

**100**  
**NCRE**

南开100题



全国计算机等级考试  
笔试考试习题集

# 二级C语言程序设计

全国计算机等级考试命题研究组 编

南开大学出版社

全国计算机等级考试

# 笔试考试习题集

二级 C 语言程序设计

(2008 版)

全国计算机等级考试命题研究组 编



南开大学出版社

天 津

**图书在版编目 (CIP) 数据**

全国计算机等级考试笔试考试习题集: 2008版. 二级  
C语言程序设计/全国计算机等级考试命题研究组编. —4  
版. —天津: 南开大学出版社, 2007. 11

ISBN 978-7-310-02269-4

I. 全… II. 全… III. ①电子计算机—水平考试—习题  
②C语言—程序设计—水平考试—习题 IV. TP3-44

中国版本图书馆 CIP 数据核字 (2007) 第 153759 号

**版权所有 侵权必究**

**南开大学出版社出版发行**

出版人: 肖占鹏

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022) 23508339 23500755

营销部传真: (022) 23508542 邮购部电话: (022) 23502200

\*

天津泰宇印务有限公司印刷

全国各地新华书店经销

\*

2007 年 11 月第 4 版 2007 年 11 月第 5 次印刷

787 × 1092 毫米 16 开本 19.375 印张 489 千字

定价: 31.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022) 23507125

## 前 言

信息时代,计算机与软件技术日新月异,在国家经济建设和社会发展的过程中,发挥着越来越重要的作用,已经成为不可或缺的关键性因素。国家教育部考试中心自1994年推出“全国计算机等级考试”以来,已经经过了十几年,考生超过千万人。

计算机等级考试需要考查学生的实际操作能力以及理论基础。因此,经全国计算机等级考试委员会专家的论证,以及教育部考试中心有关方面的研究,我们编写了《全国计算机等级考试上机考试习题集》,供考生考前学习使用。该习题集的编写、出版和发行,对考生的帮助很大,自出版以来就一直受到广大考生的欢迎。为配合社会各类人员参加考试,能顺利通过“全国计算机等级考试”,我们组织多年从事辅导计算机等级考试的专家在对近几年的考试深刻分析、研究的基础上,结合上机考试习题集的一些编写经验,并依据教育部考试中心最新考试大纲的要求,编写出这套“全国计算机等级考试笔试考试习题集”。

编写这样一套习题集,是参照上机考试习题集的做法,其内容同实际考试内容接近,使考生能够有的放矢地进行复习,希望考生能顺利通过考试。

本书针对参加全国计算机等级考试的考生,同时也可以作为普通高校、大专院校、成人高等教育以及相关培训班的练习题和考试题使用。

为了保证本书及时面市和内容准确,很多朋友做出了贡献,夏菲、李煜、孙正、宋颖、张志刚、苏鹏、刘一、李岩、毛卫东、李占元、刘时珍、敖群星等老师在编写文档、调试程序、排版、查错等工作中加班加点,付出了很多辛苦,在此一并表示感谢!

全国计算机等级考试命题研究组

2007年8月

## 目 录

|         |       |
|---------|-------|
| 第 1 套   | (1)   |
| 第 2 套   | (15)  |
| 第 3 套   | (30)  |
| 第 4 套   | (45)  |
| 第 5 套   | (61)  |
| 第 6 套   | (75)  |
| 第 7 套   | (89)  |
| 第 8 套   | (104) |
| 第 9 套   | (118) |
| 第 10 套  | (132) |
| 第 11 套  | (147) |
| 第 12 套  | (160) |
| 第 13 套  | (174) |
| 第 14 套  | (188) |
| 第 15 套  | (201) |
| 第 16 套  | (216) |
| 第 17 套  | (232) |
| 第 18 套  | (247) |
| 第 19 套  | (262) |
| 第 20 套  | (275) |
| 附录 参考答案 | (288) |

## 第1套

### 一、选择题

下列各题 A、B、C、D 四个选项中，只有一个选项是正确的，请将正确选项涂写在答题卡相应位置上，答在试卷上不得分。

- 下面关于对象概念的描述中，( ) 是错误的。
  - 对象就是 C 语言中的结构体变量
  - 对象代表着正在创建的系统中的—个实体
  - 对象是一个状态和操作（或方法）的封装体
  - 对象之间的信息传递是通过消息进行的
- 支持数据库各种操作的软件系统叫做 ( )。
  - 数据库管理系统
  - 文件系统
  - 数据库系统
  - 操作系统
- 在关系数据库模型中，通常可以把外码所在的关系称为 ( )。
  - 被参照关系
  - 参照关系
  - 主码
  - 主关系
- 下面数据结构中，属于非线性的是 ( )。
  - 线性表
  - 树
  - 队列
  - 堆栈
- 下面概念中，不属于面向对象方法的是 ( )。
  - 对象
  - 继承
  - 类
  - 过程调用
- 设有如下三个关系表

