



www.cip.com.cn
 版权所有 侵权必究

全国计算机等级考试 历年试卷汇编及详解

2005年9月~2009年3月全部真卷+详尽解析

二级



○ 模拟软件安装序列号: 2310-5797-6686-0190



销售分类建议: 计算机/等级考试



定价: 19.00元

773-44/88D
 *2009(3)
 2009

出版

全国计算机等级考试历年试卷汇编及详解 二级 C

最新大纲
 2009年9月考试专用



附1CD-ROM

全国计算机等级考试
 新思路系列辅导用书

全国计算机等级考试 历年试卷汇编及详解

2005年9月~2009年3月全部真卷+详尽解析

二级



- 超级模拟软件: 模拟真实的上机考试环境, 自动组卷、自动评分, 给考生提供一个难得的上机练习机会
- 每题均有解析: 试题配有答案和详细的解析

全国计算机等级考试命题研究组
 新思路教育科技研究中心

编



化学工业出版社

全国计算机等级考试

历年试卷汇编及详解

2005年9月~2009年3月全部真题+详尽解析

二级 C

(2009年9月考试专用)

全国计算机等级考试命题研究组 编
新思路教育科技研究中心



化学工业出版社

· 北京 ·

内 容 简 介

2009年全国计算机等级考试在新大纲标准下实施,为了向考生提供专业、准确的辅导资料,本书编写组设计、开发了本套历年试卷合集。

本套试卷不仅仅是整理、编排自2005年9月考试以来所有的历年真题,且对真题进行严谨地校验,使试卷内容规范、正确。更重要的是,每道试题一一予以解题分析,尤其对于起点低、基础薄弱的考生而言,这将是备考的最佳学习方案。

图书在版编目(CIP)数据

全国计算机等级考试历年试卷汇编及详解. 二级 C / 全国计算机等级考试命题研究组, 新思路教育科技研究中心编. —北京: 化学工业出版社, 2009. 5
(全国计算机等级考试新思路系列辅导用书)

ISBN 978-7-122-05236-0

ISBN 978-7-89472-016-0 (光盘)

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

中国版本图书馆CIP数据核字(2009)第050776号

策划编辑: 张立 陈静

装帧设计: 尹琳琳

责任编辑: 陈静 张敏

出版发行: 化学工业出版社(北京市东城区青年湖南街13号 邮政编码100011)

印 装: 三河市延风印装厂

787mm×1092mm 1/8 印张9 1/4 字数314千字 2009年5月北京第1版第1次印刷

购书咨询: 010-64518888 (传真: 010-64519686) 售后服务: 010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书, 如有缺损质量问题, 本社销售中心负责调换。

定 价: 19.00元(含1CD-ROM)

版权所有 违者必究

目录 Chapter

备考捷径——练真题

每年都有很多考生问向同一个问题,那就是:全国计算机等级考试过关的捷径在哪里?根据我们多年来对考试的研究和教学成果来看,任何考试都有规律可循,一旦了解和顺应这种规律去学习,将会取得事半功倍的效果。其实,很多善于总结的考生早已经发现,无论任何考试,“学真题、练真题”是一种非常省时、高效的学习方法。

纵观等考十余年,尽管大纲数次变动,考试题型、题型屡有修改,但核心的内容没有变,命题风格、出题形式、考核重点几乎是一脉相承,未曾变动。尤其是近两年的真题极具参考和学习价值,考生有时很难相信:有的题目是今年考了一次,同样的题目、同样的位置,明年会再次出现。如果考生有心去研究一下最近几年的真题,就会发现很多似曾相识的“老面孔”。

既然对于计算机等级考试而言,真题如此重要,该如何利用真题搞好学习呢?

1. 选对真题

很多考生为图省事、省钱,往往从网上下载历年题目。其实网上提供下载的题目大多是没有经过严谨的校对和出版的,有的甚至连题目都不是完整的。这样的题目只能起负面的作用。

2. 认真对待每一题

先做题,遇到自己完全不解的题目或是模糊的题目,翻阅答案的解析,弄清考核知识点。因为凡是历年真题中出现的知识点,一般就是下一次考试的考点,只不过有的题目变换参数,其考核的知识点还是一致的。

3. 养成记录+分析的习惯

题目做错了或是不会做,证明考生对此考核知识点学习得不够,建议考生在试题旁做一标记,集中一起去翻阅答案解析或其他辅导资料,弄清考点。

4. 真题+模拟题双练

做完真题,最好再分阶段地做几套质量好的模拟试题。同样是试题,仿真度高的模拟试题不仅能使考生对知识点的掌握情况进行一番测试,同时还具有一定的预测能力,说不定考试的时候真的会遇到几道自己做过的题目呢。

最后我们还要强调的是:任何试卷都不是万能的药,它只是我们提供给考生的一项工具、一个好助手,考试的关键还在于考生掌握的知识技能及心态。

预祝广大考生通过自己的智慧和努力,早日考试成功!

