

全国计算机等级考试 历年试卷汇编及详解

2005年4月~2008年4月全部真题+详尽解析

二级

C语言

○ 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: 18.00元



全国计算机等级考试 历年试卷汇编及详解 一级 C语言



化学工业出版社

最新大纲
2008年9月考试专用

全国计算机等级考试
新思路系列辅导用书



附1CD

全国计算机等级考试 历年试卷汇编及详解

2005年4月~2008年4月全部真题+详尽解析

二级

C语言

- 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

编

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



化学工业出版社

2008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4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——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。

全国计算机等级考试 历年试卷汇编及详解

2005年4月~2008年4月全部真题+详尽解析

二级C语言 (2008年9月考试专用)

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
· 北 京 ·

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语言/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,2008.5
(全国计算机等级考试新思路系列辅导用书)

ISBN 978-7-122-02453-4

ISBN 978-7-900239-49-5 (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(2008)第037312号

策 划: 张 立 陈 静

装帧设计: 尹琳琳

责任编辑: 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: 三河市延风装订厂

787mm×1092mm 1/8 印张9 字数282千字 2008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: 18.00元(含1CD)

版权所有 违者必究

等考捷径——练真题

每年都有很多考生问同样一个问题，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排版的，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，只不过有的题目换算参数，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

做完了真题，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！

目录 Chapter

2005 年 4 月全国计算机等级考试笔试试卷	(共 6 页)
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)
2005 年 4 月笔试试卷参考答案及详解	(共 4 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)

D) scanf("%d%s",b,&c);

C) scanf("%d%s",b,c);

(19) 有下列程序:

```
main()
{ int m,n,p;
  scanf("m=%dn=%dp=%d",&m,&n,&p);
  printf("%d%d%d\n",m,n,p);
}
```

若想从键盘上输入数据,使变量 m 中的值为 123, n 中的值为 456, p 中的值为 789, 则正确的输入是 ()。

- A) m=123 n=456 p=789
B) m=123 n=456 p=789
C) m=123,n=456,p=789
D) 123 456 789

(20) 下列程序的输出结果是 ()。

```
main()
{ int a,b,d=25;
  a= d/10%9;
  b=a&&(-1);
  printf("%d,%d\n",a,b);
}
```

A) 6,1
B) 2,1
C) 6,0
D) 2,0

(21) 下列程序的输出结果是 ()。

```
main()
{ int i=1,j=2,k=3;
  if(i++ += 1&&(+ +j= 3 || k++ += 3)) printf("%d %d %d\n",i,j,k);
}
```

A) 1 2 3
B) 2 3 4
C) 2 2 3
D) 2 3 3

(22) 若整型变量 a、b、c、d 中的值依次为: 1、4、3、2。则条件表达式 a<b? a: c<d? c: d 的值是 ()。

A) 1
B) 2
C) 3
D) 4

(23) 下列程序的输出结果是 ()。

```
main()
{ int p[8]={11,12,13,14,15,16,17,18},i=0,j=0;
  while(i++ < 7)
  { if(p[i]%2) j++ = p[i];
    printf("%d\n",j);
  }
```

A) 42
B) 45
C) 56
D) 60

(24) 下列程序的输出结果是 ()。

```
main()
{ char a[7]="a0\0a0\ 0"; int i,j;
  i=sizeof(a);
  j=strlen(a);
```

printf("%d %d\n",i,j);

- A) 2 2
B) 7 6
C) 7 2
D) 6 2

(25) 下列能正确定义一维数组的选项是 ()。

A) int a[5]={0,1,2,3,4,5};
B) char a[]={0,1,2,3,4,5};
C) char a={'A','B','C'};
D) int a[5]="0123";

(26) 下列程序的输出结果是 ()。

```
int f1(int x,int y){return x>y?x:y;}
int f2(int x,int y){return x>y?y:x;}
main()
```

{ int a=4,b=3,c=5,d=2,e,f,g;

e=f2(f1(a,b),f1(c,d));

f=f1(f2(a,b),f2(c,d));

g=a+b+c+d-e-f;

printf("%d,%d,%d\n",e,f,g);

}

A) 4,3,7
B) 3,4,7
C) 5,2,7
D) 2,5,7

(27) 已有定义: char a[]="xyz",b[]={ 'x', 'y', 'z' };, 下列叙述中正确的是 ()。

A) 数组 a 和 b 的长度相同
B) a 数组长度小于 b 数组长度
C) a 数组长度大于 b 数组长度
D) 上述说法都不对

(28) 下列程序的输出结果是 ()。

```
void f(int *x, int *y)
{ int t;
  t=*x,*x=*y,*y=t;
}
```

main()