| R |  |
|---|--|
| A |  |
| m |  |
| n |  |

| S |   |
|---|---|
| B | C |
| 1 | 3 |

| T |   |   |
|---|---|---|
| A | B | C |
| m | 1 | 3 |
| n | 1 | 3 |

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. 结构化程序设计主要强调的是 ( )。

A. 程序的规模

B. 程序的效率

C. 程序设计语言的先进性

D. 程序易读性

8. 下列模式中, 能够给出数据库物理存储结构与物理存取方法的是 ( )。

A. 内模式

B. 外模式

C. 概念模式

D. 逻辑模式

9. 按照“先进先出”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

10. 从用户角度看, 下面列出的条目中 ( ) 是数据库管理系统应具有的目标。

I、用户界面友好

II、内部结构清晰、层次分明

III、开放性, 即符合标准和规范

IV、负责管理企业组织的数据库资源

A. I、II

B. I、II、III

C. III、IV

D. 都是

11. 以下叙述正确的是 ( )。

A. C 语言比其他语言高级

B. C 语言可以不用编译就能被计算机识别执行

C. C 语言以接近英语国家的自然语言和数学语言作为语言的表达形式

D. C 语言出现的最晚、具有其他语言的一切优点

12. 以下选项中可作为 C 语言合法常量的是 ( )。

A. -80.

B. -080

C. -8el.0

D. -80.0e

13. 设 d 为字符变量, 下列表达式不正确的是 ( )。

A.  $d=97$

B.  $b='a'$

C.  $d="a"$

D.  $d='g'$

14. 若有定义:  $\text{int } a=2, b=3; \text{float } x=3.5, y=2.5;$  则下面表达式的值为 ( )。

$(\text{float})(a+b)/2+(\text{int})x\%(\text{int})y$

A. 3.5

B. 35

C. 3.500000

D. 35.00000

15. C语言程序的三种基本结构是( )。

- A. 顺序结构、选择结构、循环结构
- B. 递归结构、循环结构、转移结构
- C. 嵌套结构、递归结构、顺序结构
- D. 循环结构、转移结构、顺序结构

16. 正确的说明语句为( )。

- A. `int A['a']`
- B. `int A[3,4];`
- C. `int A[];`
- D. `int *A[10];`

17. 以下( )不是C语言中用于结构化程序设计的三种基本结构之一。

- A. 顺序结构
- B. 选择结构
- C. 递归结构
- D. 循环结构

18. 有定义语句: `int b;char c[10];`, 则正确的输入语句是( )。

- A. `scanf("%d%s",&b,&c);`
- B. `scanf("%d%s",&b,c);`
- C. `scanf("%d%s",b,c);`
- D. `scanf("%d%s",b,&c);`

19. 有以下程序:

```
main()
{ unsigned a=32768; int b;
  b=a;
  printf(" b=%d\n",b);
}
```

程序执行后的输出结果是( )。

- A. `b=32768`
- B. `b=32767`
- C. `b=-32768`
- D. `b=-1`

20. 如下程序的输出结果是( )。

```
#include<stdio.h>
main()
{ float x=2.0,y;
  if(x<0.0)y=0.0
  else if(x<10.0)y=1.0/x;
  else y=1.0;
  printf(" %f\n",y);
}
```

- A. `0.000000`
- B. `0.250000`
- C. `0.500000`
- D. `1.000000`



21. 以下不正确的说法是 ( )。

- A. 实参可以是常量、变量或表达式
- C. 实参可以为任意类型

- B. 形参可以是常量、变量或表达式
- D. 形参应与其对应的实参类型一致

22. 下面程序段的运行结果是 ( )。

```
char a[7]=" abcdef" ;  
char b[4]=" ABC" ;  
strcpy(a,b);  
printf(" %c" ,a[5]);
```

- A.  (其中 表示空格)
- C. e

- B. \0
- D. f

23. 有以下程序:

```
void main()  
{ int a=5,b=4,c=3,d=2;  
  if(a>b>c)  
    printf("%d\n",d);  
  else if((c-1>=d)==1)  
    printf("%d\n",d+1);  
  else  
    printf("%d\n",d+2);  
}
```

执行后输出的结果是 ( )。

- A. 2
- C. 4

- B. 3
- D. 编译时有错, 无结果

24. 有以下程序:

```
void main()  
{ int i;  
  for(i=1; i<=40; i++)  
  { if(i++%3==0)  
    if(++i%8==0) printf("%d ",i);  
  }  
  printf("\n");  
}
```