2005年9月全国计算机等级考试笔试试卷·····	(共6页)
2006年4月全国计算机等级考试笔试试卷·····	(共6页)
2006年9月全国计算机等级考试笔试试卷·····	(共6页)
2007年4月全国计算机等级考试笔试试卷·····	(共6页)
2007年9月全国计算机等级考试笔试试卷·····	(共6页)
2008年4月全国计算机等级考试笔试试卷·····	(共6页)
2008年9月全国计算机等级考试笔试试卷·····	(共6页)
2009年3月全国计算机等级考试笔试试卷·····	(共6页)
2005年9月笔试试卷参考答案及详解·····	(共4页)
2006年4月笔试试卷参考答案及详解·····	(共4页)
2006年9月笔试试卷参考答案及详解·····	(共4页)
2007年4月笔试试卷参考答案及详解·····	(共4页)
2007年9月笔试试卷参考答案及详解·····	(共4页)
2008年4月笔试试卷参考答案及详解·····	(共3页)
2008年9月笔试试卷参考答案及详解·····	(共3页)
2009年3月笔试试卷参考答案及详解·····	(共2页)

2005 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)

R		
A	B	C
1	1	2
2	2	3

S		
A	B	C
3	1	3

T		
A	B	C
1	1	2
2	2	3
3	1	3

一、选择题 [(1) ~ (10) 每小题 2 分, (11) ~ (50) 每小题 1 分, 共 60 分]

(1) 下列叙述中正确的是 ()。

- A) 程序设计就是编制程序
- B) 程序的调试必须由程序员自己去完成
- C) 程序经调试改错后还应进行再测试
- D) 程序经调试改错后不必进行再测试

(2) 下列数据结构中, 能用二分法进行查找的是 ()。

- A) 顺序存储的有序线性表
- B) 线性链表
- C) 二叉链表
- D) 有序线性链表

(3) 下列关于栈的描述正确的是 ()。

- A) 在栈中只能插入元素而不能删除元素
- B) 在栈中只能删除元素而不能插入元素
- C) 栈是特殊的线性表, 只能在一端插入或删除元素
- D) 栈是特殊的线性表, 只能在一端插入元素, 而在另一端删除元素

(4) 下列叙述中正确的是 ()。

- A) 一个逻辑数据结构只能有一种存储结构
- B) 数据的逻辑结构属于线性结构, 存储结构属于非线性结构
- C) 一个逻辑数据结构可以有多种存储结构, 且各种存储结构不影响数据处理的效率
- D) 一个逻辑数据结构可以有多种存储结构, 且各种存储结构影响数据处理的效率

(5) 下列描述中正确的是 ()。

- A) 软件工程只是解决软件项目的管理问题
- B) 软件工程主要解决软件产品的生产率问题
- C) 软件工程的主要思想是强调在软件开发过程中需要应用工程化原则
- D) 软件工程只是解决软件开发中的技术问题

(6) 在软件设计中, 不属于过程设计工具的是 ()。

- A) PDL (过程设计语言)
- B) PAD 图
- C) N-S 图
- D) DFD 图

(7) 下列叙述中正确的是 ()。

- A) 软件交付使用后还需要再进行维护
- B) 软件工具交付使用就不需要再进行维护
- C) 软件交付使用后其生命周期就结束
- D) 软件维护是指修复程序中被破坏的指令

(8) 数据库设计的根本目标是要解决 ()。

- A) 数据共享问题
- B) 数据安全问题
- C) 大量数据存储空间问题
- D) 简化数据维护

(9) 设有如下关系表:

则下列操作中正确的是 ()。

A) $T \cap S$ B) $T \cup S$

C) $T \times S$

D) T / S

(10) 数据库系统的核心是 ()。

- A) 数据模型
- B) 数据库管理系统
- C) 数据库
- D) 数据库管理员

(11) 下列叙述中错误的是 ()。

- A) 用户所定义的标识符允许使用关键字
- B) 用户所定义的标识符应尽量做到“见名知意”
- C) 用户所定义的标识符必须以字母或下划线开头
- D) 用户定义的标识符中, 大、小写字母代表不同标识

(12) 下列叙述中错误的是 ()。

- A) C 语句必须以分号结束
- B) 复合语句在语法上被看作一条语句
- C) 空语句出现在任何位置都不会影响程序运行
- D) 赋值表达式末尾加分号就构成赋值语句

(13) 下列叙述中正确的是 ()。

- A) 调用 printf() 函数时, 必须要有输出项
- B) 使用 putchar() 函数时, 必须在之前包含头文件 stdio.h
- C) 在 C 语言中, 整数可以以十二进制、八进制或十六进制的形式输出
- D) 调节 getch() 函数读入字符时, 可以从键盘上输入字符所对应的 ASCII 码

(14) 下列关于函数的叙述中正确的是 ()。

- A) 每个函数都可以被其他函数调用 (包括 main 函数)
- B) 每个函数都可以被单独编译
- C) 每个函数都可以单独运行
- D) 在一个函数内部可以定义另一个函数

(15) 若有语句: `char *line[5];` 下列叙述中正确的是 ()。

- A) 定义 line 是一个数组, 每个数组元素是一个基类型为 char 的指针变量
- B) 定义 line 是一个指针变量, 该变量可以指向一个长度为 5 的字符型数组
- C) 定义 line 是一个指针数组, 语句中的 * 号称为间址运算符
- D) 定义 line 是一个指向字符型函数的指针

(16) 有下程序段:

```
typedef struct NODE
{ int num; struct NODE *next;
} OLD;
```

下列叙述中正确的是 ()。

- A) 以上的说明形式非法
- B) NODE 是一个结构体类型
- C) OLD 是一个结构体类型
- D) OLD 是一个结构体变量

(17) 下列叙述中错误的是 ()。

- A) 在 C 语言中, 对二进制文件的访问速度比文本文件快

- B) 在 C 语言中, 随机文件以二进制代码形式存储数据
C) 语句 FILE fp; 定义了一个名为 fp 的文件指针
D) C 语言中的文本文件以 ASCII 码形式存储数据

