdio.h>
```

```
main()
```

```
{ int i, j;
```

```
for(i=3; i>=1; i--)
```

```
{ for(j=1; j<=2; j++) printf("%d", i+j);
```

```
printf("\n");
```

```
}
```

```
}
```

程序的运行结果是 ( )。

A) 2 3 4 B ) 4 3 2

3 4 5 5 4 3

C) 2 3 D ) 4 5

3 4 3 4

4 5 2 3

( 28 ) 有以下程序

```
#include <stdio.h>
```

```
main()
```

```
{ int x=1, y=2, z=3;
```

```
if(x>y)
```

```
if(y<z) printf("%d", ++z);
```

```
else printf("%d", ++y);
```

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

```
}
```

程序的运行结果是 ( )。

A) 331 B ) 41 C ) 2 D ) 1

( 29 ) 有以下程序

```
# include <stdio.h>
```

```
main()
```

```
{ int i=5;
```

```
do
```

```
{ if (i%3=1)
```

```
if (i%5==2)
```

```
{ printf("%d", i); break;}
```

```
i++;
```

```
} while(i!=0);
```

```
printf("\n");
```

```
}
```

程序的运行结果是 ( )。

A) \*7 B ) \*3\*5 C ) \*5 D ) \*2\*6

( 30 ) 有以下程序

```
#include <stdio.h>
```

```
int fun(int a,int b)
```

```
{ if(b==0) return a;
```

```
else return(fun(--a,--b));
```

```
}
```

```
main()
```

```
{ printf("%d\n", fun(4,2));}
```

程序的运行结果是 ( )。

A) 1 B ) 2 C ) 3 D ) 4

( 31 ) 有以下程序

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int fun(int n)
```

```

{ int *p;
p=(int*)malloc(sizeof(int));
*p=n; return *p;
}
main()
{ int a;
a = fun(10); printf("%d\n", a+fun(10));
}

```

程序的运行结果是 ( )。

A) 0 B) 10 C) 20 D) 出错

( 32) 有以下程序

```

#include <stdio.h>
void fun(int a, int b)
{ int t;
t=a; a=b; b=t;
}
main()
{ int c[10]={1,2,3,4,5,6,7,8,9,0}, i;
for (i=0; i<10; i+=2) fun(c[i], c[i+1]);
for (i=0; i<10; i++) printf("%d,", c[i]);
printf("\n");
}

```

程序的运行结果是 ( )。

A) 1,2,3,4,5,6,7,8,9,0, B ) 2,1,4,3,6,5,8,7,0,9, C ) 0,9,8,7,6,5,4,3,2,1, D )  
0,1,2,3,4,5,6,7,8,9,

( 33) 有以下程序

```

#include <stdio.h>
struct st
{ int x, y;} data[2]={1,10,2,20};
main()
{ struct st *p=data;
printf("%d,", p->y); printf("%d\n", (++p)->x);
}

```

程序的运行结果是 ( )。

A) 10,1 B) 20,1 C) 10,2 D) 20,2

( 34) 有以下程序

```

#include <stdio.h>
void fun(int a[], int n)
{ int i, t;
for(i=0; i<n/2; i++) {t=a[i]; a[i]=a[n-1-i]; a[n-1-i]=t;}
}
main()
{ int k[10]={1,2,3,4,5,6,7,8,9,10}, i;
fun(k,5);
for(i=2; i<8; i++) printf("%d", k[i]);
printf("\n");
}

```

```
}
```

程序的运行结果是 ( )。

A) 345678 B) 876543 C) 1098765 D) 321678

( 35 ) 有以下程序

```
#include <stdio.h>
#define N 4
void fun(int a[][N], int b[])
{ int i;
  for(i=0; i<N; i++) b[i]=a[i][i];
}
main()
{ int x[][N]={1,2,3},{4},{5,6,7,8},{9,10}},y[N], i;
  fun(x,y);
  for (i=0; i<N; i++) printf("%d,", y[i]);
  printf("\n");
}
```

程序的运行结果是 ( )。

A) 1,2,3,4, B ) 1,0,7,0, C ) 1,4,5,9, D ) 3,4,8,10,

( 36 ) 有以下程序

```
#include <stdio.h>
int fun(int (*s)[4],int n, int k)
{ int m, i;
  m=s[0][k];
  for(i=1; i<n; i++) if(s[i][k]>m) m=s[i][k];
  return m;
}
main()
{ int a[4][4]={1,2,3,4},{11,12,13,14},{21,22,23,24},{31,32,33,34}};
  printf("%d\n", fun(a,4,0));
}
```

程序的运行结果是 ( )。

A) 4 B ) 34 C ) 31 D) 32

( 37 ) 有以下程序

```
#include <stdio.h>
main()
{ struct STU { char name[9]; char sex; double score[2]; };
  struct STU a={"Zhao",'m',85.0,90.0), b={"Qian",'f',95.0,92.0);
  b=a;
  printf("%s,%c,%2.0f,%2.0f\n",b.name,b.sex,b.score[0],b.score[1]);
}
```

程序的运行结果是 ( )。

A) Qian,f,95,92 B ) Qian,m,85,90 C ) Zhao,f,95,92 D ) Zhao,m,85,90

( 38 ) 假定已建立以下链表结构 , 且指针 p 和 q 已指向如图所示的结点 :

```
head a b c
data next
p    q
```



则以下选项中可将 q 所指结点从链表中删除并释放该结点的语句组是 ( )。

A) (\*p).next=(\*q).next; free(p); B ) p=q->next; free(q);

C) p=q; free(q); D ) p->next=q->next; free(q);

( 39) 有以下程序

```
#include <stdio.h>
main()
{ char a=4;
printf("%d\n", a=a<<1);
}
```

程序的运行结果是 ( )。

A) 40 B) 16 C) 8 D) 4

( 40) 有以下程序

```
#include <stdio.h>
main()
{ FILE *pf;
char *s1="China",*s2="Beijing";
pf=fopen("abc.dat","wb+");
fwrite(s2,7,1,pf);
rewind(pf);
fwrite(s1,5,1,pf);
fclose(pf);
}
```

以上程序执行后 abc.dat 文件的内容是 ( )。

B) China B ) Chinang C) ChinaBeijing D ) BeijingChina

## 二、填空题 ( 每空 2 分 , 共 30 分 )

( 1) 对下列二叉树进行中序遍历的结果 【1】 。

A

B C

D E F

X Y Z

( 2) 按照软件测试的一般步骤 , 集成测试应在 【2】 测试之后进行。

( 3) 软件工程三要素包括方法、工具和过程 , 其中 , 【3】 支持软件开发的各个环节的控制和管理。

( 4) 数据库设计包括概念设计、 【4】 和物理设计。

( 5) 在二维表中 , 元组的 【5】 不能再分成更小的数据项。

( 6) 设变量 a 和 b 已正确定义并赋初值。请写出与 a-=a+b 等价的赋值表达式 【6】 。

( 7) 若整型变量 a 和 b 中的值分别为 7 和 9 , 要求按以下格式输出 a 和 b 的值 :

a=7

b=9

请完成输出语句 : printf ( " 【7】 ",a,b); 。

( 8) 以下程序的输出结果是 【8】 。

```
#include <stdio.h>
main()
{ int i,j,sum;
for(i=3;i>=1;i--)
{ sum=0;
for(j=1;j<=i;j++) sum+=i*j;
```

```

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

```

( 9 ) 以下程序的输出结果是 【9】 。

```

#include <stdio.h>
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: printf("%d",*p++); break;
case 3: printf("%d",*(--p));
}
}
}

```

( 10 ) 以下程序的输出结果是 【10】 。

```

#include <stdio.h>
#define N 5
int fun(int *s, int a, int n)
{ int j;
*s=a; j=n;
while(a!=s[j])j--;
return j;
}
main()
{ int s[N+1]; int k;
for(k=1; k<=N; k++) s[k]=k+1;
printf("%d\n",fun(s,4,N));
}

```

( 11 ) 以下程序的输出结果是 【11】 。

```

#include <stdio.h>
int fun(int x)
{ static int t=0;
return(t +=x);
}
main()
{ int s,i;
for(i=1;i<=5;i++) s=fun(i);
printf("%d\n",s);
}

```

( 12 ) 以下程序按下面指定的数据给 x 数组的下三角置数，并按如下形式输出，请填空。

```

4
3 7
2 6 9
1 5 8 10
#include <stdio.h>

```

```

main()
{ int x[4][4],n=0,i,j;
for(j=0;j<4;j++)
for(i=3;i>=j;    【12】 ) {n++;x[i][j]=    【13】 ;}
for(i=0;i<4;i++)
{ for(j=0;j<=i;j++) printf("%3 d",x[i][j]);
printf("\n");
}
}

```

( 13) 以下程序的功能是：通过函数 func 输入字符并统计输入字符的个数。输入时用字符 @ 作为输入结束标志。请填空。

```

#include <stdio.h>
long 【14】 ;
main()
{ long n;
n=func(); printf("n=%ld\n",n);
}
long func()
{ long m;
for( m=0; getchar()!='@';    【15】 );
return m;
}

```

## 2008 年9 月笔试卷参考答案

一、选择题 ( 1 ) B) ( 2 ) D) ( 3 ) C) ( 4 ) A) ( 5 ) D) ( 6 ) B) ( 7 ) A) ( 8 ) B)  
( 9 ) C) ( 10 ) D) ( 11 ) C) ( 12 ) D) ( 13 ) A) ( 14 ) C) ( 15 ) D) ( 16 ) A)  
( 17 ) D) ( 18 ) C) ( 19 ) B) ( 20 ) A) ( 21 ) C) ( 22 ) A) ( 23 ) B) ( 24 ) A)  
( 25 ) B) ( 26 ) C) ( 27 ) D) ( 28 ) D) ( 29 ) A) ( 30 ) B) ( 31 ) C) ( 32 ) A)  
( 33 ) C) ( 34 ) D) ( 35 ) B) ( 36 ) C) ( 37 ) D) ( 38 ) D) ( 39 ) C) ( 40 ) B)

二、填空题 ( 1 ) 【1】 DBXEAYFZC ( 2 ) 【2】 单元 ( 3 ) 【3】 过程 ( 4 ) 【4】 逻辑设计  
( 5 ) 【5】 分量 ( 6 ) 【6】  $a=a-(a+b)$  或  $(a=-b)$  ( 7 ) 【7】  $a=\%d\backslash nb=\%d$  ( 8 ) 【8】 1  
( 9 ) 【9】 9911 ( 10 ) 【10】 3 ( 11 ) 【11】 15 ( 12 ) 【12】 i-- ( 13 ) 【13】 n  
( 14 ) 【14】 func() ( 15 ) 【15】 m++

## 2008年4月二级 C 语言笔试真题及答案

### 一、 选择题（共 70 分）

(1) 程序流程图中指有箭头的线段表示的是

A) 图元关系 B) 数据流 C) 控制流 D) 调用关系

(2) 结构化程序设计的基本原则不包括

A) 多态性 B) 自顶向下 C) 模块化 D) 逐步求精

(3) 软件设计中模块划分应遵循的准则是

A) 低内聚低耦合 B) 高内聚低耦合 C) 低内聚高耦合 D) 高内聚高耦合

(4) 在软件开发中，需求分析阶段产生的主要文档是

A) 可行性分析报告 B) 软件需求规格说明书 C) 概要设计说明书 D) 集成测试计划

(5) 算法的有穷性是指

A) 算法程序的运行时间是有限的 B) 算法程序所处理的数据量是有限的

C) 算法程序的长度是有限的 D) 算法只能被有限的用户使用

(6) 对长度为 n 的线性表排序，在最坏情况下，比较次数不是  $n(n-1)/2$  的排序方法是

A) 快速排序 B) 冒泡排序 C) 直接插入排序 D) 堆排序

(7) 下列关于栈的叙述正确的是

A) 栈按“先进先出”组织数据 B) 栈按“先进后出”组织数据

C) 只能在栈底插入数据 D) 不能删除数据

(8) 在数据库设计中，将 E-R图转换成关系数据模型的过程属于 [C]

A) 需求分析阶段 B) 概念设计阶段 C) 逻辑设计阶段 D) 物理设计阶段

(9) 有三个关系 R、S 和 T 如下：

T

B C D

a 0 k1

R S

B C D

a 0 k1

b 1 n1

B C D

f 3 h2

a 0 k1

n 2 x1

有关系 R 和 S 通过运算得到关系 T，则所使用的运算为

A) 并 B) 自然连接 C) 笛卡尔积 D) 交

(10) 设有表示学生选课的一张表，学生 S(学号，姓名，性别，年龄，身份证号)，课程 C(课号，课名)，选课 SC(学号，课号，成绩)，则表 SC 的关键字(键或码)为

A) 课号，成绩 B) 学号，成绩 C) 学号，课号 D) 学号，姓名，成绩

(11) 以下叙述中正确的是

A) C 程序中的注释只能出现在程序的开始位置和语句的后面 B) C 程序书写格式严格，要求一行内只能写一个语句

C) C 程序书写格式自由，一个语句可以写在多行上 D) 用 C 语言编写的程序只能放在一个程序文件中

(12) 以下选项中不合法的标识符是

A) print B) FOR C) &a D) \_00

(13) 以下选项中不属于字符常量的是

A) 'C' B) "C" C) '\xCC0' D) '\072'

(14) 设变量已正确定义并赋值，以下正确的表达式是

A)x=y\*5=x+z B)int(15.8%5) C)x=y+z+5,++y D)x=25%5.0

(15) 以下定义语句中正确的是

A)int a=b=0 ; B)char A=65+1,b='b';

C)float a=1,\*b=&a,\*c=&b ; D)double a=0.0,b=1.1;

(16) 有以下程序段

```
char ch; int k;
```

```
ch='a'; k=12;
```

```
printf("%c,%d",ch,ch,k); printf("k=%d\n",k);
```

已知字符 a 的 ASCII 十进制代码为 97，则执行上述程序段后输出结果是

A) 因变量类型与格式描述符的类型不匹配输出无定值

B) 输出项与格式描述符个数不符，输出为零值或不定值

C)a,97,12k=12 D)a,97,k=12

(17) 已知字母 A 的 ASC 代码值为 65，若变量 kk 为 char 型，以下不能正确判断出 kk 中的值为大写字母的表达式是

A)kk>='A'&&kk<='Z' B)!(kk>='A' kk<='Z')

C)(kk+32)>='a'&&(kk+32)<='z' D)isalpha(kk)&&(kk<91)

(18) 当变量 c 的值不为 2、4、6 时，值也为“真”的表达式是

A)(c==2) (c==4) (c==6) B)(c>=2&& c<=6) (c!=3) (c!=5)

C)(c>=2&& c<=6)&&!(c%2) D)(c>=2&& c<=6)&&(c%2!=1)

(19) 若变量已正确定义，有以下程序段

```
int a=3,b=5,c=7;
```

```
if(a>b) a=b; c=a;
```

```
if(c!=a) c=b;
```

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

其输出结果是

A) 程序段有语法错 B)3 , 5 , 3 C)3 , 5 , 5 D)3 , 5 , 7

(20) 有以下程序

```
#include <stdio.h>
```

```
main()
```

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

```
switch(x)
```

```
{ case 1:
```

```
switch(y)
```

```
{ case 0: a++; break;
```

```
case 1: b++; break;
```

```
}
```

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

```
case 3: a++; b++;
```

```
}
```

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

```
}
```

程序的运行结果是

A ) a=1 , b=0      B      ) a=2,b=2      C      ) a=1,b=1      D      ) a=2,b=1

(21) 有以下程序

```
#include <stdio.h>
```

```
main()
```

```
{ int x=8;
```

```
for( ; x>0; x--)  
{ if(x%3) {printf(      “ %d,” ,x); continue;}  
printf(      “ %d,” ,--x);  
}
```