{ int a[8]={1,2,3,4,5,6,7,8},i,*p,*q;

p=a;q=&a[7];

while(p<q)

{ f(p,q); p++; q--; }

for (i=0;i<8;i++) printf("%d",a[i]);

}

A) 8,2,3,4,5,6,7,1

B) 5,6,7,8,1,2,3,4

C) 1,2,3,4,5,6,7,8

D) 8,7,6,5,4,3,2,1

(29) 下列程序的输出结果是 ()。

```
main()
{ int a[3][3],*p,i; p=&a[0][0];
  for(i=0;i<9;i++) p[i]=i;
  for(i=0;i<3;i++) printf("%d",a[1][i]); }
```

A) 0 1 2

B) 1 2 3

C) 2 3 4

D) 3 4 5

(30) 下列叙述中错误的是 ()。

- 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) 在程序执行中, 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”的输出信息
- D) 可以通过赋初值的方式确定数组元素的个数

(31) 下列程序的输出结果是 ()。

```
#define N 20
fun(int a[],int n,int m)
{
    int i,j;
    for(i=m;i>n;i--)a[i+1]=a[i];
}
main()
{
    int i,a[N]={1,2,3,4,5,6,7,8,9,10};
    fun(a,2,9);
    for(i=0;i<5;i++) printf("%d",a[i]);
}
```

(32) 运行下列程序时, 输入 1 2 3<CR>的输出结果为 ()。

```
main()
{
    int a[3][2]={0},(*ptr)[2],i,j;
    for(i=0;i<2;i++)
    {
        ptr=a+i; scanf("%d",ptr); ptr++;
    }
    for(i=0;i<3;i++)
    {
        for(j=0;j<2;j++)
        {
            printf("%2d",a[i][j]);
            printf("\n");
        }
    }
}
```

(33) 下列程序的输出结果是 ()。

```
prt(int *m,int n)
{
    int i;
    for(i=0;i<n;i++)m[i]++;
}
```

main()

```
{
    int a[]={1,2,3,4,5},i;
    prt(a,5);
    for(i=0;i<5;i++) printf("%d",a[i]);
}
```

A) 1,2,3,4,5

C) 3,4,5,6,7

(34) 下列程序的输出结果是 ()。

```
main()
{
    int a[]={1,2,3,4,5,6,7,8,9,0},*p;
    for(p=a;p<a+10;p++)
        printf("%d",*p);
}
```

A) 1,2,3,4,5,6,7,8,9,0,

C) 0,1,2,3,4,5,6,7,8,9,

(35) 下列程序的输出结果是 ()。

```
#define P 3
void F(int x){ return(P*x*x); }
main()
{
    printf("%d\n",F(3+5));
}
```

A) 192

B) 29

(36) 下列程序的输出结果是 ()。

```
main()
{
    int c=35; printf("%d\n",c&c);
}
```

A) 0

B) 70

(37) 下列叙述中正确的是 ()。

- A) 预处理命令行必须位于源文件的开头
- B) 在源文件的一行上可以有多个预处理命令
- C) 宏名必须用大写字母表示
- D) 宏替换不占用程序的运行时间

(38) 若有下列说明和定义。

```
union dt
{
    int a; char b; double c; }data;
```

下列叙述中错误的是 ()。

- A) data 的每个成员起始地址都相同
- B) 变量 data 所占内存字节数与成员 c 所占字节数相等
- C) 程序段: data.a=5;printf("%f\n",data.c);输出结果为 5.000000
- D) data 可以作为函数的实参

(39) 下列语句或语句组中, 能正确进行字符串赋值的是 ()。

- A) char *sp; *sp="right!";
- B) char s[10]; s="right!";

D) char *sp="right!";

C) char s[10]; *s="right!";

(40) 设有如下说明:

typedef struct ST

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

则下列叙述中正确的是 ()。

A) 以上的说明形式非法

C) NEW 是一个结构体类型名

(41) 下列程序的输出结果是 ()。

```
main()
{ int a=1,b;
  for(b=1;b<=10;b++)
  { if(a==8) break;
    if(a%2==1)
    { a+=5; continue; }
    a=3;
  }
  printf("%d\n",b);
}
```

A) 3 B) 4 C) 5 D) 6

(42) 下列程序的输出结果是 ()。

```
main()
{ char s[]="159",*p;
  p=s;
  printf("%c",*p++);
  printf("%c",*p++);
}
```

A) 15 B) 16 C) 12 D) 59

(43) 有下列函数:

```
fun(char *a,char *b)
{ while(*a!='\0')&&(*b!='\0')&&(*a==*b)}
{ a++; b++; }
return(*a-*b);
}
```