(18) 当把四个表达式用作 if 语句的控制表达式时, 有一个选项与其他三个选项含义不同, 这个选项是 ()。

- A) k%2
C) (k%2)!=0
(19) 下列不能正确计算代数式 $\frac{1}{3} \sin^2(\frac{1}{2})$ 值的 C 语言表达式是 ()。
A) 1/3 *sin(1/2)*sin(1/2)
C) pow(sin(0.5),2)/3

- B) k%2==1
D) !k%2==1
B) sin(0.5)*sin(0.5)/3
D) 1/3.0*pow(sin(1.0/2),2)

(20) 下列能正确定义且赋初值的语句是 ()。

- A) int n1=n2=10;
C) float f=f+1.1;
(21) 下列程序的功能是: 给 r 输入数据后计算半径为 r 的圆面积 s。程序在编译时出错。
main()
/* Beginning */
{ int r; float s;
scanf("%d",&r);
s=*a*r*r;
printf("s=%f\n",s);
}

- 出错的原因是 ()。
A) 注释语句书写位置错误
B) 存放圆半径的变量 r 不应该定义为整型
C) 输出语句中格式描述符非法
D) 计算圆面积的赋值语句中使用了非法变量

(22) 设有定义: int k=1,m=2; float f=7;则下列选项中错误的表达式是 ()。
A) k=k>=k
C) k%int(f)
B) -k++
D) k>=f>=m

(23) 设有定义: int a=2,b=3,c=4;则下列选项中值为 0 的表达式是 ()。
A) (! a==1)&&(b==0)
C) a && b
B) (a<b)&&! c || 1
D) a || (b+a)&&(c-a)

(24) 有下列程序段:
int k=0,a=1,b=2,c=3;
k=a<b?b:a; k=k>c?c:k;
执行该程序段后, k 的值是 ()。
A) 3
B) 2
C) 1
D) 0

(25) 设变量 a、b、c、d 和 y 都已正确定义并赋值。若有下列 if 语句
if(a<b)
if(c==d) y=0;
else y=1;
该语句所表示的含义是 ()。

- A) $y = \begin{cases} 0 & a < b \text{ 且 } c = d \\ 1 & a \geq b \end{cases}$
B) $y = \begin{cases} 0 & a < b \text{ 且 } c = d \\ 1 & a \geq b \text{ 且 } c \neq d \end{cases}$
C) $y = \begin{cases} 0 & a < b \text{ 且 } c = d \\ 1 & a < b \text{ 且 } c \neq d \end{cases}$
D) $y = \begin{cases} 0 & a < b \text{ 且 } c = d \\ 1 & c \neq d \end{cases}$

(26) 有下列程序段

```
int n,t=1,s=0;
scanf("%d",&n);
do { s=s+t;t=t-2; } while(t!=n);
```

为使此程序段不陷入死循环, 从键盘输入的数据应该是 ()。
A) 任意正奇数
C) 任意正偶数
B) 任意负偶数
D) 任意负奇数

(27) 设变量 i 已正确定义, 则以下能正确计算 f=n! 的程序段是 ()。
A) f=0;
C) f=1;
for(i=1;i<=n;i++) f*=i;
for(i=n;i>1;i--) f*=i;

B) f=1;
D) f=i;
for(i=1;i<=n;i++) f*=i;
for(i=n;i>=2;i--) f*=i;

(28) 设有定义: int n1=0,n2,*p=&n2,*q=&n1; 以下赋值语句中与 n2=n1; 语句等价的是 ()。
A) *p=*q;
C) *p=&n1;
B) p=q;
D) p=*q;

(29) 若有定义: int x=0,*p=&x;则语句 printf("%d\n",*p); 的输出结果是 ()。
A) 随机值
C) x 的地址
B) 0
D) p 的地址

(30) 设 fun() 函数的定义形式为 ()。
void fun(char ch,float x){...}
则下列对函数 fun 的调用语句中, 正确的是
A) fun("abc",3.0);
C) fun('65',2.8);
B) t=fun('D',16.5);
D) fun(32,32);

(31) 有下列程序:
main()
{ int a[10]={1,2,3,4,5,6,7,8,9,10}; *p=&a[3]; *q=p+2;
printf("%d\n",*p+*q);
}
程序运行后的输出结果是 ()。
A) 16
B) 10
C) 8
D) 6

(32) 有下列程序:
main()
{ char p[]={'a','b','c'},q[]="abc";
printf("%d%d\n",sizeof(p),sizeof(q));
};
程序运行后的输出结果是 ()。
A) 4 4
B) 3 3
C) 3 4
D) 4 3

(33) 有下列程序:
#define f(x)=(x*x)

```
main()
{ int i,i2;
  i1=f(8)/f(4); i2=f(4+4)/f(2+2);
  printf("%d, %d\n",i1,i2);
}
```

程序运行后的输出结果是 ()。

- A) 64,28 B) 4,4 C) 4,3 D) 64,64

(34) 有下列程序:

```
main()
{ char a1='M',a2='m';
  printf("%c\n",a1,a2);
}
```

下列叙述中正确的是 ()。

- A) 程序输出大写字母 M B) 程序输出小写字母 m
C) 格式说明符不足, 编译出错 D) 程序运行时产生出错信息

(35) 有下列程序:

```
#include <stdio.h>
main()
{ char c1='1',c2='2';
  c1=getchar(); c2=getchar(); putchar(c1); putchar(c2);
}
```

