

2010 年 3 月全国计算机等级考试二级笔试试卷

C 语言程序设计

(考试时间 90 分钟, 满分 100 分)

一、选择题 (1) ~ (10)、(21) ~ (40) 每题 2 分, (11) ~ (20) 每题 1 分, 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, 请将正确选项填涂在答题卡相应位置上, 答在试卷上不得分。

- (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 $(\log_2 n)$
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 $(n \log_2 n)$
- (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

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

T

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
- (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) AaBc
 C) a_b
 D) a--b
- (14) 若有定义: double a=22; 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;
- (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, <CR>代表回车, 注意: 回车也是一个字符)
- ```
12<CR>
34<CR>
```
- 则输出结果是 ()。
 A) 1234
 B) 12
 C) 12
 D) 12
 3
 34
- (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) 有如下嵌套的 if 语句:
- ```
if(a<b)
 if(a<c)k=a;
 else k=c;
else
 if(b<c)k=b;
 else k=c;
```
- 以下选项中与上述 if 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) 9,18  
B) 8,11  
C) 7,11  
D) 10,14

(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) 12  
B) 11  
C) 10  
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];

(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=0;i<5;i++) s=s+a[b[i]];
```

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

```
}
```

程序运行后的输出结果是 ( )。

A) 6

B) 10

C) 11

D) 15

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

```
for(j=i;j<=i;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="\n123\\";
```

```
printf("%d %d\n",strlen(s1),strlen(s2));
```

则输出的结果是 ( )。

A) 5 5

B) 10 5

C) 10 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=0;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) 10

C) 14

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) -12

C) -20

D) 10

(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);

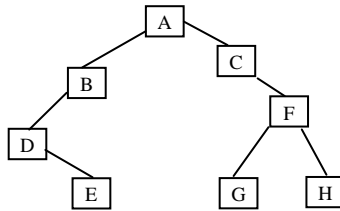
C) data2. real=data1. real;

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 A t);
main()
{ struct A a={1001,"ZhangDa",1098.0};
```





(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=0;i<3;i++)
 for(j=i;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=0;i<3;i++) b[i]=a[i][2]+a[2][i];
 for(i=0;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);}

```

程序运行后的输入：abcdef<回车>，则输出结果是 **【12】**。

(13) 以下程序的功能：将值为 3 位正整数的变量 x 中的数值按个位、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);
}

```

(14) 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 s[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);
}

```

(15) 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);
}

```



## C 语言程序设计参考答案及详解

## 一、选择题

(1) A) 【解析】本题主要考查的知识点为查找技术。顺序查找的使用情况：①线性表为无序表；②表采用链式存储结构。二分法查找只适用于顺序存储的有序表，并不适用于线性链表。

(2) D) 【解析】算法的时间复杂度，是指执行算法所需要的计算工作量。算法的工作量可以用算法在执行过程中所需基本运算的执行次数来度量。

(3) B) 【解析】软件根据应用目标的不同，是多种多样的。软件按功能可以分为：应用软件、系统软件、支撑软件（或工具软件）。应用软件是为解决特定领域的应用而开发的软件。系统软件是计算机管理自身资源，提高计算机使用效率并为计算机用户提供各种服务的软件。支撑软件是介于系统软件和应用软件之间，协助用户开发软件的工具性软件，包括辅助和支持开发和维护应用软件的工具软件，还包括辅助管理人员控制开发进程和项目管理的工具软件。

(4) A) 【解析】在对程序进行了成功的测试之后将进入程序调试（通常称 Debug，即排错）。程序调试的任务是诊断和改正程序中的错误。它与软件测试不同，软件测试是尽可能多地发现软件中的错误。先要发现软件的错误，然后借助于一定的调试工具去执行并找出错误的具体位置。软件测试贯穿整个软件生命期，调试主要在开发阶段。