程序的运行结果是

A ) 7 , 4 , 2    B    ) 8 , 7 , 5 , 2    C    ) 9 , 7 , 6 , 4    D           ) 8 , 5 , 4 , 2

( 22 ) 以下不构成无限循环的语句或者语句组是

A ) n=0; B ) n=0;

do{++n;}while(n<=0); while(1){n++;}

C)n=10; D)for(n=0,i=1; ;i++) n+=i;

while(n);{n--;}

( 23 ) 有以下程序

```
#include <stdio.h>
```

```
main()
```

```
{ int a[ ]={1,2,3,4},y,*p=&a[3];
```

```
-- p; y=*p; printf(      “ y=%d ” ,y);
```

```
}
```

程序的运行结果是

A    ) y=0 B ) y=1 C ) y=2 D ) y=3

( 24 ) 以下错误的定义语句是

A    ) int x[][3]={0},{1},{1,2,3};                    B                    ) int x[4][3]={1,2,3},{1,2,3},{1,2,3},{1,2,3};

C    ) int x[4][]={{1,2,3},{1,2,3},{1,2,3},{1,2,3}};    D                    ) int x[][3]={1,2,3,4};

( 25 ) 设有如下程序段

```
char s[20]=      “ Beijing ” ,*p;
```

```
p=s;
```

则执行 p=s; 语句后 , 以下叙述正确的是

A        ) 可以用\*p 表示 s[0] B    ) s 数组中元素的个数和 p 所指字符串长度相等

C    ) s 和 p 都是指针变量 D) 数组 s 中的内容和指针变量 p 中的内容相等

( 26 ) 若有定义 : int a[2][3]; , 以下选项中对 a 数组元素正确引用的是

A    ) a[2][!1] B    ) a[2][3] C    ) a[0][3] D    ) a[1>2][!1]

( 27 ) 有定义语句 : char s[10]; , 若要从终端给 s 输入 5 个字符 , 错误的输入语句是

A ) gets(&s[0]); B    ) scanf( “ %s” ,s+1); C    ) gets(s); D    ) scanf( “ %s” ,s[1]);

( 28 ) 以下叙述中错误的是

A ) 在程序中凡是以 “ # ” 开始的语句行都是预处理命令行

B ) 预处理命令行的最后不能以分号表示结束

C ) #define MAX是合法的宏定义命令行

D ) C程序对预处理命令行的处理是在程序执行的过程中进行的

( 29 ) 以下结构体类型说明和变量定义中正确的是

A ) typedef struct B    ) struct REC;

```
{int n; char c;}REC; {int n; char c};
```

```
REC t1,t2; REC t1,t2;
```

C ) typedef struct REC ; D    ) struct

```
{int n=0; char c=      ‘ A ’ ;}t1,t2; {int n;char c;}REC t1,t2;
```

( 30 ) 以下叙述中错误的是

A ) gets 函数用于从终端读入字符串

B ) getchar 函数用于从磁盘文件读入字符

C) fputs 函数用于把字符串输出到文件

D) fwrite 函数用于以二进制形式输出数据到文件

(31) 有以下程序

```
#include <stdio.h>

main()
{ int s[12]={1,2,3,4,4,3,2,1,1,1,2,3},c[5]={0},i;
for(i=0;i<12;i++) c[s[i]]++;
for(i=1;i<5;i++) printf(        “ %d” ,c[i]);
printf(        “ n ” );
}
```

程序的运行结果是

A)1 2 3 4 B)2 3 4 4 C)4 3 3 2 D )1 1 2 3

(32) 有以下程序

```
#include <stdio.h>

void fun(int *s,int nl,int n2)
{ int i,j,t;
i=nl; j=n2;
while(i<j) {t=s[i];s[i]=s[j];s[j]=t;i++;j--;}
}

main()
{ int a[10]={1,2,3,4,5,6,7,8,9,0},k;
fun(a,0,3); fun(a,4,9); fun(a,0,9);
for(k=0;k<10;k++)printf(        “ %d” ,a[k]); printf(        “ \n ” );
}
```

程序运行的结果是

A)0987654321 B)4321098765 C)5678901234 D)0987651234

(33) 有以下程序

```
#include <stdio.h>
#include <string.h>

void fun(char *s[ ],int n)
{ char *t; int i,j;
for(i=0;i<n-1;i++)
for(j=i+1;j<n;j++)
if(strlen(s[i])>strlen(s[j])) {t=s[i];s[i]=s[j];s[j]=t;}
}

main()
{char *ss[]={        “ bcc ” , “ bbcc ” , “ xy ” , “ aaaacc ” , “ aabcc ” };
fun(ss,5); printf(        “ %s,    n ” ,ss[0],ss[4]);
}
```

程序的运行结果是

A)xy,aaaacc B)aaaacc,xy C)bcc,aabcc D)aabcc,bcc

(34) 有以下程序

```
#include <stdio.h>

int f(int x)
{int y;
if(x==0||x==1) return (3);
```

```

y=x*x-f(x-2);
return y;
}
main()
{int z;
  z=f(3); printf(      “ %d ” ,z);
}

```

程序的运行结果是

A)0 B)9 C)6 D)8

(35) 有以下程序

```

#include <stdio.h>
void fun(char *a,char *b)
{while(*a==  ' * ' ) a++;
while(*b=*a) {b++;a++;}
}
main()
{char *s=  " *****a*b*****  ",t[80];
fun(s,t); puts(t);
}

```

程序的运行结果是

A)\*\*\*\*\*a\*b B) a\*b C) a\*b\*\*\*\*\* D) ab

(36) 有以程序

```

#include <stdio.h>
#include <string.h>
typedef struct { char name[9]; char sex; float score[2]; } STU;
void f( STU a)
{ STU b={  “ Zhao ” ,  ' m' ,85.0,90.0} ; int i;
strcpy(a.name,b.name);
a.sex=b.sex;
for(i=0;i<2;i++) a.score[i]=b.score[i];
}
main()
{ STU c={  “ Qian ” ,  ' p ' ,95.0,92.0};
f(c); printf(      “ %s,%c,%2.0f,%2.0fn ” ,c.name,c.sex,c.score[0],c.score[1]);
}

```

程序的运行结果是

A)Qian,f,95,92 B) Qian,m,85,90 C) Zhao,f,95,92 D) Zhao,m,85,90

(37) 有以下程序

```

#include <stdio.h>
main()
{FILE *fp; int a[10]={1,2,3},i,n;
fp=fopen(  “ dl.dat  ” ,  “ w ” );
for(i=0;i<3;i++) fprintf(fp,      “ %d” ,a[i]);
fprintf(fp,      “ n ” );
fclose(fp);
fp=fopen(  “ dl.dat  ” ,  “ r ” );

```



```
fscanf(fp, " %d", &n);
fclose(fp);
printf( " %d", n);
}
```

程序的运行结果是

A)12300 B) 123 C) 1 D) 321

(38) 变量 a 中的数据用二进制表示的形式是 01011101, 变量 b 中的数据用二进制表示的形式是 11110000 若要求将 a 的高 4 位取反, 低 4 位不变, 所要执行的运算是

A)a^b B)a|b C)a&b D)a<<4

(39) 在 C 语言中, 只有在使用时才占用内存单元的变量, 其存储类型是

A)auto 和 register B)extern 和 register C) auto 和 static D) static 和 register

(40) 设有定义语句 int(\*f)(int);, 则以下叙述正确的是

A)f 是基类型为 int 的指针变量 B) f 是指向函数的指针变量, 该函数具有一个 int 类型的形参  
C) f 是指向 int 类型一维数组的指针变量 D) f 是函数名, 该函数的返回值是基类型为 int 类型的地址

## 二、填空题 ( 每空 2 分 , 共 30 分 )

( 1 ) 测试用例包括输入值集和【 1 】值集。

( 2 ) 深度为 5 的满二叉树有【 2 】个叶子结点。

( 3 ) 设某循环队列的容量为 50, 头指针 front=5 ( 指向队头元素的前一位置 ), 尾指针 rear=29 ( 指向队尾元素 ), 则该循环队列中共有【 3 】个元素。

( 4 ) 在关系数据库中, 用来表示实体之间联系的是【 4 】。

( 5 ) 在数据库管理系统提供的数据库定义语言、数据操纵语言和数据控制语言中, 【 5 】负责数据的模式定义与数据的物理存取构建。

( 6 ) 已有定义 : char c= ' ' ;int a=1,b;( 此处的初值为空格字符 ), 执行 b=!c&&a;后 b 的值为【 6 】。

( 7 ) 设变量已正确定义为整型, 则表达式 n=i=2,++i,i++ 的值为【 7 】。

( 8 ) 若有定义 : int k; , 以下程序段的输出结果是【 8 】。

```
for(k=2;k<6;k++,k++) printf( " ##%d", k);
```

(9) 以下程序段的定义语句中, x[1] 的初值是【 9】, 程序运行后输出的内容是【 10】。

```
#include<stdio.h>
main()
{ int x[]={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16},*p[4],i;
for(i=0;i<4;i++)
{ p[i]=&x[2*i+1];
printf( " %d", p[i][0]);
}
printf( " \n");
}
```

(10) 以下程序的输出结果是【 11】。

```
#include<stdio.h>
void swap(int *a,int *b)
{ int *t;
t=a; a=b; b=t;
}
main()
{ int i=3,j=5,*p=&i,*q=&j;
swap(p,q); printf( " %d %d", *p,*q);
```

}(11) 以下程序的输出结果是【 12】。

```
#include<stdio.h>

main()
{ int a[5]={2,4,6,8,10}, *p;
p=a; p++;
printf(    “ %d” ,*p);
}
```

(12) 以下程序的输出结果是【 13】。

```
#include<stdio.h>

void fun(int x)
{ if(x/2>0) fun(x/2);
printf(    “ %d ” ,x);
}

main()
{fun(3); printf(    “ n ” );}
```

( 13 ) 以下程序中函数 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(    【14】 == ' M' ) n++;
return n;
}

main()
{SS W[N]={ {1 , “ AA” , ' F ' }, {2, “ BB” , ' M' }, {3, “ CC” , ' M' } }; int n;
n=fun(W); printf(    “ n=%d” ,n);
}
```

(14) 以下程序从名为 filea.dat 的文本文件中逐个读入字符并显示在屏幕上。请填空：

```
#include<stdio.h>

main()
{FILE *fp; char ch;
fp=fopen(    【15】 );
ch=fgetc(fp);
while(!feof(fp)) { putchar(ch); ch=fgetc(fp);}
putchar(    ‘ n ’ ); fclose(fp); }
```

## 参考答案：

### 一、选择题

1-10: CABBA DBCDC 11-20: CCBCB DBBBD 21-30: DADCA DDDAB 31-40: CCACC ABAAB

二、填空题：1、输出 2 、 16 3 、 24 4 、 关系 5 、 数据定义语言 6 、 0 7 、 3 8 、 ##2# #4 9 、 2 10 、 2 4 6 8 11 、 3 5 12 、 4 13 、 1 3

2007年9月

一、选择题

(1) 软件是指

A) 程序 B) 程序和文档 C) 算法加数据结构 D) 程序、数据和相关文档的集合

(2) 软件调试的目的是

A) 发现错误 B) 改正错误 C) 改善软件的性能 D) 验证软件的正确性

(3) 在面向对象方法中，实现信息隐蔽是依靠

A) 对象的继承 B) 对象的多态 C) 对象的封装 D) 对象的分类

(4) 下列叙述中，不符合良好程序设计风格的是

A) 程序的效率第一，清晰第二 B) 程序的可读性好

C) 程序中有必要的注释 D) 输入数据前要有提示信息

(5) 下列叙述中正确的是

A) 程序执行的效率与数据的存储结构密切相关 B) 程序执行的效率只取决于程序的控制结构

C) 程序执行的效率只取决于所处理的数据量 D) 以上三种说法都不对

(6) 下列叙述中正确的是

A) 数据的逻辑结构与存储结构必定是一一对应的

B) 由于计算机存储空间是向量式的存储结构，因此，数据的存储结构一定是线性结构

C) 程序设计语言中的数组一般是顺序存储结构，因此，利用数组只能处理线性结构

D) 以上三种说法都不对

(7) 冒泡排序在最坏情况下的比较次数是

A)  $n(n+1)/2$  B)  $n\log_2 n$  C)  $n(n-1)/2$  D)  $n/2$

(8) 一棵二叉树中共有70个叶子结点与80个度为1的结点，则该二叉树中的总结点数为

A) 219 B) 221 C) 229 D) 231

(9) 下列叙述中正确的是

A) 数据库系统是一个独立的系统，不需要操作系统的支持

B) 数据库技术的根本目标是要解决数据的共享问题

C) 数据库管理系统就是数据库系统

D) 以上三种说法都不对

(10) 下列叙述中正确的是

A) 为了建立一个关系，首先要构造数据的逻辑关系

B) 表示关系的二维表中各元组的每一个分量还可以分成若干数据项

C) 一个关系的属性名表称为关系模式 D) 一个关系可以包括多个二维表

(11) C语言源程序名的后缀是

A) .exe B) .C C) .obj D) .cp

(12) 可在C程序中用做用户标识符的一组标识符是

A) and B) Date C) Hi D) case

\_2007 y-m-d Dr.Tom Bigl

(13) 以下选项中，合法的一组C语言数值常量是

A) 028 B) 12. C) .177 D) 0x8A

. 5e-3 OXa23 4c1.5 10,000

-0xf 4.5e0 Oabc 3.e5

(14) 以下叙述中正确的是

A) C语言程序将从源程序中第一个函数开始执行

B) 可以在程序中由用户指定任意一个函数作为主函数，程序将从此开始执行

C) C语言规定必须用main作为主函数名，程序将从此开始执行，在此结束

D) main可作为用户标识符，用以命名任意一个函数作为主函数

(15) 若在定义语句：int a,b,c,\*p=&c; 之后，接着执行以下选项中的语句，则能正确执行的语句是

A) scanf("%d",a,b,c); B)scanf("%d%d%d",a,b,c); C)scanf("%d",p); D)scanf("%d",&p);

(16) 以下关于 long、int 和 short 类型数据占用内存大小的叙述中正确的是

A)均占4个字节 B) 根据数据的大小来决定所占内存的字节数

C) 由用户自己定义 D) 由 C语言编译系统决定

(17) 若变量均已正确定义并赋值，以下合法的 C语言赋值语句是

A) x=y==5; B)x=n%2.5; C)x+n=l; D)x=5=4+1;

(18) 有以下程序段

```
int j; float y; char name[50];
```

```
scanf("%2d%f%s",&j,&y,name);
```

当执行上述程序段，从键盘上输入 55566 7777abc后，y 的值为

A) 55566.0 B) 566.0 C) 7777.0 D) 566777.0

(19) 若变量已正确定义，有以下程序段

```
i=0;
```

```
do printf("%d",i);while(i++);
```

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

其输出结果是

A) 0 , 0 B) 0 , 1 C) 1 , 1 D) 程序进入无限循环

(20) 有以下计算公式

若程序前面已在命令中包含 math.h 文件，不能够正确计算上述公式的程序段是

A) if(x>=0) y=sqrt(x); B)y=sqrt(x)

else y=sqrt(-x); if(x<0) y=sqrt(-x);

C)if(x>=0)y=sqrt(x); D)y=sqrt(x>=0?x:-x);

lf(x<0)y=sqrt(-x);

(21) 设有条件表达式：(EXP)?i++;j--，则以下表达式中(EXP)完全等价的是

A)( EXP==0) B )( EXP!=0) C )( EXP==1) D )( EXP!=1)

(22) 有以下程序

```
#include
```

```
main()
```

```
{int y=9;
```

```
for( y>0;y--)
```

```
if(y%3==0) printf("%d",--y);
```

```
}
```

程序的运行结果是

A) 741 B) 963 C) 852 D) 875421

(23) 已有定义：char c;，程序前面已在命令行中包含 ctype.h 文件，不能用于判断 c 中的字符是否为大写字母的表达式是

A)isupper(c) B) ' A ' <=c<=' Z '

C) ' A ' <=c&& c<=' Z ' D)c<=(' 2'-32)&&(' a '-32)<=c

(24) 有以下程序