当运行时输入:a<CR>后, 下列叙述正确的是 ()。

- A) 变量 c1 被赋予字符 a,c2 被赋予回车符
B) 程序将等待用户输入第 2 个字符
C) 变量 c1 被赋予字符 a,c2 中仍是原有字符 2
D) 变量 c1 被赋予字符 a,c2 中将无确定值

(36) 有下列程序:

```
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

(37) 有下列程序:

```
main()
{ int a[ ]={2,4,6,8,10},y=0,x,*p;
```

```
p=&a[1];
for(x=1;x<3;x++) y+=p[x];
printf("%d\n",y);
}
```

程序运行后的输出结果是 ()。

- A) 10 B) 11 C) 14 D) 15

(38) 有下列程序:

```
void sort(int a[ ],int n)
{ int i,j,t;
  for(i=0;i<n-1;i++)
  for(j=i+1;j<n;j++)
    if(a[i]<a[j]){ t=a[i]; a[i]=a[j]; a[j]=t;
  }
```

```
main()
{ int aa[10]={1,2,3,4,5,6,7,8,9,10};
  sort(aa+2,5);
  for(i=0;i<10;i++)printf("%d,",aa[i]);
  printf("\n");
}
```

程序运行后的输出结果是 ()。

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

(39) 有下列程序:

```
void sum(int a[ ])
{ a[0]=a[-1]+a[1];
}
main()
{ int a[10]={1,2,3,4,5,6,7,8,9,10};
  sum(&a[2]);
  printf("%d\n",a[2]);
}
```

程序运行后的输出结果是 ()。

- A) 6 B) 7 C) 5 D) 9

(40) 有下列程序:

```
void swap1 (int c0[ ],int c1[ ])
{ int t;
  t=c0[0]; c0[0]=c1[0]; c1[0]=t;
}
void swap2(int *c0,int *c1)
{ int t;
  t=*c0; *c0=*c1; *c1=t;
}
main()
{ int a[2]={3,5}, b[2]={3,5};
  swap1(a,a+1); swap2(&b[0],&b[1]);
```

```
printf("%d %d %d %d\n",a[0],a[1],b[0],b[1]);
}
```

程序运行后的输出结果是 ()。

A) 3 5 5 3 B) 5 3 3 5 C) 3 5 5 5 D) 5 3 5 3

(41) 有下列程序:

```
#include <string.h>
main()
{ char p[] = {'a', 'b', 'c'}, q[10] = {'a', 'b', 'c'};
  printf("%d%d\n", strlen(p), strlen(q));
}
```

- 下列叙述中正确的是 ()。
- A) 在给 p 和 q 数组赋初值时, 系统会自动添加字符串结束符, 故输出的长度都为 3
 B) 由于 p 数组中没有字符串结束符, 长度不能确定, 但 q 数组中字符串长度为 3
 C) 由于 q 数组中没有字符串结束符, 长度不能确定, 但 p 数组中字符串长度为 3
 D) 由于 p 和 q 数组中都没有字符串结束符, 故长度都不能确定

(42) 有下列程序, 其中函数 f() 的功能是将多个字符串按字典顺序排序 ()。

```
#include <string.h>
void f(char *p[], int n)
{ char *t; int i, j;
  for(i=0; i<n-1; i++)
    for(j=i+1; j<n; j++)
      if(strcmp(p[i], p[j])>0){ t=p[i]; p[i]=p[j]; p[j]=t; }
}
main()
{ char *p[5] = {"abc", "aabdfig", "abbd", "dcdbce", "cd"};
  f(p, 5);
  printf("%d\n", strlen(p[1]));
}
```

- 程序运行后的输出结果是 ()。
- A) 2 B) 3 C) 6 D) 4
- (43) 有下列程序:

```
#include <string.h>
void f(char *s, char *t)
{ char k;
  k=*s; *s=*t; *t=k;
  s++; t--;
  if(*s) f(s, t);
}
main()
{ char str[10] = "abcdefg", *p;
  p=str+strlen(str)/2+1;
  f(p, p-2);
  printf("%s\n", str);
}
```

程序运行后的输出结果是 ()。

- A) abcdefg B) gfedcba C) gbcdefa D) abedcfig

(44) 有下列程序:

```
float f1(float n)
{ return n*n; }
float f2(float n)
{ return 2*n; }
main()
{ float (*p1)(float), (*p2)(float), (*t)(float), y1, y2;
  p1=f1; p2=f2;
  y1=p2(p1(2.0));
  t=p1; p1=p2; p2=t;
  y2=p2(p1(2.0));
  printf("%3.0f, %3.0f\n", y1, y2);
}
```

程序运行后的输出结果是 ()。

- A) 8, 16 B) 8, 8 C) 16, 16 D) 4, 8

(45) 有下列程序:

```
int a=2;
int f(int n)
{ static int a=3;
  int t=0;
  if(n%2){ static int a=4; t+=a++; }
  else{ static int a=5; t+=a++; }
  return t+a++;
}
main()
{ int s=a, i;
  for(i=0; i<3; i++) s+=f(i);
  printf("%d\n", s);
}
```

程序运行后的输出结果是 ()。

- A) 26 B) 28 C) 29 D) 24

(46) 有下列程序:

