

科举网★内部

## 全国计算机等级考试二级 C 语言 2009 年 9 月 笔试试卷及答案

### 一、选择题答案

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

做题很吃力吗？基础没打牢！[www.kejuwang.com](http://www.kejuwang.com) 边学边做，轻松考二级。

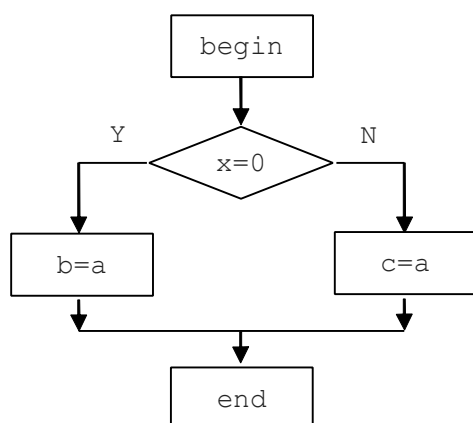
### 二、选择题 ((1)~(10)、(21)~(40) 每小题 2 分，(11)~(20) 每小题 1 分，共 70 分)

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

- 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
- 下列数据结构中，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
- 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. 面向对象方法的需求分析工具
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. 椭圆形
- B. 矩形
- C. 菱形
- D. 三角形

10. 设有三个关系 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	1

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的数据结构
- 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 <stdio.h>
main()
{
    int case;
    float printf;
    printf("请输入 2 个数:");
    scanf("%d %f", &case, &printf);
}
```

```
printf("%d %f\n", case, printf);
}
```

该程序在编译时产生错误，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 <stdio.h>
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 <stdio.h>
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\n",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. if((a>b)&&(b>c)) k=1;  
else k=0;
- B. if((a>b)|| (b>c)) k=1;  
else k=0;
- C. if(a<=b) k=0;  
else if(b<=c) k=1;
- D. if(a>b) k=1;  
else if(b>c) k=1;  
else k=0;

20. 有以下程序

```
#include <stdio.h>
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 <stdio.h>
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. char a='a';
- B. char a='\n';
- C. char a='aa';
- D. char a='\x2d';

23. 有以下程序

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

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 <stdio.h>
void fun(int p)
{
    int d=2;
    p=d++;
    printf("%d",p);
}
main()
{
```

```
int a=1;
fun(a);
printf("%d\n",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 -2147463647
int findmax(int x[],int n)
{
    int i,max;
    for(i=0;i<n;i++)
    {
        max=MIN;
        if(max<x[i])max=x[i];
    }
    return max;
}
```

造成错误的原因是 \_\_\_\_\_。

- A. 定义语句 int i,max 中 max 未赋值
- B. 赋值语句 max=MIN; 中，不应该给 max 赋 MIN 值
- C. 语句 if (max<x[i])max=x[i]; 中判断条件设置错误
- D. 赋值语句 max=MIN; 放错了位置

26. 有以下程序

```
#include <stdio.h>
main()
{
    int m=1,n=2,*p=&m,*q=&n,*r;
    r=p;p=q;q=r;
    printf("%d,%d,%d,%d\n",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 <stdio.h>
#include <string.h>
main()
{
    char str[][20]={"One*World","One*Dream!"},*p=str[1];
    printf("%d,",strlen(p));
    printf("%s\n",p);char s[]={"012xy"};
}
```

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 <stdio.h>
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("\n");
}
```

程序运行后的输出结果是 \_\_\_\_\_ 。

- 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 <stdio.h>
#include <string.h>
main()
{
    char a[10]="abcd";
    printf("%d,%d\n",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 <stdio.h>
int f(int n);
main()
{
    int a=3,s;
    s=f(a);
    s=s+f(a);
    printf("%d\n",s);
}
int f(int n)
{
    static int a=1;
    n+=a++;
    return n;
}
```

程序运行后的输出结果是 \_\_\_\_\_ 。

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

35. 有以下程序

```
#include <stdio.h>
#define f(x) x*x*x
main()
{
    int a=3,s,t;
    s=f(a+1);t=f((a+1));
    printf("%d,%d\n",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;} n;
- 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 <stdio.h>
#include <string.h>
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\n",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. 1002,ZhangDa,1202.0
- C. 1001,ChangRong,1098.0
- D. 1002,ChangRong,1202.0

39. 有以下程序

```
int r=8;
printf("%d\n",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. 文件由一系列数据一次排列组成，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，只能是文本文件

### 三、填空题答案

1. 14
2. 逻辑条件
3. 需求分析
4. 多对多
5. 身份证号
6. 5
7. 1
8. 1 0
9. 5
10. 0 1 123
11. 3025
12. &p.ID
13. How are you? How
14. 16
15. 1 3 6

做题很吃力吗？基础没打牢！[www.kejuwang.com](http://www.kejuwang.com) 边学边做，轻松考二级。

### 四、填空题（每空 2 分，共 30 分）

请将每空的正确答案写在答题卡【1】至【15】序号的横线上，答在试卷上不得分。

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 的值是 **【7】** 。

8. 以下程序运行后的输出结果是 **【8】** 。

```
#include <stdio.h>
main()
{
    int x = 20;
    printf("%d ", 0<x<20);
    printf("%d\n", 0<x && x<20);
}
```

9. 以下程序运行后的输出结果是 **【9】** 。

```
#include <stdio.h>
main()
{
    int a=1, b=7;
    do{
        b = b/2;
        a += b;
    } while (b>1);
    printf("%d\n", a);
}
```

10. 有以下程序

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

```

11. 有以下程序

```

#include <stdio.h>
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\n",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 <stdio.h>
main()
{
    char a[20]="How are you?",b[20];
    scanf("%s",b);
    printf("%s %s\n",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 <stdio.h>

```

```
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\n",a.num);
}
```

程序运行后输出的结果是     【14】     。

15. 有以下程序

```
#include <stdio.h>
fun(int x)
{
    if(x/2>0)
        fun(x/2);
    printf("%d ",x);
}
main()
{
    fun(6);
    printf("\n");
}
```

程序运行后输出的结果是     【15】     。

## 五、选择题答案

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