```
#include
```

```
main()
```

```
{int i,j,m=55;
```

```
for(i=1;i<=3;i++)
```

```
for(j=3;j<=i;j++) m=m%j;
```

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

```
}
```

程序的运行结果是

A) 0 B) 1 C) 2 D) 3

(25) 若函数调用时的实参为变量时，以下关于函数形参和实参的叙述中正确的是

A) 函数的实参和其对应的形参共占同一存储单元

B) 形参只是形式上的存在，不占用具体存储单元

C) 同名的实参和形参占同一存储单元

D) 函数的形参和实参分别占用不同的存储单元

(26) 已知字符 'A' 的ASC 代码值是65，字符变量 c1 的值是 'A' ,c2 的值是 'D' 。执行语句 `printf("%d,%d",c1,c2-2);` 后，输出结果是

A) A, B B) A, 68 C) 65, 66 D) 65, 68

(27) 以下叙述中错误的是

A) 改变函数形参的值，不会改变对应实参的值 B) 函数可以返回地址值

C) 可以给指针变量赋一个整数作为地址值 D) 当在程序的开头包含文件 `stdio.h` 时，可以给指针变量赋 `NULL`

(28) 以下正确的字符串常量是

A) " B) 'abc' C) OlympicGames D) ""

(29) 设有定义：`char p[]={ '1', '2', '3' },*q=p;`，以下不能计算出一个 `char` 型数据所占字节数的表达式是

A) `sizeof(p)` B) `sizeof(char)` C) `sizeof(*q)` D) `sizeof(p[0])`

(30) 有以下函数

```
int aaa(char *s)
```

```
{char *t=s;
```

```
while(*t++);
```

```
t--;
```

```
return(t-s);
```

```
}
```

以下关于 `aaa` 函数的功能叙述正确的是

A) 求字符串 `s` 的长度 B) 比较两个串的大小 C) 将串 `s` 复制到串 `t` D) 求字符串 `s` 所占字节数

(31) 若有定义语句：`int a[3][6];`，按在内存中的存放顺序，`a` 数组的第 10 个元素是

A) `a[0][4]` B) `a[1][3]` C) `a[0][3]` D) `a[1][4]`

(32) 有以下程序

```
#include
```

```
void fun(char **p)
```

```
{++p; printf("%s\n",*p);}
```

```
main()
```

```
{char *a[]={"Morning","Afternoon","Evening","Night"};
```

```
fun(a);
```

```
}
```

程序的运行结果是

A) Afternoon B) fternoon C) Morning D) orning

(33) 若有定义语句：`int a[2][3],*p[3];`，则以下语句中正确的是

A) `p=a;` B) `p[0]=a;` C) `p[0]=&a[1][2];` D) `p[1]=&a;`

(34) 有以下程序

```
#include
```

```
void fun(int *a,int n)/*fun 函数的功能是将 a 所指数组元素从大到小排序*/
```

```
{int t,i,j;
```

```
for(i=0;i<N-1;i++)
```

```

for(j=i+1;j<N;J++)
if (a)
main()
{int c[10]={1,2,3,4,5,6,7,8,9,0},i;
fun(c+4,6);
for (i=0;i<10;i++) printf("%d,",c);
printf("\n");
}

```

程序运行的结果是

A) 1,2,3,4,5,6,7,8,9,0, B)0,9,8,7,6,5,1,2,3,4, C) 0,9,8,7,6,5,4,3,2,1, D)1,2,3,4,9,8,7,6,5,0,

(35) 有以下程序

```

#include
int fun(char s[])
{int n=0;
while(*s<= ' 9 ' &&*s>=' 0 ' ) {n=10*n+*s - ' 0 ' ;s++;}
return(n);
}
main()
{char s[10]={ ' 6 ' , ' 1 ' , ' * ' , ' 4 ' , ' * ' , ' 9 ' , ' * ' , ' 0 ' , ' * ' };
printf("%d\n",fun(s));
}

```

程序运行的结果是

A) 9 B ) 61490 C ) 61 D ) 5

( 36) 当用户要求输入的字符串中含有空格时，应使用的输入函数是

A)scanf() B)getchar() C)gets() D)getc()

(37) 以下关于字符串的叙述中正确的是

A) C语言中有字符串类型的常量和变量 B ) 两个字符串中的字符个数相同时才能进行字符串大小的比较  
C) 可以用关系运算符对字符串的大小进行比较 D ) 空串一定比空格打头的字符串小

(38) 有以下程序：

```

#include
void fun(char *t,char *s)
{
while(*t!=0)t++;
while((*t++=*s++)!=0);
}
main()
{
char ss[1 0]= " acc " ,aa[10]= " bbxyy " ;
fun(ss,aa);
printf( " %s,%s " ,ss,aa);
}

```

程序运行结果是

A) accxyy , bbxyy B) acc, bbxyyC) accxyy,bbxyy D) accbbxyy,bbxyy

(39) 有以下程序

```

#include
#include
void fun(char s[][10],int n)
{
char t;int i,j;

```

```

for(i=0;i<N-1;i++)
for(j=i+1,j<N;j++)
/* 比较字符串的首字符大小，并交换字符串的首字符 */
if(s[i]>s[j]){t=s[i];s[i]=s[j];s[j]=t;}
}
main()
{
char ss[5][10]={    “ bcc ” , “ bbcc ” , “ xy ” , “ aaaacc ” “ aabcc ” }
fun(ss,5); printf(    “ %s,%s ” ,ss[0],ss[4]);
} 程序运行结果是

```

A) xy,aaaacc B) aaaacc,xy C) xcc,aabcc D) acc,xabcc

(40) 在一个 C 语言源程序文件中所定义的全局变量，其作用域为：

- A) 所在文件的全部范围 B) 所在程序的全部范围  
C) 所在函数的全部范围 D) 由具体定义位置和 extern 说明来决定范围

(41) 有以下程序

```

#include
int a=1;
int f(int c)
{static int a=2;
c=c+1;
return (a++)+c;}
main()
{ int i,k=0;
for(i=0;i<2;i++){int a=3;k+=f(a);}
k+=a;
printf(    “ %d ” ,k);
}

```

程序运行结果是

A) 14 B) 15 C) 16 D) 17

(42) 有以下程序

```

#include
void fun(int n,int *p)
{ int f1,f2;
if(n==1||n==2) *p=1;
else
{ fun(n-1,&f1); fun(n-2,&f2);
*p=f1+f2;
}
}
main()
{ int s;
fun(3,&s); printf("%d\n",s);
}

```

程序的运行结果是

A)2 B)3 C)4 D)5

(43) 若程序中有宏定义行:#define N 100 则以下叙述中正确的是

- A)宏定义行中定义了标识符 N的值为整数100      B)      在编译程序对 C源程序进行预处理时用 100替换标识符 N  
C) 对 C源程序进行编译时用100替换标识符 N      D)      在运行时用100替换标识符 N

(44) 以下关于 typedef 的叙述错误的是

- A)用 typedef 可以增加新类型      B)typedef      只是将已存在的类型用一个新的名字来代表  
C)用 typedef 可以为各种类型说明一个新名,但不能用来为变量说明一个新名  
D)用 typedef 为类型说明一个新名,通常可以增加程序的可读性

(45) 有以下程序

```
#include
struct tt
{int x;struct tt *y;} *p;
struct tt a[4]={20,a+1,15,a+2,30,a+3,17,a};
main()
{ int i;
p=a;
for(i=1;i<=2;i++) {printf("%d,",p->x); p=p->y;}
}
```

程序的运行结果是

- A)20,30, B)30,17 C)15,30, D)20,15,

(46) 有以下程序

```
#include
#include
typedef struct{ char name[9];char sex; float score[2]; } STU;
STU f(STU a)
{ STU b={"Zhao",'m',85.0,90.0}; int i;
strcpy(a.name,b.name);
a. sex=b.sex;
for(i=0;i<2;i++) a.score=b.score;
return a;
}
main()
{STU c={"Qian",'f',95.0,92.0},d;
d=f(c); printf("%s,%c,%2.0f,%2.0f\n",d.name,d.sex,d.score[0],d.score[1]);
} 程序的运行结果是
```

- A)Qian,f,95,92 B)Qian,m,85,90 C)Zhao,m,85,90 D)Zhao,f,95,92

(47) 设有以下定义

```
union data
{ int d1; float d2; }demo;
```

则下面叙述中错误的是

- A)变量 demo与成员 d2所占的内存字节数相同      B)      变量 demo中各成员的地址相同  
C)变量 demo和各成员的地址相同      D)      若给 demo.d1赋99后, demo.d2 中的值是99.0

(48) 有以下程序

```
#include
main()
{ int a=1,b=2,c=3,x;
x=(a^b)&c; printf("%d\n",x);
}
```



程序的运行结果是

A)0 B)1 C)2 D)3

(49) 读取二进制文件的函数调用形式为 :fread(buffer,size,count,fp); , 其中 buffer 代表的是

- A)一个文件指针, 指向待读取的文件                      B)                      一个整型变量, 代表待读取的数据的字节数  
C)一个内存块的首地址, 代表读入数据存放的地址      D)                      一个内存块的字节数

(50) 有以下程序

```
#include
main()
{FILE *fp; int a[10]={1,2,3,0,0},i;
fp=fopen("d2.dat","wb");
fwrite(a,sizeof(int),5,fp);
fwrite(a,sizeof(int),5,fp);
fclose(fp);
fp=fopen("d2.dat","rb");
fread(a,sizeof(int),10,fp);
fclose(fp);
for(i=0;i<10;i++) printf("%d",a);
} 程序的运行结果是
```

- A)1,2,3,0,0,0,0,0,0,0,                      B)1,2,3,1,2,3,0,0,0,0,  
C)123,0,0,0,0,123,0,0,0,0,                      D)1,2,3,0,0,1,2,3,0,0,

二、填空题( 每空2分, 共40分)

请将每一个空的正确答案写在答题卡 [1] 至[20] 序号的横线上, 答在试卷上不得分.

- (1) 软件需求规格说明书应具有完整性、无歧义性、正确性、可验证性、可修改性等特性, 其中最重要的 \_[1]\_\_\_\_\_.  
(2) 在两种基本测试方法中, \_[2]\_\_\_\_\_ 测试的原则之一是保证所测模块中每一个独立路径至少要执行一次 .  
(3) 线性表的存储结构主要分为顺序存储结构和链式存储结构 . 队列是一种特殊的线性表, 循环队列是队列的 \_[3]\_\_\_\_\_ 存储结构.  
(4) 对下列二叉树进行中序遍历的结果为 \_[4]\_\_\_\_\_

```
F
/\
C E
/\ \
A D G
//\
B H P
```

- (5) 在 E-R 图中矩形表示 \_[5]\_\_\_\_\_  
(6) 执行以下程序时输入 1234567, 则输出结果是 \_[6]\_\_\_\_\_.

```
#include
main()
{ int a=1,b;
scanf("%2d%2d",&a&b);printf("%d %d\n",a,b);
}
```

(7) 以下程序的功能是: 输出 a、b、c 三个变量中的最小值. 请填空。

```
#include
main()
{ int a,b,c,t1,t2;
scanf("%d%d%d",&a,&b,&c);
```

```
t1=a
t2=c
printf("%d\n",t2);
}(8) 以下程序的输出结果是_[9]_____.
```

```
#include
main()
{ int n=12345,d;
while(n!=0){ d=n%10; printf("%d",d); n/=10;}
}(9) 有以下程序段,且变量已正确定义和赋值
for(s=1.0,k=1;k<=n;k++) s=s+1.0/(k*(k+1));
printf("s=%f\n\n",s);
```

请填空,使下面程序段的功能为完全相同

```
s=1.0;k=1;
while(_[10]_____){ s=s+1.0/(k*(k+1)); _[11]_____;}
printf("s=%f\n\n",s);
```

(10) 以下程序的输出结果是\_[12]\_\_\_\_\_.

```
#include
main()
{ int i;
for(i='a';i<'f';i++,i++) printf("%c",i-'a'+'A');
printf("\n");
}(11) 以下程序的输出结果是_[13]_____.
```

```
#include
#include
char *fun(char *t)
{ char *p=t;
return(p+strlen(t)/2);
}
main()
{ char *str="abcdefgh";
str=fun(str);
puts(str);
}
```

(12) 以下程序中函数 f 的功能是在数组 x 的 n 个数(假定 n 个数互不相同)中找出最大最小数,将其中最小的数与第一个数对换,把最大的数与最后一个数对换. 请填空.

```
#include
void f(int x[],int n)
{ int p0,p1,i,j,t,m;
i=j=x[0]; p0=p1=0;
for(m=0;m<N;M++)
{ if(x[m]>i) {i=x[m]; p0=m;}
else if(x[m]}
t=x[p0]; x[p0]=x[n-1]; x[n-1]=t;
t=x[p1];x[p1]=_[14]_____; _[15]_____=t;
}
main()
{ int a[10],u;
```

```
for(u=0;u<10;u++) scanf("%d",&a);
f(a,10);
for(u=0;u<10;u++) printf("%d",a);
printf("\n");
```

}(13) 以下程序统计从终端输入的字符中大写字母的个数,num[0]中统计字母 A的个数,num[1]中统计字母 B的个数,其它依次类推.用#号结束输入,请填空.

```
#include
#include
main()
{ int num[26]={0},i; char c;
while((_[16]_____)!='#')
if(isupper(c)) num[c- ' A ' ]+= _[17]_____;
for(i=0;i<26;i++)
Printf("%c:%d\n",i+'A',num);
```

}(14) 执行以下程序的输出结果是 \_[18]\_\_\_\_\_.

```
#include
main()
{ int i,n[4]={1};
for(i=1;i<=3;i++)
{ n=n[i-1]*2+1; printf("%d",n); }
```

}(15) 以下程序的输出结果是 \_[19]\_\_\_\_\_.