```
#include <string.h>
struct STU
{ int num;
  float TotalScore;
};
void f(struct STU p)
{ struct STU s[2] = {{20044, 550}, {20045, 537}};
  p.num=s[1].num; p.TotalScore=s[1].TotalScore;
}
main()
```

```

{ struct STU s[2]={{"20041,703"}, {"20042,580"};
f(s[0]);
printf("%d %3.0f\n",s[0].num,s[0].TotalScore);
}

```

程序运行后的输出结果是 ()。

A) 20045 537 B) 20044 550 C) 20042 580 D) 20041 703

(47) 有下列程序:

```

#include <string.h>
struct STU
{ char name[10];
int num;
};
void f(char *name,int num)
{ struct STU s[2]={{"SunDan",20044}, {"Penghua",20045}};
num=s[0].num;
strcpy(name,s[0].name);
}

```

main()

```

{ struct STU s[2]={{"YangSan",2004}, {"LiSiGuo",20042}}; *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) SunDan 20042 B) SunDan 20044
C) LiSiGuo 20042 D) YangSan 20041

(48) 有下列程序:

```

struct STU
{ char name[10]; int num; float TotalScore; };
void f(struct STU *p)
{ struct STU s[2]={{"SunDan",20044,550}, {"Penghua",20045,537}};
*q=s; +p; +q; *p=*q;
}
main()
{ struct STU s[3]={{"YangSan",20041,703}, {"LiSiGuo",20042,580}};
f(s);
printf("%s %d %3.0f\n",s[1].name,s[1].num,s[1].TotalScore);
}

```

程序运行后的输出结果是 ()。

A) SunDan 20044 550 B) Penghua 20045 537
C) LiSiGuo 20042 580 D) SunDan 20041 703

(49) 下列程序的功能是进行位运算:

```

main()
{ unsigned char a,b;
a=7^3; b=~4 & 3;
printf("%d %d\n",a,b);
}

```

程序运行后的输出结果是 ()。

A) 4 3 B) 7 3 C) 7 0 D) 4 0

(50) 有下列程序:

```

#include <stdio.h>
main()
{ FILE *fp; int i,k,n;
fp=fopen("data.dat","w+");
for(i=1;i<6;i++)
{ fprintf(fp,"%d ",i);
if(i%3==0) fprintf(fp,"\n");
}
rewind(fp);
fscanf(fp,"%d%d",&k,&n); printf("%d%d\n",k,n);
fclose(fp);
}

```

程序运行后的输出结果是 ()。

A) 0 0 B) 123 45 C) 1 4 D) 1 2

二、填空题(每空 2 分,共 40 分)

- (1) 数据管理技术发展过程经过人工管理、文件系统和数据库系统 3 个阶段,其中数据独立性最高的阶段是 【1】。
- (2) 算法复杂度主要包括时间复杂度和 【2】 复杂度。
- (3) 在进行模块测试时,要为每个被测试的模块另外设计两类模块:驱动模块和承接模块(桩模块)。其中 【3】 的作用是将测试数据传送给被测试的模块,并显示被测试模块所产生的结果。
- (4) 一棵二叉树第六层(根结点为第一层)的结点数最多为 【4】 个。
- (5) 数据结构分为逻辑结构和存储结构,循环队列属于 【5】 结构。
- (6) 下列程序运行后的输出结果是 【6】。

```

main()
{ int x=0210; printf("%X\n",x);
}

```

(7) 下列程序运行后的输出结果 【7】。

```

main()
{ int a=1,b=2,c=3;
if(c=a) printf("%d\n",c);
else printf("%d\n",b);
}

```

- (8) 已有定义: double *p; 请写出完整的语句,利用 malloc 函数使 p 指向一个双精度型的动态存储单元 【8】。
- (9) 下列程序运行后的输出结果是 【9】。

```

main()
{ char c; int n=100;
float f=10; double x;
x=f* n/=(c=50);
printf("%d %f\n",n,x);
}

```

- (10) 下列程序的功能是计算: $s=1+12+123+1234+12345$ 。请填空。

```
main()
{ int t=0,s=0,i;
  for(i=1;i<=5;i++)
  { t+=【10】; s=s*t; }
  printf("s=%d\n",s);
}
```

- (11) 已知字母 A 的 ASCII 码为 65, 下列程序运行后的输出结果是【11】。

```
main()
{ char a,b;
  a='A'+5-'3'; b='A'+6-'2';
  printf("%d %c\n",a,b);
}
```

- (12) 有下列程序:

```
int sub(int n) { return(n/10+n%10); }
main()
{ int x,y;
  scanf("%d",&x);
  y=sub(sub(x));
  printf("%d\n",y);
}
```

若运行时输入:1234<CR>, 程序的输出结果是【12】。

- (13) 下列函数 strstr 的功能是实现字符串的连接,即将 t 所指字符串复制到 s 所指字符串的尾部。例如: s 所指字符串为 abcd, t 所指字符串为 efgh, 函数调用后 s 所指字符串为 abcdefgh。请填写。

```
#include <string.h>
void strstr(char *s,char *t)
{ int n;
  n=strlen(s);
  while(*(s+n)=【13】){s++;t++;}
}
```

- (14) 下列程序运行后的输出结果是【14】。

```
#include <string.h>
char *ss(char *s)
{ char *p,t;
  p=s+1;t=*s;
  while(*p){*(p-1)=*p;p++;}
  *(p-1)=t;
  return s;
}
main()
{ char *p,str[10]="abcdefg";
  p=ss(str);
  printf("%s\n",p);
}
```