(5) C) 【解析】本题考查数据流程图 DFD 的概念。对于面向数据流的结构化分析方法，按照 DeMarco 的定义，“结构化分析就是使用数据流图（DFD）、数据字典（DD）、结构化英语、判定表和判定树等工具，来建立一种新的、称为结构化规格说明的目标文档”。结构化分析方法的实质是着眼于数据流，自顶向下，逐层分解，建立系统的处理流程，以数据流图和数据字典为主要工具，建立系统的逻辑模型。数据流图（DFD——Data Flow Diagram）是描述数据处理过程的工具，是需求理解的逻辑模型的图形表示，它直接支持系统的功能建模。数据流图从数据传递和加工的角度，来刻画数据流从输入到输出的移动变换过程。由此可得数据流程图是结构化方法的需求分析工具。

(6) B) 【解析】本题考查软件生命周期的相关概念。通常，将软件产品从提出、实现、使用维护到停止使用退役的过程称为软件生命周期。也就是说，软件产品从考虑其概念开始，到该软件产品不能使用为止的整个时期都属于软件生命周期。一般包括可行性研究与需求分析、设计、实现、测试、交付使用以及维护等活动。还可以将软件生命周期分为软件定义、软件开发及软件运行维护三个阶段。

(7) A) 【解析】数据库管理系统一般提供相应的数据语言，它们分别是：数据定义语言，负责数据的模式定义与数据的物理存取构建；数据操纵语言，负责数据的操纵，包括查询及增、删、改等操作；数据控制语言，负责数据完整性、安全性的定义与检查以及并发控制、故障恢复等功能。

(8) D) 【解析】本题考查关系数据库中，数据单位的相关概念。文件是指存储在外部介质上的数据的集合。数据库是存储在计算机存储设备上，结构化的相关数据集合。它不仅包括描述事物的数据本身，而且还包括相关事物之间的联系。在数据库中，表的“行”称为“记录”，“列”称为“字段”。由此可得，题中存取一个学生信息的数据单位为记录。

(9) C) 【解析】E-R 模型（实体联系模型）是将现实世界的要求转化成实体、联系、属性等几个基本概念，以及它们间的两种基本联接关系，并且可用一种图非常直观地表示出来。它属于数据库设计的概念设计阶段。

(10) A) 【解析】由关系 R 和 T 所包含的元素可知，关系 R 经过选择操作就可以得到关系 T。

(11) B) 【解析】在 C 语言中，允许函数的嵌套调用，即递归调用。在 C 语言中不存在过程的概念，所以选项 A) 是错误的，在 C 语言中函数允许单独编译，可以作为单独的文件形式存在，因而选项 C) 和 D) 是错误的。

(12) A) 【解析】C 语言中的注释不可以夹在变量名或关键字的中间；C 语言的变量必须先定义后使用；一个浮点数可以和一个整数相加，运算符两侧的运算类型也可以不一致；数值常量中不允许夹带空格。

(13) D) 【解析】在 C 语言中的用户标识符只能由字母，数字和下划母组成，并且第一个字符必须是字母或下划线，在选项 D) 中出现了非法的字符--。

(14) C) 【解析】在 C 语言中，求余运算符的运算对象只能是整型，在题目中，变量 a 是一个双精型变量，因此选项 C) 中赋值表达式的右侧的表达式是一个非法的表达式。

(15) C) 【解析】在程序中，scanf 函数把 1 和 2 分别赋给变量 a 和 b，getchar() 函数把回车符赋给变量 c，getchar() 函数把 3 赋给变量 d。

(16) D) 【解析】在 C 语言中没有定义逻辑类型，而是用 0 代表假，用非零代表真。

(17) A) 【解析】在逻辑表达式中， $a \parallel b$ ，只有在 a 跟 b 都为 0 时，表达式的结果才为 0。在题中所给的表达式中 a 要么等于 1，要么不等于 1，肯定会有一方的值不为真。

(18) B) 【解析】if else 语句的含义是，如果变量 a 的值等于 1，则把变量 b 的值赋给变量 a，否则，变量 a 的值加 1。在本题的 4 个选项中，与其含义不同的是选项 B)，表示如果变量 a 的值等于 1，则把变量 a 的值加 1，否则，把变量 b 的值赋给变量 a。