```
#include
#define M 5
#define N M+M
main()
{ int k;
k=N*N*5; printf("%d\n",k);
```

}(16) 函数 main()的功能是:在带头结点的单链表中查找数据域中值最小的结点.请填空

```
#include
struct node
{ int data;
struct node *next;
};
int min(struct node *first)/* 指针 first 为链表头指针*/
{ struct node *p; int m;
p=first->next; m=p->data;p=p->next;
for(;p!=NULL;p=_[20]_____)
if(p->data<m)
return m;
```

2007年9月参考答案一、选择题

1-5 DBCAA 6-10 CCABA 11-15 BABCC 16-20 DABBB 21-25 BCBBD 26-30 CCDAA  
31-35 BACDC 36-37 CDDBC 42-45 AABAD 46-50 CDDCD、填空题

1、无歧义性 2、白盒测试 3、顺序 4、ACBDFEHGP、5实体集 6、12 34

7、a:b 8、c:t1 9、54321 10、k<=n 11、k++

2007 年 4 月

## 一、选择题

(1)下列叙述中正确的是

- A)算法的效率只与问题的规模有关，而与数据的存储结构无关
- B)算法的时间复杂度是指执行算法所需要的计算工作量
- C)数据的逻辑结构与存储结构是一一对应的
- D)算法的时间复杂度与空间复杂度一定相关

(2)在结构化程序设计中，模块划分的原则是

- A)各模块应包括尽量多的功能 B)各模块的规模应尽量大
- C)各模块之间的联系应尽量紧密 D)模块内具有高内聚度、模块间具有低耦合度

(3)下列叙述中正确的是

- A)软件测试的主要目的是发现程序中的错误
- B)软件测试的主要目的是确定程序中错误的位置
- C)为了提高软件测试的效率，最好由程序编制者自己来完成软件测试的工作
- D)软件测试是证明软件没有错误

(4)下面选项中不属于面向对象程序设计特征的是

- A)继承性 B)多态性 C)类比性 D)封闭性

(5)下列对列的叙述正确的是

- A)队列属于非线性表 B)队列按“先进后出”原则组织数据
- C)队列在队尾删除数据 D)队列按“先进先出”原则组织数据

(6)对下列二叉树

进行前序遍历的结果为

- A) DYBEAFCZX B) YDEBFZXCAC) ABDYECFXZ D) ABCDEFXYZ

(7)某二叉树中有  $n$  个度为 2 的结点，则该二叉树中的叶子结点为

- A)  $n+1$  B)  $n-1$  C)  $2n$  D)  $n/2$

(8)在下列关系运算中，不改变关系表中的属性个数但能减少元组个数的是

- A) 并 B) 交 C) 投影 D) 笛卡儿乘积

(9)在 E-R 图中，用来表示实体之间联系的图形是

- A) 矩形 B) 椭圆形 C) 菱形 D) 平行四边形

(10)下列叙述中错误的是

- A) 在数据库系统中，数据的物理结构必须与逻辑结构一致
- B)数据库技术的根本目标是要解决数据的共享问题
- C)数据库设计是指在已有数据库管理系统的基础上建立数据库
- D)数据库系统需要操作系统的支持

(11)算法中，对需要执行的每一步操作，必须给出清楚、严格的规定，这属于算法的

- A) 正当性 B) 可行性 C) 确定性 D) 有穷性

(12)下列叙述中错误的是

- A) 计算机不能直接执行用 C 语言编写的源程序
- B) C 程序经 C 编译后，生成后缀为 .obj 的文件是一个二进制文件
- C) 后缀为 .obj 的文件，经连接程序生成后缀为 .exe 的文件是一个二进制文件
- D) 后缀为 .obj 和 .exe 的二进制文件都可以直接运行

(13)按照 C 语言规定的用户标识符命名规则，不能出现在标识符中的是

- A) 大写字母 B) 连接符 C) 数字字符 D) 下划线

(14)以下叙述中错误的是

- A) C语言是一种结构化程序设计语言
- B) 结构化程序有顺序、分支、循环三种基本结构组成
- C) 使用三种基本结构构成的程序只能解决简单问题
- D) 结构化程序设计提倡模块化的设计方法

(15) 对于一个正常运行的 C 程序，以下叙述中正确的是

- A) 程序的执行总是从 main 函数开始，在 main 函数结束
- B) 程序的执行总是从程序的第一个函数开始，在 main 函数结束
- C) 程序的执行总是从 main 函数开始，在程序的最后一个函数中结束
- D) 程序的执行总是从程序的第一个函数开始，在程序的最后一个函数中结束

(16) 设变量均已正确定义，若要通过 scanf(“ %d%c%d%c” ,&a1,&c1,&a2,&c2);语句为变量 a1 和 a2 赋数值 10 和 20，为变量 c1 和 c2 赋字符 X 和 Y。以下所示的输入形式正确的是（注： 代表空格字符）

- A) 10 X 20 Y 回车
- B) 10 X20 Y 回车
- C) 10 X 回车
- D) 10X 回车

(17) 若有代数式（其中 e 仅代表自然对数的底数，不是变量），则以下能够正确表示该代数式的 C 语言表达式是

- A) sqrt(abs(n^x+e^x))
- B) sqrt(fabs(pow(n,x)+pow(x,e)))
- C) sqrt(fabs(pow(n,x)+exp(x,e)))
- D) sqrt(fabs(pow(x,n)+exp(x)))

(18) 设有定义：int k=0;，以下选项的四个表达式中与其他三个表达式的值不相同的是

- A) k++
- B) k+=1
- C) ++k
- DD) k+1

(19) 有以下程序，其中 %u 表示按无呼号整数输出

```

Main()
{unsigned int x=0xFFFF; /* x的初值为十六进制数 */
  Printf(“ %u\n” ,x);
}

```

程序运行后的输出结果是

- A) -1
- B) 65535
- C) 32767
- D) 0xFFFF

(20) 设变量 x 和 y 均已正确定义并赋值，以下 if 语句中，在编译时将产生错误信息的是

- A) if(x++)
- B) if(x>y&y!=0);
- C) if(x>y)x- -
- D) if(y<0) {;}

(21) 以下选项中，当 x 为大于 1 的奇数时，值为 0 的表达式

- A) x%2==1
- B) x/2
- C) x%2!=0
- D) x%2==0

(22) 以下叙述中正确的是

- A) break 语句只能用于 switch 语句题中
- B) continue 语句的作用是：使程序的执行流程跳出包含它的所有循环
- C) break 语句只能用在循环体内和 switch 语句体内
- D) 在循环体内使用 break 语句和 continue 语句的作用相同

(23) 有以下程序

```

Main()
{int k=5,n=0;
  do
  {switch(k)
   {case1: case3:n+=1; break;
    Default;n=0;k--;
    Case2: case4:n+=2;k--;break;
   }
  }
}

```

```
Printf( “ %d” ,n);
}while(k>0&& n<5);
```

}程序运行后的输出结果是

A ) 235    B) 0235    C) 02356    D) 2356

( 24 ) 有以下程序

```
mian()
{int i,j;
  for(i=1;i<4;i++)
{for(j=i;j<4;j++) printf( “ %d*%d=%d “ ,i,j,i*j);
Printf( “ \n ” );
}
```

}程序运行后的输出结果是

```
*p=&n;
```

```
Printf( “ Input n:” ); scanf(“ %d ” ,&p); printf( “ output n:” ); printf( “ %d\n” ,p);
```

```
}
```

该程序试图通过指针 p 为变量 n 读入数据并输出，但程序有多处错误，以下语句正确的是

A ) int n,\*p=NULL;    B)\*p=&n;    C)scanf(“ %d ” ,&p)    D)printf( “ %d\n” ,p);

(31)以下程序中函数 f 的功能是：当 flag 为 4 时，进行有小到大排序；当 flag 为 0 时，进行由大到小排序。

```
void f(int b[],int n,int flag)
{int i,j,t;
for(i=0;i<n-1;i++)
  for (j=i+1;j<n;j++)
    if(flag?b[ i ]>b[j]:b[ i ]<b[j]) {t=b[ i ];b[ i ]=b[j];b[j]=t; }
}
```

```
main()
```

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

```
  f(&a[2],5,0); f(a,5,1);
```

```
  for(i=0;i<10;i++) printf( “ %d,” a[ i ]);
```

}程序运行后的输出结果是

A ) 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 ,    B ) 3 , 4 , 5 , 6 , 7 , 2 , 1 , 8 , 9 , 10 ,

C ) 5 , 4 , 3 , 2 , 1 , 6 , 7 , 8 , 9 , 10 ,    D ) 10 , 9 , 8 , 7 , 6 , 5 , 4 , 3 , 2 , 1 ,

( 32 ) 有以下程序

```
void f(int b[])
{int l;
  for(i=2;i<6;i++) b[ i ]*=2;
}
```

```
main()
```

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

```
f
```

```
for(i=0;i<10;i++) printf( “ %d,” ,a[ i ]);
```

}程序运行后的输出结果是

A ) 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 10 ,    B ) 1 , 2 , 6 , 8 , 10 , 12 , 7 , 8 , 9 , 10

C ) 1 , 2 , 3 , 4 , 10 , 12 , 14 , 16 , 9 , 10 ,    D ) 1 , 2 , 6 , 8 , 10 , 12 , 14 , 16 , 9 , 10 ,

( 33 ) 有以下程序

```
typedef struct{int b,p;}A;
```

```
void f(A c) /*注意：c 是结构变量名 */
```

```

    {int j;
c.b+=1; c.p+=2;
}
main(){int i;
A a={1,2};
f
printf( “ %d,%d\n” ,a.b,a.p);
}程序运行后的输出结果是

```

A) 2 , 3   B) 2 , 4   C) 1 , 4   D) 1 , 2  
( 34) 有以下程序

```

main()
{int a[4][4]={1,4,3,2},{8,6,5,7},{3,7,2,5},{4,8,6,1}},i,j,k,t;
for(i=0;i<4;i++)
    for(j=0;j<3;j++)
        for(k=j+1;k<4;k++)
            if(a[j][ i]>a[k][ i]){t=a[j][ i];a[j][ i ]=a[k][ i ];a[k][ i ]=t;}/*    按列排序*/
for(i=0;i<4;i++)printf( “ %d, ” ,a[ i ][j]);
}程序运行后的输出结果是

```

A)1,6,5,7,   B)8,7,3,1,   C)4,7,5,2,   D)1,6,2,1,  
(35) 有以下程序

```

main()
{int a[4][4]={1,4,3,2},{8,6,5,7},{3,7,2,5},{4,8,6,1}},i,k,t;
for(i=0;i<3;i++)
    for(k=i+1;k<4;k++) if(a[ i ][ i ]<a[k][k]){t=a[ i ][ i ];a[ i ][ i ]=a[k][k];a[k][k]=t;}
for(i=0;i<4;i++)printf( “ %d, ” ,a[0][ i ]);
}程序运行后的输出结果是

```

A)6,2,1,1,   B)6,4,3,2,   C)1,1,2,6,   D)2,3,4,6,

(36) 有以下程序

```

void f(int *q)
{int i=0;
for( i<5;i++)(*q)++;
}
main()
{int a[5]={1,2,3,4,5},i;
f
for(i=0;i<5;i++)printf( “ %d, ” ,a[ i ]);
}程序运行后的输出结果是

```

A)2,2,3,4,5,   B)6,2,3,4,5,   C)2,3,4,5,6,

(37) 有以下程序

```

#include <string.h>
main()
{char p[20]={ ‘ a ’ , ‘ b ’ , ‘ c ’ , ‘ d ’ },q[]= “ abc” , r[]= “ abcde” ;
Strcpy(p+strlen(q),r); strcat(p,q);
Printf( “ %d%d\n” ,sizeof(p),strlen(p));
}程序运行后的输出结果是

```

A)20 9   B)9 9   C)20 11   D)11 11

(38) 有以下程序

```
#include <string.h>

main()
{char p[20]={ ' a ', ' b ', ' c ', ' d ' },q[]=" abc", r[]=" abcde"
strcat(p,r); Strcpy(p+strlen(q),q);
Printf( " %d\n",sizeof(p));
}程序运行后的输出结果是
```

A)9 B)6 C)11 D)7

(39) 有以下程序

```
#include <string.h>

main()
{ char p[20]={ ' a ', ' b ', ' c ', ' d ' }, q[]=" abc", r[]=" abcde" ;
Strcat(p,r); strcpy(p+strlen(q),q);
Printf( " %d\n",strlen(p));
}程序运行后的输出结果是
```

A) 9 B) 6 C) 11 D) 7

```
else {int a=7; t +=a++; }
return t+a++;
}
```

```
main ( )
{int s=a, i=0 ;
for ( ; i < 2 ; i++ s+=f(i);
printf ( " %d\n",s);
}程序运行后的输出结果是
```

A)24 B)28 C)32 D)36

(43) 有一个名为 init.txt 的文件，内容如下：

```
#define HDY(A,B) A/B
# define PRINT(Y) Printf(" y=%d\n.",Y)
```

有以下程序

```
#include " init.txt "

main()
{int a=1,b=2,c=3,d=4,k;
K=HDY ( a+c, b+d );
PRINT ( K );
}
```

下面针对该程序的叙述正确的是

A ) 编译有错 B) 运行出错  
C ) 运行结果为 y=0 D)运行结果为 y=6

(44) 有以下程序

```
Main()
{char ch[]=" uvwxyz ",*pc ;
Pc=ch; printf(" %c\n",*(pc+5));
}
```

程序运行后的输出结果是

A)z B)0 C)元素 ch[5]地址 D)字符 y 的地址

(45) 有以下程序

```
struct S {int n; int a[20];};
```



```
void f(struct S *P)
{int i,j,t;
for(i=0;i<p->n-1;i++)
for(j=j+1;j<p->n-1;j++)
程序运行后的输出结果是
```

A ) 3    B) 4    C) 5    D) 6

( 49) 有以下程序

```
#include <stdio.h>
Main()
{FILE *fp; int l,a[6]={1,2,3,4,5,6};
fp=fopen(“ d2.dat” , ” w ” );
fprintf(fp, ” %d%d\n” ,a[0],a[1],a[2]); fprintf(fp, “ %d%d\n” ,a[3],a[4],a[5]);
fclose(fp);
fp=fopen(“ d2.dat” , ” r ” );
fscanf(fp,” “ %d%d\n” ,&k,&n); printf( “ %d%d\n” ,k,n);
fclose(fp);
}程序运行后的输出结果是
```

A)1 2    B)1 4    C)123 4    D) 123 456

(50)有以下程序

```
#include <stdio.h>
main ()
{FILE *fp; int l,a[6]={1,2,3,4,5,6k};
fp=fopen(“ d3.dat” , ” w+b ” );
fwrite(a,size(int),6,fp);
fseek(fp,sizeof(int)*3,SEEK SET);/*该语句使读文件的位置指针从文件头向后移动 3个 int 型数据*/
fread(a,sizeof(int),3,fp); fclose(fp);
for(i=0;i<6;i++) printf( “ %d, ” ,a[ i]);
}程序运行后的输出结果是
```

A)4,5,6,4,5,6,    B)1,2,3,4,5,6,    C)4,5,6,1,2,3,    D)6,5,4,3,2,1,

二 . 填空题 ( 每空 2 分 , 共 40 分 )

- ( 1 ) 在深度为 7 的满二叉树中 , 度为 2 的结点个数为\_\_\_\_\_。
- ( 2 ) 软件测试分为白箱 ( 盒 ) 测试和黑箱 ( 盒 ) 测试 , 等价类划分法属于 \_\_\_\_\_测试。
- ( 3 ) 在数据库系统中 , 实现各种数据管理功能的核心软件称为 数据库管理系统\_\_\_\_\_。
- ( 4 ) 软件生命周期可分为多个阶段 , 一般分为定义阶段、开发阶段和维护阶段。编码和测试属于 \_\_\_\_\_阶段。
- ( 5 ) 在结构化分析使用的数据流图 ( DFD ) 中 , 利用 \_\_\_\_\_对其中的图形元素进行确切解释。
- ( 6 ) 执行以下程序后的输出结果是 \_\_\_\_\_ 。

```
main()
{int a=10;
a=(3*5,a+4); printf(“ a=%d\n” ,a);
}
```

(7) 当执行以下程序时 , 输入 1234567890回车> , 则其中 while 循环体将执行 \_\_\_\_\_次。

```
# include <stdio.h>
main()
{char ch;
While((ch=getchar())!=' 0 ' ) printf( “ # ” );
} (8) 以下程序的运行结果是 _____。
```

```
int k=0;
void fun(int m)
{ m+=k; k+=m; printf( " m=%d\n k=%d " ,m,k++);}
main()
{ int i=4;
fun(i++); printf( " i=%d k=%d\n" ,i,k);
} (9) 以下程序的运行结果是_____。
```

```
main()
{int a=2,b=7,c=5;
Switch(a>0)
{case 1:switch(b<0)
{case 1:switch( @ " ); break;
Case 2: printf( ! " ); break;
}
Case 0: switch(c==5)
{ case 0: printf( " * " ); break;
Case 1: printf( " # " ); break;
Case 2: printf( " $ " ); break;
}
default : printf( " & " );
}
Printf( " \n " );
}
```

(10) 以下程序的输出结果是 \_\_\_\_\_

```
# include <string.h>
main()
{ printf( " %d\n" ,strlen( " " ));
}
```

(11) 已定义 char ch= " \$ " ;int i=1,j;执行 j=!ch&&i++ 以后 , i 的值为 \_\_\_\_\_

( 12 ) 以下程序的输出结果是 \_\_\_\_\_

```
# include <string.h>
main()
{ char a[]={ ' \1 ' , ' \2 ' , ' \3 ' , ' \4 ' , ' \0 ' };
Printf( " %d %d\n" ,sizeof ,srlen );
}
```

(13) 设有定义语句：int a[][3]={0},{1},{2};, 则数组元素 a[1][2]的值为 \_\_\_\_\_

( 14 ) 以下程序的功能是：求出数组 x 中各相邻两个元素的和，依次存放到 a数组中，然后输出，请填空。