- (15) 下列程序运行后的输出结果是【15】。

```
int f(int a[],int n)
{ if(n>=1)return f(a,n-1)+a[n-1];
  else return 0;
}
main()
{ int aa[5]={1,2,3,4,5},s;
  s=f(aa,5); printf("%d\n",s);
}
```

- (16) 下列程序运行后的输出结果是【16】。

```
struct NODE
{ int num; struct NODE *next;
};
main()
{ struct NODE s[3]={ {1,'0'}, {2,'0'}, {3,'0'} },*p,*q,*r;
  int sum=0;
  s[0].next=s+1; s[1].next=s+2; s[2].next=s;
  p=s; q=p->next; r=q->next;
  sum+=q->num; sum+=r->num; sum+=r->next->num;
  printf("%d\n",sum);
}
```

- (17) 下列程序的功能是输出如下形式的方阵:

13	14	15	16
9	10	11	12
5	6	7	8
1	2	3	4

请填写。

```
main()
{ int i,j,x;
  for(j=4;j>【17】;j--)
  { for(i=1;i<=4;i++)
    { x=(j-1)*4+【18】;
      printf("%4d",x);
    }
    printf("\n");
  }
}
```

- (18) 下列函数 rotate 的功能是: 将 a 所指 N 行 N 列的二维数组中的最后一行放到 b 所指二维数组的第 0 列中, 把 a 所指二维数组中的第 0 行放到 b 所指二维数组的最后一列中, b 所指二维数组中其他数据不变。

```
#define N 4
void rotate(int a[][N],int b[][N])
{ int i,j;
  for(i=0;i<N;i++)
  { b[i][N-1]=【19】; 【20】=a[N-1][i];
  }
}
```

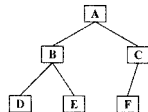
2006 年 4 月全国计算机等级考试笔试试卷

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) 下列选项中不属于结构化程序设计的方法是 ()。
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) DBEAFD
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

C) DBS 包含 DB 和 DBMS

B) DBMS 包含 DB 和 DBS

D) 没有任何关系

- (11) 下列不合法的用户标识符是 ()。

A) j2_KEY

B) Double

C) 4d

D) _8

- (12) 下列不合法的数值常量是 ()。

A) 011

B) 1e1

C) 8.0E0.5

D) 0xabed

- (13) 下列不合法的字符常量是 ()。

A) '018'

B) '\n'

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) 下列能正确定义字符串的语句是 ()。

A) char str[]={'064'};

B) char str="kx43";

C) char str="";

D) char str[]="0";

- (16) 下列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) int 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) getch()

B) gets()

C) getchar()

D) scanf()

- (18) 下列 4 个程序中, 完全正确的是 ()。

A) #include <stdio.h>

B) #include <stdio.h>

main();

main()

/*programming*/

/*programming*/

printf("programming! \n");

printf("programming! \n");

C) #include <stdio.h>

D) include <stdio.h>

main()

main()

/*programming*/

/*programming*/

printf("programming! \n");

printf("programming! \n");

- (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)

B) switch((int)x);

{ case 1.0:printf("x \n");

{case 1:printf("x \n");

case 2.0:printf("x \n");

case 2:printf("x \n");

C) switch(a+b)

D) switch(a-b)

{ case 1:printf("x \n");

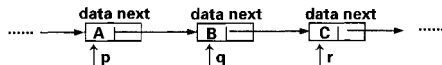
{ case 1:printf("x \n");

case 2+1:printf("x \n");

case c:printf("x \n");

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

- A) `main()`
`{.....x = fun(2,10);.....}`
`float fun(int a,int b){.....}`
- C) `float fun(int,int);`
`main()`
`{.....x = fun(2,10);.....}`
`float fun(int a,int b){.....}`
- (21) 在下列给出的表达式中,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,下列正确的程序段是 ()。
- A) `while( (ch=getchar())!= 'N') printf("%c",ch);`
B) `while(ch=getchar())!= 'N') printf("%c",ch);`
C) `while(ch=getchar())!= 'N') printf("%c",ch);`
D) `while((ch=getchar())!= 'N') printf("%c",ch);`
- (23) 已定义下列函数:
- ```
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的 3 个结点。



- ```
struct node
{ 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,下列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) 有下列程序段

- ```
struct st
{ int x;int *y;} *pt;
int a[] = {1,2},b[] = {3,4};
struct st c[2] = {10,a,20,b};
pt=c;
```
- 下列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 `fgetc(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) 有下列程序:
- ```
#include <stdio.h>
main()
{ char c1,c2,c3,c4,c5,c6;
 scanf("%c%c%c%c%c",&c1,&c2,&c3,&c4);
 c5=getchar(); c6=getchar();
 putchar(c1);putchar(c2);
 printf("%c%c\n",c5,c6);
}
```
- 程序运行后,若从键盘输入(从第 1 列开始)
- ```
123<CR>
45678<CR>
```
- 则输出结果是 ()。
- A) 1267 B) 1256
C) 1278 D) 1245
- (30) 有下列程序:
- ```
main()
{ int y=10;
 while(y--);printf("y=%d\n",y);
}
```
- 程序执行后的输出结果是 ( )。
- A) `y=0` B) `y=-1`  
C) `y=1` D) `while` 构成无限循环
- (31) 有下列程序:
- ```
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\n",a,b,c,d);
}