(19) C) 【解析】嵌套的 if else 语句的含义是，在变量 a 的值小于变量 b 的值，同时，变量 a 的值小于变量 c 的值时，把变量 a 的值赋给变量 k。在变量 a 的值小于变量 b 的值，同时，变量 a 的值大于等于变量 c 的值时，把变量 c 的值赋给变量 k。在变量 a 的值大于等于变量 b 的值，同时，变量 b 的值小于变量 c 的值时，把变量 b 的值赋给变量 k。在变量 a 的值大于等于变量 b 的值，同时，变量 b 的值大于等于变量 c 的值时，把变量 c 的值赋给变量 k。

(20) A) 【解析】这道题考查了 for 循环的嵌套应用。对于外层 for 循环来说，变量 i 从 1 增到 2，循环执行 2 次。当变量 i 的值等于 1 时，对于内层 for 循环来说，变量 j 从 3 减到 1，求得变量 m 的值等于 6。当变量 i 的值等于 2 时，第一遍执行内层 for 循环，变量 j 的值等于 3，那么  $i*j$  的值等于 6，其值大于 3，这时执行 break 语句，退出内层 for 循环的执行。所以程序输出变量 m 的值为 6。

(21) D) 【解析】在这段程序中变量 a 的初值等于 1，变量 b 的初值等于 2。第 1 次执行 for 循环语句后，变量 a 的值等于 3，变量 b 的值等于 3。第 2 次执行 for 循环后，变量 a 等于 6，变量 b 的值等于 7。第 3 次执行 for 循环后，变量 a 等于 9，变量 b 的值等于 14。第 4 次执行 for 循环后，变量 a 等于 10，这时退出 for 循环，此时变量 b 的值等于 14。因而程序输出变量 a,b 的值分别为 10,14。

(22) D) 【解析】变量 k 的值为八制数 011，其对应的十制进数为 9。

(23) A) 【解析】S 是一个字符指针变量，让它指向一个字符串常量。

(24) B) 【解析】return 语句主要用于返回函数的值。在一个自定义函数中，可以根据不同的情况设置多条 return 语句返回函数的值。

(25) D) 【解析】在定义一维数组时，数组的下标应该是一个确定的整数值。要注意的是在定义二维数组时，其第一维下标可以省略，但第二维下标不能省略。

(26) A) 【解析】在这道题目中要 fun 函数的参数传递方式，形参 c 是传地址，形参 d 是传值。所以在主函数中，执行 fun 时，把变量 b 的地址传给了 c，把 a 的值传递给了 d，经过运算，输出的值为 b,B。在主函数的输出语句中，输出变量 b 的值为 b，变量 a 的值为 A。

(27) D) 【解析】首先，pt 跟 \* 结合，所以是一个指针，所指向的对象是含有 3 个元素的数组。所定义的指针没加 const，所以是个指针变量，还可以指向类似对象的其他元素，也就是说还可以指向其他含有 3 个元素的数组。这也是选项 D) 所表达的意思。可以指向每行 3 个整数元素的二维数组。

如果是 \*pt[3]，由于 [] 的优先级比 \* 高，所以首先是数组形式 pt[3]，然后再与 \* 结合，这样指针数组包含 3 个指针，pt[0]、pt[1]、pt[2]。int (\*p)[3]；是数组指针，p 是个指针，指向一个长度为 3 的 int 型数组。

(28) B) 【解析】语句 \*s=a，使用指针变量 s 指向一维数组 a 的首地址，所以 \*(s+3) 表示的是引用数组 a[3]。

(29) C) 【解析】这道题目是计算机一维数组 a 中某些元素的和。这些元素的下标值是一维数组 b 中元素值。 $s=0+a[0]+a[2]+a[1]+a[3]+a[0]=0+1+3+2+4+1=11$ 。

(30) C) 【解析】这道题主要考查的是二维数组的有关知识。在两层 for 循环语句中，每次执行变量 t 的累加时，二维数组元素 b 的第二下标的值与第一下标相同的值，所以程序执行的过程为  $t=1+b[1][1]+b[2][2]=1+1+2=4$ 。