```
Main()
{int x[10],a[9],l;
For (i=0;i<10;i++)
Scanf( " %d" ,&x[ i]);
For( _____ i<10;i++)
A[i-1]=x[ i]+ _____
For(i=0;i<9;i++)
Printf( " %d" ,a[ i]);
Printf( " \n " );
}
```

(15) 以下程序的功能是：利用指针指向三个整型变量，并通过指针运算找出三个数中的最大值，输出到屏幕上，请填空：

```
Main()
{int x,y,z,max,*px,*py,*pz,*pmax;
```

```
scanf(“ %d%d%d” ,&x,&y,&z);
Px=&x;
Py=&y;
Pz=&z;
Pmax=&max;
```

```
_____  
if(*pmax<*py)*pmax=*py;  
if(*pmax<*pz)*pmax=*pz;  
printf( “ max=%d\n” ,max);  
(16) 以下程序的输出结果是 _____
```

```
int fun(int*x,int n)  
{if(n==0)  
return x[0];  
else return x[0]+fun(x+1,n-1);  
}
```

```
main()  
{int a[]={1,2,3,4,5,6,7};  
printf( “ %d\n” ,fun(a,3));  
(17) 以下程序的输出结果是 _____
```

```
#include<stdlib.h>  
main()  
{char *s1,*s2,m;  
s1=s2=(char*)malloc(sizeof(char));  
*s1=15;  
*s2=20;  
m=*s1+*s2;  
printf( “ %d\n” ,m);  
(18) 设有说明
```

```
struct DATE{int year;int month; int day;};
```

请写出一条定义语句，该语句定义 d 为上述结构体变量，并同时为其成员 year、month、day 依次赋初值 2006、10、1： \_\_\_\_\_;

(19) 设有定义：FILE\*fw;，请将以下打开文件的语句补充完整，以便可以向文本文件 readme.txt 的最后续写内容。

```
fw=fopen(“ readme.txt” , “ _____” )
```

2007年4月二级C语言笔试答案选择题

1-5 : BDACD      6-10: CABCA      11-15: CDBCA      16-20: DCABC      21-25: DCABA      26-30: ADCDA

31-35: BBDDDB      36-40: DCACA      41-45: DADAA      46-50: ACBDA

填空题 1、 63    2、 黑盒    3、 数据库管理系统    4、 开发    5、 数据字典    6、 a=14    7、 0

8、 m=4 k=4 i=5 k=5    9、 #&    10、 9    11、 1    12、 5 4    13、 0    14、 i=1    15、 x[i-1]

16、 \*pmax=\*px    17、 10    18、 40    19、 struct DATE d={2006,10,1}    20、 a

2006年 9 月

## 一、选择题

- (1) 下列选项中不符合良好程序设计风格的是 \_\_\_\_\_。
- A)、源程序要文档化 B)、数据说明的次序要规范化  
C)、避免滥用 goto 语句 D)、模块设计要保证高耦合，高内聚
- (2) 从工程管理角度，软件设计一般分为两步完成，它们是 \_\_\_\_\_。
- A)、概要设计与详细设计 B)、数据设计与接口设计  
C)、软件结构设计与数据设计 D)、过程设计与数据设计
- (3) 下列选项中不属于软件生命周期开发阶段任务的是 \_\_\_\_\_。
- A)、软件测试 B)、概要设计 C)、软件维护 D)、详细设计
- (4) 在数据库系统中，用户所见的数据模式为 \_\_\_\_\_。
- A)、概念模式 B)、外模式 C)、内模式 D)、物理模式
- (5) 数据库设计的四个阶段是：需求分析、概念设计、逻辑设计和 \_\_\_\_\_。
- A)、编码设计 B)、测试阶段 C)、运行阶段 D)、物理设计
- (6) 设有如下三个关系表下列操作中正确的是
- A)、 $T=R \cap S$  B)、 $T=R \cup S$  C)、 $T=R \times S$  D)、 $T=R/S$
- (7) 下列叙述中正确的是
- A)、一个算法的空间复杂度大，则其时间复杂度也必定大  
B)、一个算法的空间复杂度大，则其时间复杂度必定小  
C)、一个算法的时间复杂度大，则其空间复杂度必定小  
D)、上述三种说法都不对
- (8) 在长为 64 的有序线性表中进行顺序查找，最坏情况下需要比较的次数为 \_\_\_\_\_。
- A)、63 B)、64 C)、6 D)、7
- (9) 数据库技术的根本目标是要解决数据的 \_\_\_\_\_。
- A)、存储问题 B)、共享问题 C)、安全问题 D)、保护问题
- (10) 对下列二叉树：进行中序遍历的结果是 \_\_\_\_\_。
- A)、ACBDFEG B)、ACBDFGE C)、ABDCGEF D)、FCADBEG
- (11) 下列有关内联函数的叙述中，正确的是 \_\_\_\_\_。
- A)、内联函数在调用时发生控制转移  
B)、内联函数必须通过关键字 inline 来定义  
C)、内联函数是通过编译器来实现的  
D)、内联函数体的最后一条语句必须是 return 语句
- (12) 下列情况中，不会调用拷贝构造函数的是 \_\_\_\_\_。
- A)、用一个对象去初始化同一类的另一个新对象时  
B)、将类的一个对象赋予该类的另一个对象时  
C)、函数的形参是类的对象，调用函数进行形参和实参结合时  
D)、函数的返回值是类的对象，函数执行返回调用时
- (13) 下列有关继承和派生的叙述中，正确的是 \_\_\_\_\_。
- A)、如果一个派生类私有继承其基类，则该派生类对象不能访问基类的保护成员  
B)、派生类的成员函数可以访问基类的所有成员  
C)、基类对象可以赋值给派生类对象  
D)、如果派生类没有实现基类的一个纯虚函数，则该派生类是一个抽象类

(14) 下列运算不能重载为友元函数的是\_\_\_\_\_。

A)、=()[]-> B)、+-++--C)、><>=<= D)、+=-=\*=/=

(15) 关于在调用模板函数时模板实参的使用，下列表述正确的是\_\_\_\_\_。

A)、对于虚类型参数所对应的模板实参，如果能从模板函数的实参中获得相同的信息，则都可以省略

B)、对于虚拟类型参数所对应的模板实参，如果它们是参数表中的最后的若干个参数，FC EA DBG 则都可以省略

C)、对于虚拟型参数所对应的模板实参，若能够省略则必须省略

D)、对于常规参数所对应的模板实参，任何情况下都不能省略

(16) 下列关于输入流类成员函数 getline() 的描述中，错误的是\_\_\_\_\_。

A)、该函数是用来读取键盘输入的字符串的

B)、该函数读取的字符串长度是受限制的

C)、该函数读取字符串时，遇到终止符便停止

D)、该函数读取字符串时，可以包含空格

(17) 下列符号中，正确的 C++ 标识符是\_\_\_\_\_。

A)、enum B)、2b C)、foo-9 D)、\_32

(18) 下列语句中，错误的是\_\_\_\_\_。

A)、const int buffer=256; B)、const double\*point;

C)、int const buffer=256; D)、double\*const point;

(19) if 语句的语法格式可描述为

格式 1：if (<条件>) <语句>或

格式 2：if (<条件>) <语句 1>else <语句 2>

关于上面的语法格式，下列表述中错误的是

A)、<条件>部分可以是一个 if 语句，例如 if(if(a=0)?)?

B)、<条件>部分可以是一个 if 语句，例如 if(?)if(?)?

C)、如果在<条件>前加上逻辑非运算符！并交换<语句 1>和<语句 2>的位置，语句功能不变

D)、<语句>部分可以是一个循环语句，例如 if(?)while(?)?

(20) 有如下说明

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

则数值为 9 的表达式是\_\_\_\_\_。

A)、\*p+9` B)、\*(p+8) C)、\*p+=9 D)、p+7

(21) 或下面的函数调用

fun(a+b,3,max(n-1)b)

则 fun 的实参个数是\_\_\_\_\_。

A)、3 B)、4 C)、5 D)、6

(22) 以下关键字不能用来声明类的访问权限的是\_\_\_\_\_。

A)、public B)、static C)、protected D)、private

(23) 在公有继承的情况下，允许派生类直接访问的基类成员包括\_\_\_\_\_。

A)、公有成员 B)、公有成员和保护成员

C)、公有成员、保护成员和私有成员 D)、保护成员

(24) 关于运算符重载，下列表述中正确的是\_\_\_\_\_。

A)、C++已有的任何运算符都可以重载

B)、运算符函数的返回类型不能声明为基本数据类型

C)、在类型转换函数的定义中不需要声明返回类型

D)、可以通过运算符重载来创建 C++中原来没有的运算符

(25) 关于关键字 class 和 typename, 下列表述中正确的是\_\_\_\_\_。

A)、程序中的 typename 都可以替换为 class

B)、程序中的 class 都可以替换为 typename

- C)、在模板形参表中只能用 typename来声明参数的类型  
D)、在模板形参表中只能用 class 或 typename来声明参数的类型

( 26) 有如下程序

```
#include  
#include  
using namespace std;  
int main(){  
count    return 0;  
}
```

若程序的输出是：

**\*\*12.345\*\*34.567**

则程序中下划线处遗漏的操作符是\_\_\_\_\_。

- A)、setprecision(3) B)、fixed C)、setfill(\*) D)、stew(8)

( 27) 有如下程序

```
#include  
#include  
using namespace std;  
class MyClass{  
public:  
MyClass(){cout<<?A  ” ;}  
MyClass(char c {cout<~MyClass(){sout<<?B  ” ;}  
};  
Int main(){  
MyClassp1,*p2  
p2=new MyClass(,X?);  
delete p2;  
return 0;  
}
```

执行这个程序幕上将显示输出\_\_\_\_\_。

- A)、ABX B)、ABXB C)、AXB D)、AXBB

( 28) 有如下程序

```
#include  
using namespace std;  
int i=1;  
class Fun{  
public;  
static int i;  
int value(){return i-1}  
int value ()const{return i+1;}  
};  
int Fun;;i=2;  
int main(){  
int i=3;  
Fun Fun1  
const Fun fun2;  
return 0;
```

```
}
```

若程序的输出结果是： 123

则程序中下划线处遗漏的语句是\_\_\_\_\_。

A )、cout<B) 、 cout<C) 、 cout<D) 、 cout<

( 29) 有如下程序

```
#include
using namespace std;
class Obj{
static int i;
public:
Obj(){i++;}
~Obj(){i--;}
static int getVal(){return i;}
};
int Obj::i=0;
void f () {Obj ob2;cout    Obj obl;
F();
Obj*ob3=new Obj;cout    Delete ob3;cout    }
```

程序的输出结果是\_\_\_\_\_。

A ) 、 232 B ) 、 231 C ) 、 222 D ) 、 221

( 30) 有如下程序

```
#include
using namespace std;
class Base{
protected;
Base(){cout<<"A?";}
Base(char c){cout<<c;}
class Derived; public Base{
public:
Derived(char c ){cout <<c;}
int main(){
Derived d1(,"B?");
return 0;
}
```

执行这个程序屏幕上将显示输出\_\_\_\_\_。

A ) 、 B B ) 、 BA C ) 、 AB D ) 、 BB

( 31) 有如下类定义：

```
class MyBase{
int k;
public;
MyBase(int n=0):k(n){}
int value()const{return k;}
};
class MyDerived; MyBase{
int j;
public;
```

```
MyDerived(int i): j(i){
int getK()const {return k;}
int getj()const{return j;}
};
```

编译时发现有一处语法错误，对这个错误最佳准确的描述是

- A )、函数 getK 试图访问基类的私有成员变量 K
- B )、在类 MyDerived 的定义中，基类名 MyBase 前缺少关键字 public、protected 或 private
- C )、类 MyDerived 缺少一个无参的构造函数
- D )、类 MyDerived 的构造的数没有对基数数据成员 K 进行初始化

( 32) 在一个派生类对象结束其生命周期时

- A )、先调用派生类的析构函数后调用基类的析构函数
- B )、先调用基类的析构函数后调用派生类的析构函数
- C )、如果基数没有定义析构函数，则只调用派生类的析构函数
- D )、如果派生类没有定义析构函数，则只调用基类的析构函数

( 33) 有如下的运算重载函数定义：

```
Double operator+(int i, int k){return double(i+k);}
```

但定义有错误，对这个错误的最准确的描述是 \_\_\_\_\_。

- A )、+只能作为成员函数重载，而这里的+是作为非成员函数重载的
- B )、两个 int 型参数的和应该是 int 型，而这里将+的返回类型声明为 double
- C )、没有将运算符重载函数声明为某个类的友元
- D )、C++已经提供了求两个 int 型数据之和运算符+，不能再定义同样的运算符

( 34) 语句 ofstream( “ SALARY.DAT ,ios\_base::app的功能是建立流对象 f，并试图打开文件 SALARY.DAT 与 f 关联，而且 \_\_\_\_\_。

- A )、若文件存在，将其置为空文件；若文件不存在，打开失败
- B )、若文件存在，将文件指针定位于文件尾；若文件不存在，建立一个新文件
- C )、若文件存在，将文件指针定位于文件首；若文件不存在，打开失败
- D )、若文件存在，打开失败；若文件不存在，建立一个新文件

( 35) 有如下程序

```
#include
using namespace std;
class A{
public:
virtual Void func1(){cout<<      “ A1 ” ;}
void func2(){cout<<      “ A2 ” ;}
};
class B: public A{
public:
void func1(){cout<<      “ B1 ” ;}
void func2(){cout<<      “ B2 ” ;}
};
Int main(){
A*p=new B;
p->func1();
p->func2();
return 0;
}
```



运行程序，屏幕上将显示输出\_\_\_\_\_。

A )、B1B2 B )、A1A2 C )、B1A2 D )、A1B2

## 二、填空题

( 1 ) 下列软件系统结构图的宽度为 [1] 。

( 2 ) [2] 的任务是诊断和改正程序中的错误。

( 3 ) 一个关系表的行为 [3] 。

( 4 ) 按 ‘先进后出’ 原则组织数据的数据结构是 [4] 。

( 5 ) 数据结构分为线性结构和非线性结构，带链的队列属于 [5] 。

( 6 ) 若有定义语句：int\*a,b;，则变量 b 的数据类型是 [6] 。

( 7 ) 已知数组 a 中的元素个数为 n，下列语句的作用是将下标为 i 的元素移动到下标为 i - 1 的单元，其中  $1 \leq i \leq n$ 。例如，当 n=4，a 中原有数据为 1，2，3，4 时，则移动后 a 中元素为 2,3,4,4。请将语句补充完整：

```
For(int i=0;i
```

( 8 ) 已知递归函数 f 的定义如下：

```
int f (int n)
```

```
{
```

```
if(n<=1)return 1;/递归结束情况
```

```
else return n*f(n-2);/递归} 则函数调用语句 f(5)的返回值是 [8] 。
```

( 9 ) 创建对象数组时，对数组的每一个元素都将调用一次构造函数，如果没有显式给出数组元素的初值，则调用缺省构造函数，下列程序涉及到对象数组的创建和单个对象的创建，其输出结果是 [9] 。

```
#include
```

```
using namespace std;
```

```
class Foo{
```

```
public:
```

```
Foo(int X){cout <<"A    " }
```

```
Foo(){}
```

```
};
```

```
int main()
```

```
Foo f[3],g(3);
```

```
return 0;
```

```
}
```

( 10 ) 习惯性在下列程序的输出结果是 42，请将画线处缺失的部分补充完整。

```
#include
```

```
Using namespace atd;
```

```
Class Foo{
```

```
int value;
```

```
public:
```

```
Foo():value(0){}
```

```
Void setvalue(int value)
```

```
{ [10] =value; //给 Foo 的数据成员 value 赋值}void print(){cout<<};
```

```
Int main()
```

```
{
```

```
Foo f;
```

```
f.setvalue(42);
```

```
f.print();
```

```
return 0;
}
```

( 11 ) 如果不能使用多态机制，那么通过基类的指针虽然可以指向派生类对象，但是只能访问从基数继承的成员，下列程序没有使用多态机制，其输出结果是 [11] 。