执行后的输出结果是 ( )。

- A. 8 32
- C. 32

- B. 24
- D. 40

25. 下面程序的输出结果是 ( )。

```
void main()
{ int x,i;
  for(i=1; i<=100; i++)
  { x=i;
    if(++x%2==0)
    if(++x%3==0)
    if(++x%7==0)
    printf("%d ",x);
  }
}
```

A. 39 81

B. 42 84

C. 26 68

D. 28 70

26. 下面程序的输出结果是 ( )。

```
#include <stdio.h>
void main( )
{ int i,k,a[10],p[3];
  k=5;
  for(i=0; i<10; i++)
  a[i]=i;
  for(i=0; i<3; i++)
  p[i]=a[i*(i+1)];
  for(i=0; i<3; i++)
  k+=p[i]*2;
  printf("%d\n",k);
}
```

A. 20

B. 21

C. 22

D. 23

27. 以下程序的输出结果是 ( )。

```
void main()
{ int k=4,m=1,p;
  p=func(k,m);printf("%d",p);
  p=func(k,m);printf("%d\n",p);
}

func(int a,int b)
{ static int m=0,i=2;
  i+=m+1;m=i+a+b;
  return(m);
}
```

}

A. 8,17

B. 8,16

C. 8,20

D. 8,8

28. 在 C 语言中,形参的缺省存储类是 ( )。

A. auto

B. register

C. static

D. extern

29. 有以下定义:

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

```
char a[10],*b=a;
```

不能给数组 a 输入字符串的语句是 ( )。

A. gets(a);

B. gets(a[0]);

C. gets(&a[0]);

D. gets(b);

30. 以下叙述中错误的是 ( )。

A. 对于 double 类型数组,不可以直接用数组名对数组进行整体输入或输出

B. 数组名代表的是数组所占存储区的首地址,其值不可改变

C. 当程序执行中,数组元素的下标超出所定义的下标范围时,系统将给出“下标越界”的出错信息

D. 可以通过赋初值的方式确定数组元素的个数

31. 有以下程序:

```
void main()
```

```
{ char *s[]={ "man","woman","both"},*p;
```

```
  p=s[1];
```

```
  printf("%s,%c\n",*(p+1),s[0]);
```

```
}
```

执行后输出结果是 ( )。

A. a,woman

B. w,man

C. o,man

D. 程序报错

32. 有以下程序:

```
main()
```

```
{ int k[8]={8,7,6,5,0,0},*s;
```

```
  s=k+1;
```

```
  printf("%d\n",s[2]+1);
```

```
}
```

执行后输出结果是 ( )。

A. 随机值

B. 0

C. 5

D. 6

33. 有以下程序:

```
void main()
{
    int n;
    for(n=3; n<=10; n++)
    { if(n%6==0) break; printf("%d",n); }
}
```

程序运行后的输出结果是 ( )。

A. 345

B. 456

C. 345

D. 456

34. 下面程序的输出结果是 ( )。

```
#include <stdio.h>
void fun(int *s)
{ static int j=0;
  do
  { s[j]+=s[j+1];
    }while(++j<2);
}
void main()
{ int k,a[10]={1,2,3,4,5};
  for(k=1; k<3; k++)
  fun(a);
  for(k=0; k<5; k++)
  printf("%d",a[k]);
}
```

A. 35756

B. 23445

C. 35745

D. 12345

35. 以下数组定义中不正确的是 ( )。

A. int a[2][3];

B. int b[][3]={0,1,2};

C. int c[100][100]={0};

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

36. 以下程序的输出结果是 ( )。

```
void main()
{
    int a[4][4]={{1,3,5},{2,4,6},{3,5,7}};
    printf("%d%d%d%d\n",a[0][3],a[1][2],a[2][1],a[3][0]);
}
```

}

A. 0650

B. 1470

C. 5430

D. 输出值不定

37. 阅读以下函数:

```
fun(char *s1,char *s2)
{ int i=0;
  while(s1[i]==s2[i]&& s2[i]!='\0') i++;
  return(s1[i]!='\0'&& s2[i]!='\0');
}
```

此函数的功能是 ( )。

A. 将 s2 所指字符串赋给 s1

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

C. 比较 s1 和 s2 所指字符串是否相等, 若相等, 函数值为 1, 否则函数值为 0

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

38. 有以下程序:

```
void main()
{ int k=5,n=0;
  while(k>0)
  { switch(k)
    { default : break;
      case 1 : n+=k;
      case 2 :
      case 3 : n+=k;
    }
    k--;
  }
  printf("%d\n",n);
}
```

程序运行后的输出结果是 ( )。

A. 0

B. 4

C. 6

D. 7

39. 下面程序:

```
void main()
{
  int a=-1,b=4,k;
  k=(a++<=0)&&(!(b--<=0));
  printf("%d%d%d\n",k,a,b);
}
```