(31) A) 【解析】本题要注意的是转义字符的问题，在字符串 s2 中，有两个转义字符，分别是回车换行符和反斜线。所以其字符串长度等于 5。

(32) C) 【解析】函数 fun(int \*x,int i) 的功能是把数组元素 x[i] 的值赋给 x[0]。所以在主函数中执行 fun(a,2) 后，数组元素 a[0] 的值等于 3。然后在 for 循环语句中，输出数组 a 中的前 4 个元素的值，分别是 3,2,3,4。

(33) B) 【解析】本题主要考查的是函数的递归调用，子函数 f 是一个递归函数，所以主函数中执行 f(a,4) 时，其执行过程是把数组 a 中的所有元素按逆顺相加。最后变量 s 的值等于 10。

(34) C) 【解析】在子函数 fun() 中定义局部静态变量 x，其初值等于 1，然后再将其值扩大 1 倍后返回。在主函数中执行 for 循环语句 2 次，调用函数 fun() 两次，所以变量 s 的值等于 4。这里要强调的一点是当函数 fun() 执行完毕后，局部静态变量 x 的值不会丢失，仍然有效。

(35) C) 【解析】本题主要考查的是宏替换，在程序中执行  $d=\text{sub}(a+b)*c$  这条语句时，首先要进行宏替换，依据宏 sub(a) 的定义可知，上面这条语句替换为  $d=(a+b)-(a+b)*c$ ，因为乘法的优先级比减法的优先级高，所以先进行乘法运算，即  $d=5-5*5=-20$ 。

(36) B) 【解析】本题主要考查结构体变量的定义与初始化。题目中定义了 complex 类型的结构体变量 data1 和 data2，并对 data1 进行了初始化。选项 A)、C)、D) 都是正确的，可以用一个结构体变量对另一个结构体变量赋值，用一个结构体变量的成员对另一个结构体变量的成员赋值。

(37) A) 【解析】本题主要考查的是函数调用时参数之间的传递问题。在 C 语言中参数之间的传递是传值，也就是把实参的值复制一份传递给形参，而实参的值不发生变化。所以对于本题来说，在主函数中执行 f(a)，把结构体变量 a 的值复制一份传递给形参变量 t，而实参变量 a 的值保持不变。

(38) D) 【解析】本题主要考查的是结构体变量的嵌套定义，也就是说一个结构体变量也可以是一个结构体。在本题中 works 这个结构体中包含了结构体 s。所以对结构体变量 w 中 year 成员的引用是 w.s.year。

(39) A) 【解析】因为算术运算符除法的优先级要高于按位与运算的优先级，所以表达式  $a/b\&c$  的计算过程是，首先进行  $a/b$  的运算，结果为 1，然后 1 再与 2 进行按位与运算，其结果为 0。

(40) C) 【解析】在程序中首先把字符串 abc 写入到文件 myfile.dat 中，然后再把整数 28 追加到字符串 abc 的后面，最后把文件 myfile.dat 的内容读入到字符串 str 中，并把其内容输出。

## 二、填空题

(1) 【1】 ABCDEF54321

【解析】在队列这种数据结构中，最先插入的元素将最先删除，队列又称为“先进先出”的线性表。在本题，根据队列先进先出的特点，元素退队的顺序与进队的顺序相同，即 A, B, C, D, E, F, 5, 4, 3, 2, 1。

(2) 【2】 15

【解析】本题考查队列个数的计算方法。因为循环队列个数=循环队列的容量-头指针+尾指针，即  $50-45+10=15$ 。

(3) 【3】 EDBGHFCA

【解析】对于二叉树的后序遍历，其规则是，首先遍历左子树，然后遍历右子树，最后访问根结点，并且遍历左、右子树时，仍然先遍历左子树，然后遍历右子树，最后访问根结点。依据此规则，对于题目给出的二叉树，后序遍历的结果为 EDBGHFCA。

(4) 【4】 程序

【解析】本题考查软件的概念。计算机软件是计算机系统中与硬件相互依存的另一部分，是包括程序、数据及相关文档的完整集合。其中，程序是软件开发人员根据用户需求开发的、用程序设计语言描述的、适合计算机执行的指令（语句）序列。数据是使程序能正常操纵信息的数据结构。文档是与程序开发、维护和使用有关的图文资料。可见软件由两部分组成：一是机器可执行的程序和数据；二是机器不可执行的，与软件开发、运行、维护、使用等有关的文档。