```
#include
using namespace std;
class Base{
public:
Void print(){cout<< ,B?;}};
class Derived: public Base{
public
void print(){cout<< ,D?;}}
int main()
{
Derived*pd=new Dreived);
Base*pb=pd;
pb->print();
pd->print();
delete pd;
return 0;
}
```

( 12 ) 在声明派生类时，如果不显式地给出继承方式，缺省的类继承方式是私有继承

private。已知有如下类定义：

```
class Base{
protected:
void fun(){}
};
Class Derived: Base{;
```

则 Base 类中的成员函数 fun()，在 Derived 类中的访问权限是 [12] （注意：要求填写 private、protected 或 public 中的一项）。

( 13 ) 在 MyClass 类的定义中，对赋值运算符=进行重载。请将画线处缺失的部分补充完整。

```
[13] MyClass::operator=(const MyClass rhs)
{
if(this=&rhs)return *this;
value=rhs. value;
return *this;
}
```

( 14 ) 插入排序算法的主要思想是：每次从未排序序列中取出一个数据，插入到已排序序列中的正确位置， InsertSort 类的成员函数 sort() 实现了插入排序算法，请将画线处缺失的部分补充完整。

```
class InsertSort{
public:
InsertSort(int*a0,int n0):a(a0),n(n0){参数组首地址，n 是数组元素个数
void sort()
{//此函数假设已排离序列初始化状态只包含 a[0]，未排序序列初始为 a[1]?a[n-1]
for (int i=1;i<n;i++){
for( [14] j>0;--j){
```

```

if(t>a[j-1])break;
a[j]=a[j-1];}
a[j]=t;}}
protected:
int*a,n;//指针 a 用于存放数组首地址 , n 用于存放数组元素个数
}

```

( 15) 下列程序的输出结果是 [15]

```

#include
using namespace std;
class A{
int a
public:
A():a(9){}
virtual void print() const {cout<<}
class B:public A{
char b;
public:
B(){b= ,S?;}
void print()const{cout<<}
void show(Aa&X){X.print()}
int main()
{ Ad1;*p;
Bd2;
p=&d2;
d1,print();
d2,print();
p->print();
show(d1);
show(d2);
return 0;}

```

01—05 DACBD      06—10 CDBBA      11—15 ADDAD      16—20 CCBBC  
21—25 DACBC      26—30 ABCDC      31—35 ADBAD      36—40 BBADC  
41—45 CBBCD      46—50 BABDC

1、 3      2、 调试      3、 元组      4、 栈      5、 线性      6、 12346.0      7、 10 11  
8、 1      9、 7 4      10、 n=n/10      11、 (x+8)      12、 sin(x)  
13、 0      14、 i%2==0      15、 s--      16、 \*s++      17、 18      18、 16 11  
19、 ->data      20、 " rb "

2006年 4 月

## 一选择题

1) 下列选项中不属于结构化程序设计方法的是

A) 自顶向下      B) 逐步求精      C) 模块化      D) 可复用

2) 两个或两个以上模块之间关联的紧密程度称为

A) 耦合度 B) 内聚度 C) 复杂度 D) 数据传输特性

3) 下列叙述中正确的是

A) 软件测试应该由程序开发者来完成 B) 程序经调试后一般不需要再测试

C) 软件维护只包括对程序代码的维护 D) 以上三种说法都不对

4) 按照 '后进先出' 原则组织数据的数据结构是

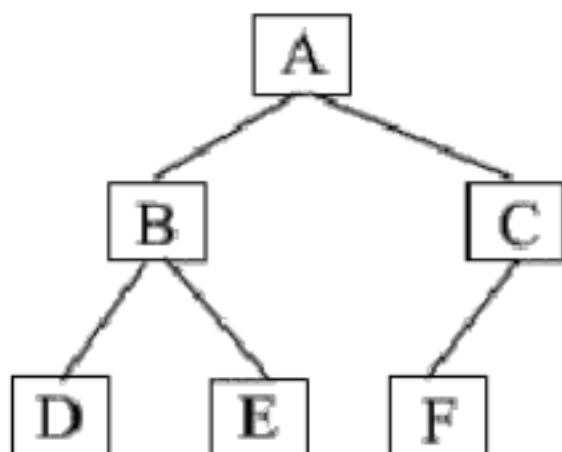
A) 队列    B) 栈 C) 双向链表 D) 二叉树

5) 下列叙述中正确的是

A) 线性链表是线性表的链式存储结构 B) 栈与队列是非线性结构

C) 双向链表是非线性结构 D) 只有根结点的二叉树是线性结构

6) 对如下二叉树



进行后序遍历的结果为

A) ABCDEF      B) DBEAFC      C) ABDECF      D) DEBFCA

7) 在深度为 7 的满二叉树中，叶子结点的个数为

A) 32 B) 31 C) 64 D) 63

8) '商品' 与 '顾客' 两个实体集之间的联系一般是

A) 一对一    B) 一对多    C) 多对一    D) 多对多

9) 在 E-R 图中，用来表示实体的图形是

A) 矩形 B) 椭圆形 C) 菱形 D) 三角形

10) 数据库 DB，数据库系统 DBS，数据库管理系统 DBMS 之间的关系是

A) DB 包含 DBS 和 DBMS

B) DBMS 包含 DB 和 DBS

C) DBS 包含 DB 和 DBMS

D) 没有任何关系

11) 以下不合法的用户标识符是

A) j2\_KEY      B) Double      C) 4d      D) \_8\_

12) 以下不合法的数值常量是

- A) 011      B) 1e1      C) 8.0 E0.5      D) 0xabcd

13) 以下不合法的字符常量是

- A) '\018'      B) '\ '      C) '\\ '      D) ', \xcc'

14) 表达式  $3.6 - 5/2 + 1.2 + 5\%2$  的值是

- A) 4.3      B) 4.8      C) 3.3      D) 3.8

15) 以下能正确定义字符串的语句是

- A) char str[]={,    064'}; B) char str=    x43 " " ;  
C) char str=    " " ; D) char str[]={0 " " ;    " "

16) 以下数组定义中错误的是

- A) int x[][3]={0}; B) int x[2][3]={{1,2},{3,4},{5,6}};  
C) nt x[][3]={{1,2,3},{4,5,6}}; D) int x[2][3]={1,2,3,4,5,6};

17) 若要求从键盘读入含有空格字符的字符串，应使用函数

- A) getc()      B) gets()      C) getchar()      D) scanf()

18) 下四个程序中，完全正确的是

- |   |  |
|---|--|
| A) #include<br>main();<br>{/*programming*/<br>printf(    " programming!" );}  | B) #include<br>main()<br>{/*/programming*/<br>printf(    " programming!" );}   |
| C) #include<br>main()<br>{/*/*programming*/<br>printf(    " programming!" );} | D) include<br>main()<br>*/ { /*programming*/<br>printf(    " programming!" );} |

19) 若有定义：float x=1.5; int a=1,b=3,c=2; 则正确的 switch 语句是

- |  |  |
|--|--|
| A) switch(x)<br>{case 1.0: printf(    n " "); *<br>Case 2.0: printf(    n "" );*}  | B) switch((int)x);<br>{case 1: printf(    n " ");*<br>case 2: printf(    "n** ");} |
| C) switch(a+b)<br>{ case 1: printf(    n " "); *<br>case 2+1: printf(    n "" );*} | D) switch(a+b)<br>case c: printf(    n "" );*}                                     |

20) 若各选项中所用变量已正确定义，函数 fun 中通过 return 语句返回一个函数值，以下选项中错误的程序是

- |   |   |
|---|---|
| A) main()<br>{ ..... x=fun(2,10);    ..... }<br>float fun(int a,int b){<br>C) float fun(int,int);<br>main()<br>{ ..... x=fun(2,10);    ..... }<br>float fun(int a,int b){ | B) float fun(int a,int b){<br>..... }<br>main()<br>{ ..... x=fun(i,j);    ..... }<br>D) main()<br>{ float fun(int i,int j);<br>..... x=fun(i,j);    ..... }<br>float fun(int a,int b){<br>..... } |
|---|---|

21) 在以下给出的表达式中，与 while(E) 中的 (E) 不等价的表达式是

- A) (!E=0)      B) (E>0||E<0)      C) (E==0)      D) (E!=0)

22) 要求通过 while 循环不断读入字符，当读入字母 N 时结束循环。若变量已正确定义，以下正确的程序段是

- A) while((ch=getchar())!='N') printf(    " %c" ,ch);  
B) while(ch=getchar()!='N') printf(    " %c" ,ch);  
C) while(ch=getchar()=='N') prin    tf(    " %c" ,ch);  
D) while((ch=getchar())=='N') printf(    " %c" ,ch);

23) 已定义以下函数

```
int fun(int *p)
{return *p;}
```

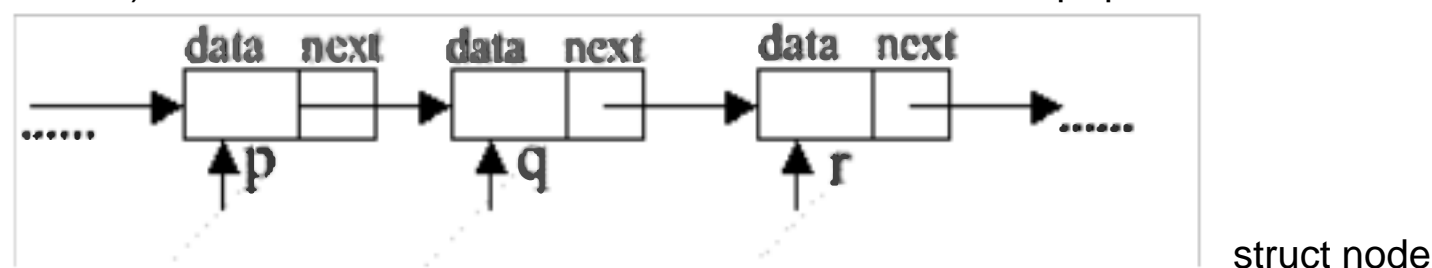
fun 函数返回值是

- A) 不确定的值      B) 一个整数      C) 形参 p 中存放的值      D) 形参 p 的地址值

24) 若有说明语句：double \*p,a; 则能通过 scanf 语句正确给输入项读入数据的程序段是

- A) \*p=&a; scanf(      “ %lf ” ,p) B) \*p=&a; scanf(      “ %f ” ,p);  
C) p=&a; scanf(      “ %lf ” ,\*p) D) p=&a; scanf(      “ %lf ” ,p);

25) 现有以下结构体说明和变量定义，如图所示，指针 p,q,r 分别指向一个链表中连续三个结点。



```
{
char data;
struct node *next;
}*p,*q,*r;
```

现要将 q 和 r 所指结点交换前后位置，同时要保持链表的连续，以下不能完成此操作的语句是

- A) q->next=r->next; p->next=r; r->next=q;  
B) p->next=r; q->next=r->next; r->next=q;  
C) q->next=r->next; r->next=q; p->next=r;  
D) r->next=q; p->next=r; q->next=r->next;

26) 有以下程序段

```
struct st
{ int x; int *y;}*pt;
int a[]={1,2} , b[]={3,4};
struct st c[2]={10,a,20,b};
pt=c;
```

以下选项中表达式的值为 11 的是

- A) \*pt->y      B) pt->x      C) ++pt->x      D) (pt++)->x

27) 设 fp 为指向某二进制文件的指针，且已读到此文件末尾，则函数 feof ( fp ) 的返回值为

- A) EOF B) 非 0 值 C) 0 D) NULL

28) 设有以下语句

```
int a=1,b=2,c;
c=a^(b<<2);
执行后，c 的值为
```

- A) 6 B) 7 C) 8 D) 9

29) 有以下程序

```
#include
main()
{
char c1,c2,c3,c4,c5,c6;
scanf(      “ %c%c%c%c” ,&c1,&c2,&c3,&c4);
c5=getchar(); c6=getchar();
putchar(c1); putchar(c2);
printf(      “ %d%c” ,c5,c6);
```

```
}
```

程序运行后，若从键盘输入（从第 1 列开始）

123< 回车>

45678< 回车>

则输出结果是

A ) 1267 B ) 1256 C ) 1278 D ) 1245

30 ) 若有以下程序

```
main()
```

```
{int y=10;
```

```
while(y- - ); printf(      “ y#%dy);
```

```
}
```

程序运行后的输出结果是

A)y=0 B)y=-1 C)y=1 D)while 构成无限循环

31 ) 有以下程序

```
main()
```

```
{
```

```
int a=0,b=0,c=0,d=0;
```

```
if(a=1) b=1;c=2;
```

```
else d=3;
```

```
printf(      “ %d,%d,%d,%d ” ,a,b,c,d);
```

```
}
```

程序输出

A ) 0,1,2,0 B) 0,0,0,3 C)1,1,2,0 D) 编译有错

32 ) 有以下程序

```
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 ” ,x);
```

```
}
```

程序执行后的输出结果是

A ) x=4 B) x=8 C) x=6 D) x=12

33) 有以下程序

```
int fun1(double a){return a*=a;}
```

```
int fun2(double x,double y)
```

```
{
```

```
double a=0,b=0;
```

```
a=fun1(x); b=fun1(y); return (int)(a+b);
```

```

}
main()
{double w; w=fun2(1.1,2.0);          ..... }

```

程序执行后变量 w 中的值是

A ) 5.21 B) 5 C) 5.0 D) 0.0

34) 有以下程序

```

main()
{
int i,t[][3]={9,8,7,6,5,4,3,2,1};
for(i=0;i<3;i++) printf(          ,t[2-i][i]%d"
} 程序的执行后的输出结果是

```

A ) 7 5 3 B ) 3 5 7 C ) 3 6 9 D ) 7 5 1

35 ) 有以下程序

```

fun(char p[][10])
{int n=0,i;
for(i=0;i<7;i++)
if(p[i][0]=='T') n++;
return n;
}

```

main()

```

{
char str[][10]={          “ Mon ” , “ Tue ” , “ Wed ” , “ Thu ” , “ Fri ” , “ Sat ” , “ Sun ” };
printf(          “ %d” ,fun(str));
} 程序执行后的输出结果是

```

A ) 1 NB ) 2C ) 3 D ) 0

36) 有以下程序

```

main()
{
int i,s=0,t[]={1,2,3,4,5,6,7,8,9};
for(i=0;i<9;i+=2) s+=*(t+i);
printf(          “ %d” ,s);
} 程序执行后的输出结果是

```

A ) 45 B ) 20 C ) 25 D ) 36

37 ) 有以下程序

```

void fun1(char *p)
{
char *q;
q=p;
while(*q!='\0')
{ (*q)++; q++; }
}

```

main()

```

{ char a[]={          “ Program ” },*p;
p=&a[3]; fun1(p); printf(          n ” “,a)%s
} 程序执行后的输出结果是

```

A ) P rohsbn B)PrphsbnC)Progsbn D)Program



38) 有以下程序

```
void swap(char *x,char *y)
{
char t;
t=*x; *x=*y; *y=t;
}
main()
{
char *s1= " abc " ,*s2= " 123 " ;
swap(s1,s2); printf( " %s,%s" ,s1,s2);
} 程序执行后的输出结果是
```

A ) 123 , abc B) abc,123 C) 1bc,a23 D) 321,cba

39 ) 有以下程序

```
int fun(int n)
{
if(n==1) return 1;
else
return (n+fun(n-1));
}
main()
{ int x;
scanf( " %d" ,&x); x=fun(x); printf( " %d\n" ,x);
} 程序执行时 , 给变量 x 输入 10 , 程序的输出结果是
```

A ) 55 B) 54 C) 65 D) 45

40) 有以下程序

```
int fun(int x[],int n)
{static int sum=0,i;
for(i=0;i<n;i++) sum+=x[i];
return sum;
}
main()
{int a[]={1,2,3,4,5},b[]={6,7,8,9},s=0;
s=fun(a,5)+fun(b,4); printf( " %d\n" ,s);
} 程序执行后的输出结果是
```

A ) 45 B ) 50 C ) 60 D ) 55

41 ) 有以下程序