```

程序输出 ()。

- A) 0, 1, 2, 0
C) 1, 1, 2, 0

- B) 0,0, 0, 3
D) 编译有错

(32) 有下列程序:

```

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\n",x);
}

```

程序执行后的输出结果是 ()。

- A) x=4
C) x=6

- B) x=8
D) x=12

(33) 有下列程序:

```

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
C) 5.0

- B) 5
D) 0.0

(34) 有下列程序:

```

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

```

程序执行后的输出结果是 ()。

- A) 7 5 3
C) 3 6 9

- B) 3 5 7
D) 7 5 1

(35) 有下列程序:

```

fun(char p[ ][10])
{ int n=0;
  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\n",fun(str));
}

```

程序执行后的输出结果是 ()。

- A) 1
C) 3

- B) 2
D) 0

(36) 有下列程序:

```

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\n",s);
}

```

程序执行后的输出结果是 ()。

- A) 45
C) 25

- B) 20
D) 36

(37) 有下列程序:

```

void fun1(char *p)
{char *q;
  q=p;
  while(*q!='\0')
  { (*q)+++;q++;}
}
main()
{ char a[ ]={'Program'}, *p;
  p=&a[3];fun1(p);printf("%s\n",a);
}

```

程序执行后的输出结果是 ()。

- A) Prohsbn
C) Progsbn

- B) Prphsbn
D) Program

(38) 有下列程序:

```

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\n",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) 有下列程序:

```
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);
}
```

执行程序时, 给变量 x 输入 10, 程序的输出结果是 ()。

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

(40) 有下列程序:

```
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) 有下列程序:

```
main()
{ union
  { char ch[2];
    int d;
  };
  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) 有下列程序:

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

程序执行后的输出结果是 ()。

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

(43) 执行下列程序后, test.txt 文件的内容是 (若文件能正常打开) ()。

```
#include <stdio.h>
main()
{ FILE *fp;
  char *s1="Fortran",*s2="Basic";
  if((fp=fopen("test.txt","wb"))==NULL)
  { printf("Can't open test.txt file\n"); 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) 下列叙述中错误的是 ()。

- A) C 语言源程序经编译后生成后缀为 .obj 的目标程序
B) C 程序经过编译、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 语言中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机器指令

(45) 下列叙述中错误的是 ()。

- 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) 下列叙述中错误的是 ()。

- 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给对应的实参

(47) 设有下列定义和语句

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

```
p=str;
```

则下列叙述中正确的是 ()。

- A) *p 与 str[0] 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) 下列叙述中错误的是 ()。

- 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) 下列叙述中正确的是 ()。

- 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 程序必须在开头包含预处理命令行, #include<stdio.h>
- D) C 语言的预处理不能实现宏定义和条件编译的功能

(50) 下列叙述中错误的是 ()。

- 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, 把数据库看成是二维表, 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) 下列程序用于判断 a、b、c 能否构成三角形, 若能, 输出 YES, 否则输出 NO。当给 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同时满足三个条件: a+b>c, a+c>b, b+c>a。请填写。

```
main()
```

```
{ float a,b,c;
```

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

```
if(【7】)printf("YES\n"); /*a、b、c 能构成三角形*/
```

```
else printf("NO\n"); /*a、b、c 不能构成三角形*/
```

```
}
```

(8) 下列程序的输出结果是 【8】。

```
main()
```

```
{ int a[3]={1,2,9},{3,4,8},{5,6,7}},i,s=0;
```

```
for(i=0;i<3;i++) s+=a[i][i]+a[3-i-1];
```

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

```
}
```

(9) 当运行下列程序时, 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("%s\n",str);
```

```
}
```

```
main()
```

```
{ char str[40];
```

```
scanf("%s",str); insert(str);
```

```
}
```

(10) 下列程序的运行结果是: 【10】。

```
fun(int f[], int n)
```

```
{ int i,m;
```

```
if(n==1)return f[0];
```

```
else
```

```
if(n>=2){m=fun(f,n-1); return m;
```

```
}
```

```
main()
```

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

```
printf("%d\n",fun(a,10));
```

```
}
```

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

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

```
#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("%c",s[n--]);
}

```

当编译连接通过后，运行程序 T18 时，输入 Thank! 则输出结果是： 11。

- (12) 下列函数 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。请填写空。

```

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) 下列程序的功能是：输出 100 以内(不含 100)能被 3 整除且个位数为 6 的所有整数，请填写空。

```

main()
{ int i,j;
  for(i=0; 13 ;i++)
  { j=i*10+6;
    if( 14 )continue;
    printf("%d\n",j);
  }
}

```

- (14) 下列 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。请填写空。

```

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

```

- (15) 下列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

```

请填写空完成该程序。

```

main()
{ int i,j,n; char ch='A';
  scanf("%d",&n);

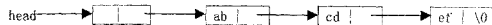
```

```

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!\n");
printf("\n");
}

```

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



```

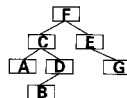
#include <stdio.h>
typedef struct node /*链表结点结构*/
{ char sub[3];
  struct node *next;
}Node;
Node fun(char s) /*建立链表*/
{ ..... }
void disp(Node *h)
{ Node *p;
  p=h->next;
  while( 19 )
  { printf("%s\n",p->sub); p= 20 ; }
}
main()
{ Node *hd;
  hd=fun(); disp(hd); printf("\n");
}