(5) 【5】 课号

【解析】本题考查关系模式的相关内容。因为两个关系模式的键分别是学号和课号。

(6) 【6】  $(x \% 3 == 0) \&\& (x \% 7 == 0)$

【解析】题目要求写出一个关系表达式，当整型变量  $x$  同时为 3 和 7 的倍数时，其值为真。那么首先这个关系表达式为逻辑与，其次要满足整型变量  $x$  同时能被 3 和 7 整除，也就是  $x$  对 3 和 7 求模时，模数为 0。

(7) 【7】 4

【解析】在程序中因为变量  $a$  的值等于 1，变量  $b$  的值等于 2，变量  $c$  的值等于 3。所以在执行 `if else` 嵌套语句时，变量  $d$  的值等于 4。

(8) 【8】 7

【解析】因为变量  $m$  的初值等于 14，变量  $n$  的初值等于 63，所以执行 `while(m < n) n = n - m;` 这条语句 4 次后，变量  $m$  的值等于 14，变量  $n$  的值等于 7，此时  $m$  的值大于  $n$  的值，执行 `while(m > n) m = m - n;` 这条语句，此时变量  $m$  的值等于 7。

(9) 【9】 123569

【解析】在程序中，首先定义了一个三行三列的二维数组  $a$ ，并赋予初始值 {1, 2, 3, 4, 5, 6, 7, 8, 9}。因为二维数组是以行来存储数组元素的，所以第一行的数组元素为 {1, 2, 3}，第二行的数组元素为 {4, 5, 6}，第三行的数组元素为 {7, 8, 9}。然后再用两层 `for` 循环语句，输出对角线及上三角上的元素，所以其输出结果为：123569。

(10) 【10】 135

【解析】在程序中定义了一维数组  $a$  和指针数组  $k$ ，在 `while` 循环语句中，当  $i$  等于 0 时， $k[0]$  的等于  $a[0]$  的地址，所以输入  $a[0]$  的值为 1。当  $i$  等于 1 时， $k[1]$  的等于  $a[2]$  的地址，所以输入  $a[2]$  的值为 3。当  $i$  等于 2 时， $k[2]$  的等于  $a[4]$  的地址，所以输入  $a[4]$  的值为 5。所以此空应该填写 135。

(11) 【11】 101418

【解析】根据程序中的定义，

$b[0] = a[0][2] + a[2][0] = 3 + 7 = 10$ ;  $b[1] = a[1][2] + a[2][1] = 6 + 8 = 14$ ;  $b[2] = a[2][2] + a[2][2] = 9 + 9 = 18$ 。所以输出结果为 101418。

(12) 【12】 abcde

【解析】函数 `fun(char *str)` 的作用是把  $str$  所指的字符串的尾字符移动到首字符的位置，其余字符向后移动一位。所以在主函数中，当把字符串 `abcdef` 赋给字符串  $s$ ，在调用函数 `fun(s)` 后，字符串  $s$  的值变为 `abcde`。所以输出结果为 `abcde`。

(13) 【13】  $x \% 10$

【解析】本题要求取一个三位整数的个位上的数字，只要用这个三位数对 10 求模即可。

(14) 【14】 `'\0'`

【解析】C 语言中字符串结束符是 `'\0'`，本题中相当于字符串  $s$  中存储着含有空格的字符，当发现空格的时候就用这个空格字符后面的字符前移覆盖这个空格字符，依次类推向后继续。可想而知，最后  $s$  字符串中不是空格的字符全部向前填充了原来的字符串的位置，得到的最终字符串肯定比之前的字符串短，如果结尾不加 `'\0'` 的话，肯定输出有问题的字符串。

(15) 【15】  $p$

【解析】指针变量  $s$  用来记录数组中最大元素的位置，当指针  $p$  所指的数组元素值大于  $s$  所指的元素的值时，把变量  $p$  的值赋给变量  $s$ ，所以此空应该填写  $p$ 。