```
main()
{
union {
char ch[2];
int d;
}s;
s.d=0x4321;
printf( " %x,%x\n" ,s.ch[0],s.ch[1]);
} 在 16 位编译系统上 , 程序执行后的输出结果是
```

A ) 21 , 43 B ) 43 , 21 C ) 43 , 00 D ) 21 , 00

42 ) 有以下程序

```
main()
{
char *p[]={      “ 3697 ” , “ 2584 ” };
int i,j; long num=0;
for(i=0;i<2;i++)
{
j=0;
while(p[i][j]!='\0')
{ if((p[i][j]-      ,0')%2)num=10*num+p[i][j]      - ,0';
j+=2;
} }
printf(      “ %d\n” num);
} 程序执行后的输出结果是
```

A ) 35 B ) 37 C ) 39 D ) 3975

43 ) 执行以下程序后 , test.txt 文件的内容是 ( 若文件能正常打开 )

```
#include
main()
{ FILE *fp;
char *s1=      “ Fortran ” , *s2=      “ Basic ” ;
if((fp=fopen(      “ test.txt ” , “ wb ” ))==NULL)
{ printf(      “ Can't open test.txt file ” ); exit(1);}
fwrite(s1,7,1,fp); /*      把从地址 s1 开始的 7 个字符写到 fp 所指文件中 */
fseek(fp,0L,SEEK_SET); /*      文件位置指针移到文件开头 */
fwrite(s2,5,1,fp);
fclose(fp);
}
```

A)Basican B)BasicFortran C)Basic D)FortranBasic

44) 以下叙述中错误的是

- A ) C 语言源程序经编译后生成后缀为 .obj 的目标程序
- B ) C 语言经过编译、连接步骤之后才能形成一个真正可执行的二进制机器指令文件
- C ) 用 C 语言编写的程序称为源程序 , 它以 ASCII 代码形式存放在一个文本文件中
- D ) C 语言的每条可执行语句和非执行语句最终都将被转换成二进制的机器指令

45 ) 以下叙述中错误的是

- A ) 算法正确的程序最终一定会结束 B ) 算法正确的程序可以有零个输出
- C ) 算法正确的程序可以有零个输入 D ) 算法正确的程序对于相同的输入一定有相同的结果

46 ) 以下叙述中错误的是

- A ) C 程序必须由一个或一个以上的函数组成 B ) 函数调用可以作为一个独立的语句存在
- C ) 若函数有返回值 , 必须通过 return 语句返回 D ) 函数形参的值也可以传回对应的实参

47 ) 设有以下定义和语句

```
char str[20]=      “ Program ” , *p;
p=str;
```

则以下叙述中正确的是

- A ) \*p 与 str[0] 中的值相等 B ) str 与 p 的类型完全相同
- C ) str 数组长度和 p 所指向的字符串长度相等
- D ) 数组 str 中存放的内容和指针变量 p 中存放的内容相同

48 ) 以下叙述中错误的是

- A) C 程序中的#include 和#define 行均不是 C 语句
- B) 除逗号运算符外，赋值运算符的优先级最低
- C) C 程序中，j++ ；是赋值语句
- D) C 程序中，+、-、\*、/、% 号是算术运算符，可用于整型和实型数的运算

49) 以下叙述中正确的是

- A) 预处理命令行必须位于 C 源程序的起始位置
- B) 在 C 语言中，预处理命令行都以“#”开头
- C) 每个 C 程序必须在开头包括预处理命令行：#include
- D) C 语言的预处理不能实现宏定义和条件编译的功能

50) 以下叙述中错误的是

- A) 可以通过 typedef 增加新的类型
- B) 可以用 typedef 将已存在的类型用一个新的名字来代表
- C) 用 typedef 定义新的类型名后，原有类型名仍有效
- D) 用 typedef 可以为各种类型起别名，但不能为变量起别名

## 二：填空题（每空2分，共40分）

- 1) 对长度为 10 的线性表进行冒泡排序，最坏情况下需要比较的次数为 \_\_\_\_\_[1]\_\_\_\_\_。
- 2) 在面向对象方法中，\_\_\_\_\_ [2]\_\_\_\_\_ 描述的是具有相似属性与操作的一组对象。
- 3) 在关系模型中，把数据看成是二维表，每一个二维表称为一个 \_\_\_\_\_ [3]\_\_\_\_\_。
- 4) 程序测试分为静态分析和动态测试，其中 \_\_\_\_\_ [4]\_\_\_\_\_ 是指不执行程序，而只是对程序文本进行检查，通过阅读和讨论，分析和发现程序中的错误。
- 5) 数据独立性分为逻辑独立性与物理独立性，当数据的存储结构改变时，其逻辑结构可以不变，因此，基于逻辑结构的应用程序不必修改，称为 \_\_\_\_\_ [5]\_\_\_\_\_。
- 6) 若变量 a,b 已定义为 int 类型并赋值 21 和 55，要求用 printf 函数以 a=21,b=55 的形式输出，请写出完整的输出语句 \_\_\_\_\_ [6]\_\_\_\_\_。
- 7) 以下程序用于判断 a,b,c 能否构成三角形，若能输出 YES，若不能输出 NO。当 a,b,c 输入三角形三条边长时，确定 a,b,c 能构成三角形的条件是需要同时满足三条件：a+b>c,a+c>b,b+c>a \_\_\_\_\_。请填空。

```
main()
{
    float a,b,c;
    scanf(    " %f%f%f " ,&a,&b,&c);
    if(____[7]____)printf(    n " YES,a,b,c 能构成三角形*/
    else printf(    n " NO,a,b,c 不能构成三角形*/
}
```

- 8) 以下程序的输出结果是 \_\_\_\_\_ [8]\_\_\_\_\_

```
main()
{ int a[3][3]={1,2,9},{3,4,8},{5,6,7}},i,s=0;
  for(i=0;i<3;i++) s+=a[i][i]+a[i][3-i-1];
  printf(    " r%d",s);
}
```

- 9) 当运行以下程序时，输入 abcd，程序的输出结果是： \_\_\_\_\_ [9]\_\_\_\_\_。

```
insert(char str[])
{ int i;
  i=strlen(str);
  while(i>0)
  { str[2*i]=str[i]; str[2*i-1]='*';i--;}
}
```

```
printf(    " r%s",str);
}
main()
{char str[40];
scanf(    " %s" ,str ); insert(str);
}
```

10) 以下程序的运行结果是：\_\_[10]\_\_

```
fun(int t[],int n)
{ int i,m;
if(n==1) return t[0];
else
if(n>=2) {m=fun(t,n-1); return m;}
}
main()
{
int a[]={11,4,6,3,8,2,3,5,9,2};
printf(    " r%d",fun(a,10));
}
```

11) 现有两个 C 程序文件 T18.c 和 myfun.c 同时在 TC 系统目录（文件夹）下，其中 T18.c 文件如下：

```
#include
#include    " myfun.c    "
main()
{ fun(); printf(    " n" );}
myfun.c 文件如下：
void fun()
{ char s[80],c; int n=0;
while((c=getchar())!='\n') s[n++]=c;
n--;
while(n>=0) printf(    " %d)",s[n]
}
```

当编译连接通过后，运行程序 T18 时，输入 Thank! 则输出的结果是：\_\_[11]\_\_。

12) 以下函数 fun 的功能是返回 str 所指字符串中以形参 c 中字符开头的后续字符串的首地址，例如：str 所指字符串为 Hello!, c 中的字符为 e，则函数返回字符串：ello! 的首地址。若 str 所指字符串为空串或不包含 c 中的字符，则函数返回 NULL。请填空。

```
char *fun(char *str,char c)
{ int n=0;char *p=str;
if(p!=NULL)
while(p[n]!=c&& p[n]!='\0') n++;
if(p[n]=='\0' return NULL;
return (_[12]);
}
```

13) 以下程序的功能是：输出 100 以内（不含 100）能被 3 整除且个位数为 6 的所有整数，请填空。

```
main()
{ int i,j;
for(i=0;__[13]__;i++)
{ j=i*10+6;
if(__[14]__) continue;
}
```

```
printf(    “ %d ” ,j);
}}
```

14) 以下 isprime 函数的功能是判断形参 a 是否为素数，是素数，函数返回 1，否则返回 0，请填空

```
int isprime(int a)
{ int i;
for(i=2;i<=a/2;i++)
if(a%i==0) __[15]__;
__[16]__;
}
```

15 )以下程序的功能是输入任意整数给 n 后，输出 n 行由大写字母 A 开始构成的三角形字符阵列图形，例如，输入整数 5 时(注意：n 不得大于 10)，程序运行结果如下：

```
A B C D E
F G H I
J K L
M N
O
```

请填空完成该程序。

```
main()
{ int i,j,n; char ch='A';
scanf(    “ %d ” ,&n);
if(n<11)
{
for(i=1;i<=n;i++)
{ for(j=1;j<=n-i+1;j++)
{ printf(    “ %2c ” ,ch);
__[17]__;
}
    【18】
}
}
else printf(    “ n is too large!”);
printf(    n ” );
}
```

16) 以下程序中函数 fun 的功能是：构成一个如图所示的带头结点的单向链表，在结点数据域中放入了具有两个字符的字符串。函数 disp 的功能是显示输出该单链表中所有结点中的字符串。请填空完成函数 disp。

```
#include
typedef struct node /*    链表结点结构*/
{ char sub[3];
Struct node *next;
}Node;
Node fun(char s) /*    建立链表*/
{    .....    }
void disp(Node *h)
{
Node *p;
p=h->next;
```

```

While( 【19】 )
{
printf(      “ r%s”,p->sub); p=    【20】 ;}
}
main()
{   Node *hd;
hd=fun();
disp(hd);
printf(   n “ ” );   }

```

参考答案

一、选择题

1-10 : DADBA DCDAC 11-20:CCADD BBBCA 21-30:CABDD CDDDB31-40:DBCBB CACAC41-50:ACADB

DCDBA

二、填空题

1、45 2、类 3、关系 4、静态分析 5、物理独立性 6、printf("a=%d,b=%d",a,b)  
7、a+b>c&&a+c>b&&b+c>a 8、30 9、a\*b\*c\*d\* 10、11 11、ahT  
12、p+n 或 str+n 13、i<=9 或 i<10 14、j%3!=0 15、return 0  
16、return 1 17、ch=ch+1 18、printf("\n") 19、p!=NULL 20、p->next

2 0 1 0 年 3 月全国计算机二级 C语言考试真题

一、选择题

- ( 1 ) 下列叙述中正确的是
- A ) 对长度为 n 的有序链表进行查找，最坏情况下需要的比较次数为 n
  - B ) 对长度为 n 的有序链表进行对分查找，最坏情况下需要的比较次数为 ( n/2 )
  - C ) 对长度为 n 的有序链表进行对分查找，最坏情况下需要的比较次数为 ( log<sub>2</sub>n )
  - D ) 对长度为 n 的有序链表进行对分查找，最坏情况下需要的比较次数为 ( log<sub>2</sub>n )
- ( 2 ) 算法的时间复杂度是指
- A ) 算法的执行时间
  - B ) 算法所处理的数据量
  - C ) 算法程序中的语句或指令条数
  - D ) 算法在执行过程中所需要的基本运算次数
- ( 3 ) 软件按功能可以分为：应用软件、系统软件和支撑软件（或工具软件）。下面属于系统软件的是
- A ) 编辑软件
  - B ) 操作系统
  - C ) 教务管理系统
  - D ) 浏览器
- ( 4 ) 软件（程序）调试的任务是
- A ) 诊断和改正程序中的错误
  - B ) 尽可能多地发现程序中的错误
  - C ) 发现并改正程序中的所有错误
  - D ) 确定程序中错误的性质
- ( 5 ) 数据流程图（ D F D 图）是
- A ) 软件概要设计的工具
  - B ) 软件详细设计的工具
  - C ) 结构化方法的需求分析工具
  - D ) 面向对象方法的需求分析工具
- ( 6 ) 软件生命周期可分为定义阶段，开发阶段和维护阶段。详细设计属于
- A ) 定义阶段
  - B ) 开发阶段
  - C ) 维护阶段
  - D ) 上述三个阶段
- ( 7 ) 数据库管理系统中负责数据模式定义的语言是
- A ) 数据定义语言
  - B ) 数据管理语言
  - C ) 数据操纵语言
  - D ) 数据控制语言
- ( 8 ) 在学生管理的关系数据库中，存取一个学生信息的数据单位是
- A ) 文件
  - B ) 数据库
  - C ) 字段
  - D ) 记录
- ( 9 ) 数据库设计中，用 E - R 图来描述信息结构但不涉及信息在计算机中的表示，它属于数据库设计的
- A ) 需求分析阶段
  - B ) 逻辑设计阶段
  - C ) 概念设计阶段
  - D ) 物理设计阶段

( 1 0 ) 有两个关系 R 和 T 如下：

R		T
A	B	C
a	1	2
b	2	2
c	3	2

d	3	2
---	---	---

A	B	C
c	3	2
d	3	2

则由关系 R 得到关系 T 的操作是

- A ) 选择                      B ) 投影                      C ) 交                      D ) 并

( 1 1 ) 以下叙述正确的是

- A ) C 语言程序是由过程和函数组成的  
 B ) C 语言函数可以嵌套调用，例如： fun ( fun ( x ) )  
 C ) C 语言函数不可以单独编译  
 D ) C 语言中除了 main 函数，其他函数不可作为单独文件形式存在

( 1 2 ) 以下关于 C 语言的叙述中正确的是

- A ) C 语言中的注释不可以夹在变量名或关键字的中间  
 B ) C 语言中的变量可以在使用之前的任何位置进行定义  
 C ) 在 C 语言算术表达式的书写中，运算符两侧的运算数类型必须一致  
 D ) C 语言的数值常量中夹带空格不影响常量值的正确表示

( 1 3 ) 以下 C 语言用户标识符中，不合法的是

- A ) \_ 1                      B ) A a B c                      C ) a \_ b                      D ) a - - b

( 1 4 ) 若有定义：double a = 2.2 ; int i = 0 , k = 18 ; , 则不符合 C 语言规定的赋值语句是

- A ) a = a + + , i + + ;                      B ) i = ( a + k ) < = ( i + k ) ;  
 C ) i = a % 11 ;                      D ) i = ! a ;

( 1 5 ) 有以下程序

```
#include<stdio.h>
main()
{char a,b,c,d;
scanf( " %c%c", &a,&b);
c=getchar();d=getchar;
printf( " %c%c%c%c\n", a,b,c,d);}
```

当执行程序时，按下列方式输入数据（从第 1 列开始，< C R > 代表回车，注意：回车也是一个字符）

1 2 < C R >

3 4 < C R >

则输出结果是

- A ) 1 2 3 4                      B ) 1 2                      C ) 1 2                      D ) 1 2  
 3                      3 4

( 1 6 ) 以下关于 C 语言数据类型使用的叙述中错误的是

- A ) 若要准确无误差的表示自然数，应使用整数类型  
 B ) 若要保存带有多位小数的数据，应使用双精度类型  
 C ) 若要处理如“人员信息”等含有不同类型的相关数据，应自定义结构体类型  
 D ) 若只处理“真”和“假”两种逻辑值，应使用逻辑类型

( 1 7 ) 若 a 是数值类型，则逻辑表达式 ( a == 1 ) || ( a !                      = 1 ) 的值是

- A ) 1                      B ) 0  
 C ) 2                      D ) 不知道 a 的值，不能确定

( 1 8 ) 以下选项中与 if ( a==1 ) a=b ; else a++ ; 语句功能不同的 switch 语句是



```

A ) switch ( a )
{ case 1 : a = b ; break ;
  default : a++ ;
}

```

```

B ) switch ( a==1 )
{ case 0 : a = b ; break ;
  case 1 : a++ ;
}

```

```

C ) switch ( a )
{ default : a++ ; break ;
  case 1 : a = b ;
}

```

```

D ) switch ( a==1 )
{ case 1 : a = b ; break ;
  case 0 : a++ ;
}

```

( 1 9 ) 有如下嵌套的 i f 语句