```

2006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) 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4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3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) $T \leftarrow R \cap S$
 - B) $T \leftarrow R \cup S$
 - C) $T \leftarrow R \times S$
 - D) $T \leftarrow R / S$
- (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) 在长度为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) ACBDFGE
 - B) ACBDFGE
 - C) ADBCGEF
 - D) FCADBEG
- (11) 下列叙述中错误的是 ()。
 - A) 一个C语言程序只能实现一种算法
 - B) C程序可以由多个程序文件组成
 - C) C程序可以由一个或多个函数组成
 - D) 一个C函数可以单独作为一个C程序文件存在
- (12) 下列叙述中正确的是 ()。
 - A) 每个C程序文件中都必须要有个main()函数
 - B) 在C程序中main()函数的位置是固定的
 - C) C程序中所有函数之间都可以相互调用, 与函数所处位置无关
 - D) 在C程序的函数中不能定义另一个函数
- (13) 下列定义变量的语句中错误的是 ()。
 - A) `int _int;`
 - B) `double int;`
 - C) `char For;`
 - D) `float USS;`
- (14) 若变量x、y已正确定义并赋值, 下列符合C语言语法的表达式是 ()。
 - A) $+x, y = x -$
 - B) $x + 1 = y$
 - C) $x = x + 10 = x + y$
 - D) `double(x)/10`
- (15) 下列关于逻辑运算符两侧运算对象的叙述中正确的是 ()。
 - A) 只能是整数0或1
 - B) 只能是整数0或非0整数
 - C) 可以是结构体类型的数据
 - D) 可以是任意合法的表达式
- (16) 若有定义 `int x,y;` 并已正确给变量赋值, 则下列选项中与表达式 $(x-y)?(x+y):(y+x)$ 的条件表达式 $(x-y)$ 等价的是 ()。
 - A) $(x-y > 0)$
 - B) $(x-y < 0)$
 - C) $(x-y < 0 \parallel x-y > 0)$
 - D) $(x-y = 0)$
- (17) 有下列程序:

```

main()
{
    int x,y,z;
    x=y=1;
    z=x++ + y + ++ + y;
    printf("%d,%d,%d\n",x,y,z);
}
    
```

程序运行后的输出结果是 ()。

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

(18) 设有定义: `int a; float b;` 执行 `scanf("%2d%f",&a,&b);` 语句时, 若从键盘输入 `876543.0<CR>`,
a 和 b 的值分别是 ()。

- A) 876 和 543.000000
C) 87 和 543.000000

- B) 87 和 6.000000
D) 76 和 543.000000

(19) 有下列程序:

```
main()
{
    int a=0,b=0;
    a=10; /*给 a 赋值
    b=20; 给 b 赋值 */
    printf("a+b=%d\n",a+b); /*输出计算结果 */
}
```

程序运行后的输出结果是 ()。

- A) a+b=10 B) a+b=30
C) 30 D) 出错

(20) 在嵌套使用 if 语句时, C 语言规定 else 总是 ()。

- A) 和之前与其具有相同缩进位置的 if 配对 B) 和之前与其最近的 if 配对
C) 和之前与其最近的且不带 else 的 if 配对 D) 和之前的第一个 if 配对

(21) 下列叙述中正确的是 ()。

- A) break 语句只能用于 switch 语句
B) 在 switch 语句中必须使用 default
C) break 语句必须与 switch 语句中的 case 配对使用
D) 在 switch 语句中, 不一定使用 break 语句

(22) 有下列程序:

```
main()
{
    int k=5;
    while(--k) printf("%d",k--3);
    printf("\n");
}
```

执行后的输出结果是 ()。

- A) 1 B) 2
C) 4 D) 死循环

(23) 有下列程序:

```
main()
{
    int i;
    for(i=1;i<=40;i++)
    {
        if(i++%5==0)
            if(++i%8==0) printf("%d",i);
    }
    printf("\n");
}
```

执行后的输出结果是 ()。

- A) 5 B) 24

C) 32

D) 40

(24) 下列选项中, 值为 1 的表达式是 ()。

- A) 1-'0'
C) '1'-0

- B) 1-'0'
D) '\0'-'0'

(25) 有下列程序:

```
fun(int x,int y){return (x+y);}
main()
{
    int a=1,b=2,c=3,sum;
    sum=fun(a++ ,b++ ,a+b),c++ ;
    printf("%d\n",sum);
}
```

执行后的输出结果是 ()。

- A) 6
C) 8

- B) 7
D) 9

(26) 有下列程序:

```
main()
{
    char s[ ]="abcde";
    s+=2;
    printf("%d\n",s[0]);
}
```

执行后的结果是 ()。

- A) 输出字符 a 的 ASCII 码
C) 输出字符 c

- B) 输出字符 c 的 ASCII 码
D) 程序出错

(27) 有下列程序:

```
fun (int x,int y)
{
    static int m=0,i=2;
    i+=m+1; m=i+x+y; return m;
}
```

main ()

```
{
    int j=1, m=1, k;
    k=fun(j,m); printf("%d",k);
    k=fun(j,m); printf("%d\n",k);
}
```

执行后的输出结果是 ()。

- A) 5, 5
C) 11, 11

- B) 5, 11
D) 11, 5

(28) 有下列程序:

```
fun(int x)
{
    int p;
    if(x==0 || x==1) return(3);
    p=x-fun(x-2);
    return p;
}
```