}

输出结果是 ( )。

A. 003

B. 012

C. 103

D. 112

40. 设有如下说明:

```
typedef struct ST
```

```
{ long a; int b; char c[2]; } NEW;
```

则下面叙述中正确的是 ( )。

A. 以上的说明形式非法

B. ST 是一个结构体类型

C. NEW 是一个结构体类型

D. NEW 是一个结构体变量

41. 有以下程序:

```
void main()
```

```
{
```

```
    int a=1,b;
```

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

```
        { if(a>=8) break;
```

```
          if(a%3!=1){a+=5; continue;}
```

```
          a+=3;
```

```
        }
```

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

```
}
```

程序运行后的输出结果是 ( )。

A. 3

B. 4

C. 5

D. 6

42. 若有以下定义:

```
char s[20]="programming",*ps=s;
```

则不能代表字符 o 的表达式是 ( )。

A. ps+2

B. s[2]

C. ps[2]

D. ps+=2,\*ps

43. 下面程序的输出结果是 ( )。

```
void main()
```

```
{ int a[]={1,2,3,4,5,6};
```

```
int *p;
```

```
p=a;
```

```
printf("%d ",*p);
```

```
printf("%d ",*(++p));
```

```

printf("%d ",*++p);
printf("%d ",*(p--));
p+=3;
printf("%d %d ",*p,*(a+3));
}

```

A. 1 2 3 3 5 4

B. 1 2 3 4 5 6

C. 1 1 2 2 3 4 5

D. 1 1 2 3 4 4 5

44. 下面程序的输出结果是 ( )。

```

#include <stdio.h>
void main()
{ int n[3][3], i, j;
  for(i=0; i<3; i++)
  for(j=0; j<3; j++)
  n[i][j]=i+j;
  for(i=0; i<2; i++)
  for(j=0; j<2; j++)
  n[i+1][j+1]+=n[i][j];
  printf("%d\n", n[i][j]);
}

```

A. 14

B. 0

C. 6

D. 不确定

45. 以下程序中，能够通过调用函数 fun，使 main 函数中的指针变量 p 指向一个合法的整型单元的是 ( )。

A. void main()

```

{ int *p;
  fun(p);
  ...
}

```

int fun(int \*p)

```

{ int s; p=&s;}

```

B. void main()

```

{ int *p;
  fun(&p);
  ...
}

```

int fun(int \*\*p)

```

{ int s; *p=&s;}

```

C. #include<stdlib.h>

```

void main()
{   int *p;
    fun(&p);
    ...
}
int fun(int **p)
{   *p=(int*)malloc(2);}
D. #include<stdlib.h>
void main()
{   int *p;
    fun(p);
    ...
}
int fun(int *p)
{   p=(int*)malloc(sizeof(int));}

```

46. 当 p 指向 s 的首地址时，以下 ( ) 表示和 s[i] 不相等。

- |            |            |
|------------|------------|
| A. * (s+i) | B. * (p+i) |
| C. p[i]    | D. *p+i    |

47. 有以下程序:

```

#include<string.h>
struct STU
{char name[10];
 int num;};
void f(char *name, int num)
{struct STU s[2]={{"Zhongling",124},{"Niuniu",125}};
 num=s[0].num;
 strcpy(name,s[0].name);
}
void main()
{struct STU s[2]={{"Lining",121},{"Luguang",122}},*p;
 p=&s[1]; f(p->name,p->num);
 printf("%s %d\n",p->name,p->num);
}

```

程序运行后的输出结果是 ( )。

- |                  |                |
|------------------|----------------|
| A. Zhongling 122 | B. Niuniu 125  |
| C. Lining 121    | D. Luguang 122 |

48. 下列程序的输出结果是 ( )。



```
#include<stdio.h>
void main()
{ union
{
    int k;
    char i[2];
} *s,a;
s=&a;
s->i[0]=0x39;s->i[1]=0x38;
printf("%x\n",s->k);
}
```

- A. cccc3839  
C. 380039

- B. 3938  
D. 390038

49. 有以下程序

```
#include <stdlib.h>
main()
{ char *p,*q;
  p=(char*)malloc(sizeof(char)*20); q=p;
  scanf("%s %s",p,q); printf("%s %s\n",p,q);
}
```

若从键盘输入: abc def<回车>, 则输出结果是 ( )。

- A. def def  
C. abc d

- B. abc def  
D. d d

50. 下面程序的输出结果是 ( )。

```
#include <stdio.h>
void main()
{ union
{ int i[2];long k;char c[4];}r,*s=&r;
s->i[0]=0x39;
s->i[1]=0x38;
printf("%c\n",s->c[0]);
}
```

- A. 39  
C. 38

- B. 9  
D. 8