```

if(a<b)
    if(a<c) k=a;
    else k=c;
if(b<c) k=b;
else k=c;

```

以下选项中与上述 i f 语句等价的语句是

- A ) k=(a<b)?a:b ; k= ( b < c ) ?b:c;
- B ) k=(a<b ) ?((b < c ) ? a:b ) ((b > c)?b:c) ;
- C ) k=(a<b ) ?((a < c ) ? a:c ) ((b < c)?b:c) ;
- D ) k=(a<b ) ?a:b ; k= ( a < c ) ?a:c ;

( 2 0 ) 有以下程序

```

#include<stdio.h>
main()
{int i,j,m=1;
 for(i=1;i<3;i++)
    {for(j=3;j>0;j--)
      {if(i+j>3) break;
       m*=i*j;}}
printf(    “ m=%d\n” ,m);
}

```

程序运行后的输出结果是

- A ) m = 6
- B ) m = 2
- C ) m = 4
- D ) m = 5

( 2 1 ) 有以下程序

```

#include<stdio.h>
main()
{int a=1,b=2;
 for(;a<8;a++) { b+=a;a+=2;}
printf(    “ %d,%d\n ” ,a,b);
}

```

}

程序运行后的输出结果是

- A ) 9 , 1 8                      B ) 8 , 1 1                      C ) 7 , 1 1                      D ) 1 0 , 1 4

( 2 2 ) 有以下程序，其中 k 的初值为八进制数

```
#include<stdio.h>
```

```
main()
```

```
{int k=011;
```

```
printf(    “ %d\n” ,k++);
```

```
}
```

程序运行后的输出结果是

- A ) 1 2                      B ) 1 1                      C ) 1 0                      D ) 9

( 2 3 ) 下列语句组中，正确的是

- A ) char \*s ; s= " Olympic " ;                      B ) char s[7] ; s= " Olympic " ;  
C ) char \*s ; s= { " Olympic " } ;                      D ) char s[7] ; s= { " Olympic " } ;

( 2 4 ) 以下关于 return 语句的叙述中正确的是

- A ) 一个自定义函数中必须有一条 return 语句  
B ) 一个自定义函数中可以根据不同情况设置多条 return 语句  
C ) 定义成 void 类型的函数中可以有带返回值的 return 语句  
D ) 没有 return 语句的自定义函数在执行结束时不能返回到调用处

( 2 5 ) 下列选项中，能正确定义数组的语句是

- A ) int num[0..2008] ;                      B ) int num[] ;  
C ) int N=2008 ;                      D ) #define N 2008  
int num[N] ;                      int num[N] ;

( 2 6 ) 有以下程序

```
#include<stdio.h>
```

```
void fun(char *c,int d)
```

```
{*c=*c+1;d=d+1;
```

```
printf(    “ %c,%c” ,*c,d);
```

```
}
```

```
main()
```

```
{char b= ‘ a ’ ,a= ‘ A ’ ;
```

```
fun(&b,a);printf(    “ %c,%c\n” , b,a);
```

```
}
```

程序运行后的输出结果是

- A ) b , B , b , A                      B ) b , B , B , A  
C ) a , B , B , a                      D ) a , B , a , B

( 2 7 ) 若有定义 int ( \*pt ) [3] ; , 则下列说法正确的是

- A ) 定义了基类型为 int 的三个指针变量  
B ) 定义了基类型为 int 的具有三个元素的指针数组 pt  
C ) 定义了一个名为 \*pt 、具有三个元素的整型数组  
D ) 定义了一个名为 pt 的指针变量，它可以指向每行有三个整数元素的二维数组

( 2 8 ) 设有定义 double a[10],\*s = a ; , 以下能够代表数组元素 a[3] 的是

- A ) (\*s ) [3]                      B ) \* ( s+3 )  
C ) \*s[3]                      D ) \*s+3

( 2 9 ) 有以下程序

```
#include<stdio.h>
```

```
main()
{int a[5]={1,2,3,4,5},b[5]={0,2,1,3,0},i,s=0;
  for(i=1;i<3;i++) s=s+a[b[i]];
  printf(    “ %d\n ” ,s);
}
```

程序运行后的输出结果是

A ) 6                      B ) 1 0                      C ) 1 1                      D ) 1 5

( 3 0 ) 有以下程序

```
#include<stdio.h>
main()
{int b[3][3]={0,1,2,0,1,2,0,1,2},i,j,t=1;
  for(i=1;i<3;i++)
    for(j=1;j<=1;j++) t+=b[i][b[j][i]];
  printf(    “ %d\n ” ,t);
}
```

程序运行后的输出结果是

A ) 1                      B ) 3                      C ) 4                      D ) 9

( 3 1 ) 若有以下定义和语句

```
char s1[10]   = " abcd! " ,   s2= "           " ;
printf ( " % d % d \ n " ,   strlen ( s1 ) , strlen ( s2 ) ) ;
```

则输出结果是

A ) 5      5                      B ) 1 0      5                      C ) 1 0      7                      D ) 5      8

( 3 2 ) 有以下程序

```
#include<stdio.h>
#define N 8
void fun(int *x,int i)
{*x=*x+i;}
main()
{int a[N]={1,2,3,4,5,6,7,8},i;
  fun(a,2); for(i=1;i<N/2;i++)
  {printf(    “ %d” ,a[i]);}
  printf(    “ \n ” );
}
```

程序运行后的输出结果是

A ) 1 3 1 3                      B ) 2 2 3 4                      C ) 3 2 3 4                      D ) 1 2 3 4

( 3 3 ) 有以下程序

```
#include<stdio.h>
int f(int t[],int n);
main()
{int a[4]={1,2,3,4},s;
  s=f(a,4);printf(    “ %d\n ” ,s);}
int f(int t[],int n)
{if (n>0)return t[n-1]+f(t,n-1);
else return 0;}
```

程序运行后的输出结果是

A ) 4                      B ) 1 0                      C ) 1 4                      D ) 6

( 3 4 ) 有以下程序

```
#include<stdio.h>

int fun()
{static int x=1;
  x*=2;return x;
}

main()
{int i,s=1;
  for(i=1;i<=2;i++) s=fun();
  printf(    “ %d\n ” ,s);
}
```

程序运行后的输出结果是

- A ) 0                      B ) 1                      C ) 4                      D ) 8

( 3 5 ) 有以下程序

```
#include<stdio.h>

#define SUB(a) (a)-(a)

main()
{int a=2,b=3,c=5,d;
  d=SUB(a+b)*c;
  printf(    “ %d\n ” ,d);
}
```

程序运行后的输出结果是

- A ) 0                      B ) - 1 2                      C ) - 2 0                      D ) 1 0

( 3 6 ) 设有定义：

```
struct complex
{int real,unreal;} data1={1,8},data2;
```

则以下赋值语句中错误的是

- A ) data2 = data1 ;                      B ) data2 = ( 2 , 6 ) ;  
C ) data2 . real = data1 . real ;                      D ) data2 . real = data1 . unreal ;

( 3 7 ) 有以下程序

```
#include<stdio.h>
#include<string.h>

struct A
{int a;char b[10];double c;};

void f(struct At);

main()
{struct Aa={1001,    “ ZhangDa” ,1098.0};f(a);
  printf(    “ %d,%s,%6.1f\n ” ,a.a,a.b,a.c);}

void f(struct At)
{t.a=1002;strcpy(t.b,    “ ChangRong” );t.c=1202.0;}
```

输出结果是

- A ) 1 0 0 1 , Z h a n g D a , 1 0 9 8 . 0  
B ) 1 0 0 2 , C h a n g R o n g , 1 2 0 2 . 0  
C ) 1 0 0 1 , C h a n g R o n g , 1 0 9 8 0  
D ) 1 0 0 2 , Z h a n g D a , 1 2 0 2 . 0

( 3 8 ) 有以下定义和语句

```
struct workers
{int num;char name[20];char c;
    struct
    {int day;int month;int year;}s
};
```

struct workers w,\*pw;

pw=&w;

能给w中 year 成员赋 1980的语句是

- A ) p w . y e a r = 1 9 8 0 ;      B ) w . y e a r = 1 9 8 0 ;  
C ) p w - > y e a r = 1 9 8 0 ;      D ) w . s . y e a r = 1 9 8 0 ;

( 3 9 ) 有以下程序

```
#include<stdio.h>
```

```
main()
```

```
{int a=2,b=a,c=2;
```

```
printf(    “ %d\n” ,a/b&c);
```

```
}
```

程序运行后的输出结果是

- A ) 0                      B ) 1                      C ) 2                      D ) 3

( 4 0 ) 有以下程序

```
#include<stdio.h>
```

```
main()
```

```
{FILE *fp;char str[10];
```

```
fp=open(    “ myfile.dat    ” , “ w” );
```

```
fputs(    “ abc ” ,pf);close(pf);
```

```
fp=open(    “ myfile.dat    ” , “ a+ ” );
```

```
fprintf(pf,    “ %d” ,28);
```

```
rewind(pf);
```

```
fscanf(pf,    “ %s” ,str);puts(str);
```

```
close(pf);
```

```
}
```

程序运行后的输出结果是

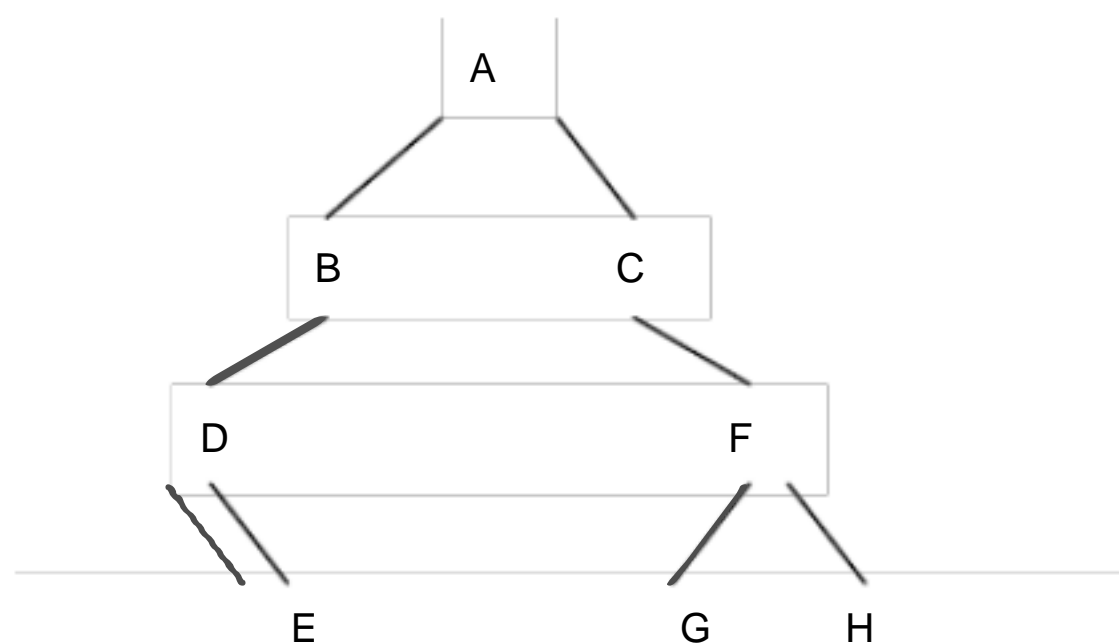
- A ) a b c                      B ) 2 8 c  
C ) a b c 2 8                      D ) 因类型不一致而出错

## 二、 填空题（ 每空 2 分， 共 3 0 分）

( 1 ) 一个队列的初始状态为空。现将元素 A , B , C , D , E , F , 5 , 4 , 3 , 2 , 1 依次入队 , 然后再依次退队 , 则元素退队的顺序为 【 1 】 。

( 2 ) 设某循环队列的容量为 50 , 如果头指针 front = 45 ( 指向队头元素的前一位置 ) , 尾指针 rear = 10 ( 指向队尾元素 ) , 则该循环队列中共有 【 2 】 个元素。

( 3 ) 设二叉树如下 :



对该二叉树进行后序遍历的结果为 【 3 】 。

( 4 ) 软件是 【 4 】 数据和文档的集合。

( 5 ) 有一个学生选课的关系，其中学生的关系模式为：学生（学号，姓名，班级，年龄），课程的关系模式为：课程（课号，课程名，学时），其中两个关系模式的键分别是学号和课号，则关系模式选课可定义为：选课（学号， 【 5 】 ，成绩）。

( 6 ) 设 x 为 int 型变量，请写出一个关系表达式 【 6 】 ，用以判断 x 同时为 3 和 7 的倍数时，关系表达式的值为真。

( 7 ) 有以下程序

```

#include<stdio.h>
main()
{int a=1,b=2,c=3,d=0;
if(a==1)
if(b!=2)
if(c==3) d=1;
else d=2;
else if(c!=3) d=3;
else d=4;
else d=5;
printf( " %d\n" ,d);
}

```

程序运行后的输出结果是 【 7 】 。

( 8 ) 有以下程序

```

#include<stdio.h>
main()
{int m,n;
scanf( " %d%d" ,&m,&n);
while(m!=n)
{while(m>n) m=m-n;
while(m<n) n=n-m;}
printf( " %d\n" ,m);
}

```

程序运行后，当输入 14 63 <回车> 时，输出结果是 【 8 】 。

( 9 ) 有以下程序

```

#include<stdio.h>

```

```

main()
{int i,j,a[[3]={1,2,3,4,5,6,7,8,9};
  for(i=1;i<3;i++)
  for(j=1;j<3;i++) printf(      “ %d” ,a[i][j]);
  printf(      “ \n ” );
}

```

程序运行后的输出结果是 【 9 】

( 1 0 ) 有以下程序

```

#include<stdio.h>
main()
{int a[]={1,2,3,4,5,6},*k[3],i=0;
  while(i<3)
  {k[i]=&a[2*i];
  printf(      “ %d” ,*k[i]);
  i++;
  }
}

```

程序运行后的输出结果是 【 10】 。

( 1 1 ) 有以下程序

```

#include<stdio.h>
main()
{int a[3][3]={1,2,3},{4,5,6},{7,8,9}};
  int b[3]={0},i;
  for(i=1;i<3;i++) b[i]=a[i][2]+a[2][i];
  for(i=1;i<3;i++)printf(      “ %d” ,b[i]);
  printf(      “ \n ” );
}

```

程序运行后的输出结果是 【 11】 。

( 1 2 ) 有以下程序

```

#include<stdio.h>
#include<string.h>
void fun(char *str)
{char temp;int n,i;
  n=strlen(str);
  temp=str[n-1];
  for(i=n-1;i>0;i--) str[i]=str[i-1];str[0]=temp;
}
main()
{char s[50];scanf(      “ %s” ,s);fun(s);printf(      “ %s\n ” ,s);}

```

程序运行后输入： abcdef <回车> , 则输出结果是 【 12】

( 1 3 ) 以下程序的功能是：将值为三位正整数的变量 x 中的数值按照个位、十位、百位的顺序拆分并输出。请填空。

```

#include<stdio.h>
main()
{int x=256;
  printf(      “ %d-%d-%d\n” , 【 13】 , x/10%10 , x/100);
}

```

```
}
```

（ 1 4 ）以下程序用以删除字符串中所有的空格，请填空。

```
#include<stdio.h>
```

```
main()
```

```
{char[100]={    “ Our teacher teach c language!    ” };int i,j;
```

```
for(i=j=0;s[i]!=    ‘ \0 ’ ;i++)
```

```
    if(s[i]!=    ‘    ’ ) {s[j]=s[i];j++;}
```

```
s[j]=    【 14 】 ;
```

```
printf(    “ %s\n ” ,s);
```

```
}
```

（ 1 5 ）以下程序的功能是：借助指针变量找出数组元素中的最大值及其元素的下标值。请填空。

```
#include<stdio.h>
```

```
main()
```

```
{int a[10],*p,*s;
```

```
for(p=a;p-a<10;p++) scanf(    “ %d” ,p);
```

```
for(p=a,s=a;p-a<10;p++) if(*p>*s) s=    【 15 】;
```

```
printf(    “ index=%d\n ” s-a);
```