

机密★启用前

2007 年 9 月全国计算机等级考试二级笔试试卷  
C 语言程序设计

24

---

注意事项

- 一、考生应严格遵守考场规则，得到监考人员指令后方可作答。
- 二、考生拿到试卷后应首先将自己的姓名、准考证号等内容涂写在答题卡的相应位置上。
- 三、选择题答案必须用铅笔填涂在答题卡的相应位置上，填空题的答案必须用蓝、黑色钢笔或圆珠笔写在答题卡的相应位置上，答案写在试卷上无效。
- 四、注意字迹清楚，保持卷面整洁。
- 五、考试结束将试卷和答题卡放在桌上，不得带走。待监考人员收毕清点后，方可离场。

---

\*\*\* 版权所有，任何单位或个人不得保留、复制和出版，违者必究 \*\*\*

教育部考试中心

二 00 七年七月制

# 2007 年 9 月全国计算机等级考试笔试试卷

## 二级公共基础知识和 C 语言程序设计

(考试时间 120 分钟, 满分 100 分)

### 一、选择题 ( (1) - (10) 每小题 2 分, (11) - (50) 每小题 1 分, 共 60 分)

- (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、数据与相关文档的完整集合
- (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) 在面向对象的方法中, 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要求的是 ( )。
- A) 程序的效率第一, 清晰第二  
B) 程序的可读性好  
C) 程序中要有必要的注释  
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据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 程序中用作用户标识符的一组标识符是 ( )。
- A) and                                      B) Date  
\_2007                                      y-m-d
- C) Hi                                      D) case  
Dr.Tom                                      Bigl
- (13) 下列选项中, 合法的一组 C 语言数值常量是 ( )。
- A) 028                                      B) 12.  
.5e-3                                      0xa23  
.0xf                                      4.5e0
- C) .177                                      D) 0x8A  
4e1.5                                      10,000  
0abc                                      3.e5
- (14) 下列叙述中正确的是 ( )。
- 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, 接着执行下列选项中的语句, 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) 下列关于 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, 下列合法的 C 语言赋值语句是 ( )。
- A) x = y = = 5;                      B) x = n%2.5;                      C) x + n = i;                      D) x = 5 = 4 + 1;
- (18) 有下列程序段:
- ```
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, 有下列程序段:
- ```
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) 有下列计算公式:





```
}  
程序的运行结果是 ( )。
```

- 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) 有下列程序:

```
#include <stdio.h>  
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));  
}
```

程序的运行结果是 ( )。

- 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) 下列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) 有下列程序:

```
#include <stdio.h>  
void fun(char *t,char *s)  
{ while(*t!=0) t++;  
  while((*t++=*s++)!=0);  
}  
main()  
{ char ss[10]="acc",aa[10]="bbxxyy";  
  fun(ss,aa); printf("%s,%s\n",ss,aa);  
}
```

程序的运行结果是 ( )。

- A) accxyy,bbxxyy                      B) acc,bbxxyy  
C) accxxyy,bbxxyy                      D) accbbxxyy,bbxxyy

(39) 有下列程序:

```
#include <stdio.h>  
#include <string.h>  
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][0]>s[j][0]){t=s[i][0];s[i][0]=s[j][0];s[j][0]=t;}
```

```

}
main( )
{ char ss[5][10]={"bcc","bbcc","xy","aaaacc","aabcc"};
  fun(ss,5); printf("%s,%s\n",ss[0],ss[4]);
}

```

程序的运行结果是 ( )。

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

(40) 在一个 C 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) 有下列程序：

```

#include <stdio.h>
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\n",k);
}

```

程序的运行结果是 ( )。

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

(42) 有下列程序：

```

#include <stdio.h>
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`，则下列叙述中正确的是 ( )。

- 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) 下列关于 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) 有下列程序:
- ```
#include <stdio.h>
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) 有下列程序:
- ```
#include <stdio.h>
#include <string.h>
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[i]=b.score[i];
  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) 设有下列定义:
- ```
union data
{int d1; float d2;}demo;
```
- 则下列叙述中错误的是 ( )。
- 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) 有下列程序:

```
#include <stdio.h>
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) 有下列程序:

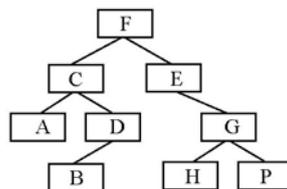
```
#include <stdio.h>
main()
{ FILE *fp; int a[10]={1,2,3,0,0};
  fp=fopen("d2.dat","wb");
  fwrite(a,sizeof(int),5,fp);
  fclose(fp);
  fp=fopen("d2.dat","rb");
  fread(a,sizeof(int),10,fp);
  fclose(fp);
  for(i=0;i<10;i++) printf("%d",a[i]);
}
```

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)

- (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, 其中最重要的是 **【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】**。



- (5) 在 E-R 图中, 矩形表示 **【5】**。
- (6) 执行下列程序时, 输入 1234567<CR>, 则输出结果是 **【6】**。

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

```

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

```

- (7) 下列程序的功能是输出 a、b、c 三个变量中的最小值。请填空。

```

#include<stdio.h>
main( )
{ int a,b,c,t1,t2;
  scanf("%d%d%d",&a,&b,&c);
  t1=a<b?【7】;
  t2=c<t1?【8】;
  printf("%d\n",t2);
}

```

- (8) 下列程序的输出结果是【9】。

```

#include <stdio.h>
main ( )
{ int n=12345,d;
  while(n!=0) { d=n%10;printf("%d",d);n/=10; }
}

```

- (9) 有下列程序段，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",s);

```

请填空，使下列程序段的功能与之完全相同

```

s=1.0;k=1;
while(【10】){s=s+1.0/(k*(k+1));【11】;}
printf("s=%f\n",s);

```

- (10) 下列程序的输出结果是【12】。

```

#include <stdio.h>
main( )
{ int i;
  for(i='a';i<'f';i++,i++) printf("%c",i-'a'+'A');
  printf("\n");
}

```

- (11) 下列程序的输出结果是【13】。

```

#include <stdio.h>
#include <string.h>
char *fun(char *t)
{ char *p=t;
  return (p+strlen(t)/2);
}
main( )
{ char *str="abcdefgh";
  str=fun(str);
  puts(str);
}

```

- (12) 下列程序中函数 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 <stdio.h>
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]<j) {j=x[m];p1=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[u]);
  f(a,10);
  for(u=0;u<10;u+ +) printf("%d",a[u]);
  printf("\n");
}
```

- (13) 下列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，其他依次类推。用#号结束输入，请填空。

```
#include <stdio.h>
#include <ctype.h>
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[i]);
}
```

- (14) 执行下列程序的输出结果是\_\_\_【18】\_\_\_。

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

- (15) 下列程序的输出结果是\_\_\_【19】\_\_\_。

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

- (16) 函数 min()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 <stdio.h>
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)m=p->data;
  return m;
}
```