该函数的功能是 ()。

A) 计算 a 和 b 所指字符串的长度之差
B) 将 b 所指字符串复制到 a 所指字符串中
C) 将 b 所指字符串连接到 a 所指字符串后面
D) 比较 a 和 b 所指字符串的大小

(44) 有下列程序:

```
main()
```

```
{ int num[4][4]={ {1,2,3,4},{5,6,7,8},{9,10,11,12},{13,14,15,16}};i,j;
  for(i=0;i<4;i++)
  { for(j=0;j<i;j++)
    printf("%4c",' ');
    for(j=__;j<4;j++)
      printf("%4d",num[i][j]);
    printf("\n");
  }
}
```

若要按下列形式输出数组右上半三角。

```
1 2 3 4
   6 7 8
      11 12
         16
```

则在程序下画线处应填入的是 ()。

A) i-1 B) i C) i+1 D) 4-i

(45) 下列程序的输出结果是 ()。

```
point(char*p){ p+=3; }
main()
{ char b[4]={ 'a','b','c','d' },*p=b;
  point(p); printf("%c\n",*p);
}
```

A) a B) b C) c D) d

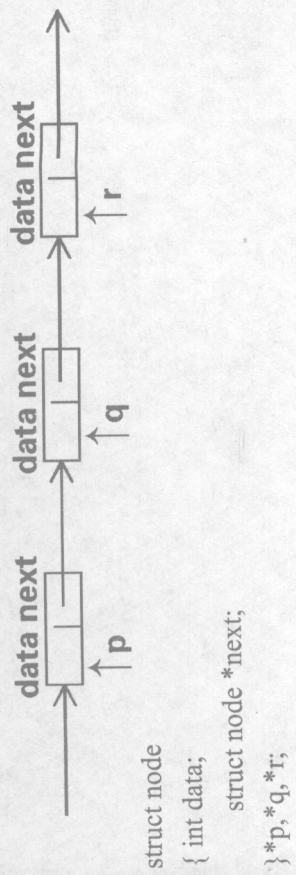
(46) 程序中若有下列说明和定义语句:

```
char fun(char *);
main()
{ char *s="one",a[5]={0},(*f1)( )=fun,ch;
  ...
}
```

下列选项中对 fun() 函数的正确调用语句是 ()。

A) (*f1)(a); B) *f1(*s); C) fun(&a); D) ch=*f1(s);

(47) 有下列结构体说明和变量定义, 如图所示, 指针 p、q、r 分别指向此链表中的三个连续结点。



现要将 q 所指结点从链表中删除，同时要保持链表的连续，下列不能完成指定操作的语句是 ()。

- A) p->next=q->next;
 - B) p-next=p->next->next;
 - C) p->next=r;
 - D) p=q->enxt;
- (48) 下列对结构体类型变量 td 的定义中，错误的是 ()。
- A) typedef struct aa { int n; float m; }AA;
 - B) struct aa { int n; float m; }AA; td;
 - C) struct { int n; float m; }aa;
 - D) struct { int n; float m; }td;
- (49) 下列与函数 fseek(fp,0L,SEEK_SET)有相同作用的是 ()。
- A) feoff(fp)
 - B) ffill(fp)
 - C) fgetc(fp)
 - D) rewind(fp)
- (50) 有下列程序:

```
#include <stdio.h>
void WriteStr(char *fn,char *str)
{ FILE *fp;
  fp=fopen(fn,"w"); fputs(str,fp); fclose(fp);
}
main()
{ WriteStr("t1.dat","start");
  WriteStr("t1.dat","end");
}
```

程序运行后，文件 t1.dat 中的内容是 ()。

- A) start
- B) end
- C) startend
- D) endrt

二、填空题 (每空 2 分，共 40 分)

- (1) 某二叉树中度为 2 的结点有 18 个，则该二叉树中有 【1】 个叶子结点。
- (2) 在面向对象方法中，类的实例称为 【2】。
- (3) 诊断和改正程序中错误的工作通常称为 【3】。
- (4) 在关系数据库中，把数据表示成二维表，每一个二维表称为 【4】。
- (5) 问题处理方案正确而完整的描述称为 【5】。
- (6) 下列程序运行时若从键盘输入：10 20 30<CR>。输出结果是 【6】。

```
#include <stdio.h>
main()
{ int i=0, j=0, k=0;
```

```
scanf("%d%d*d", &i, &j, &k);
printf("%d%d%d\n", i, j, k);
}
```

- (7) 下列程序运行后的输出结果是 【7】。
- ```
#define S(x) 4*x*x+1
main()
{ int i=6, j=8;
 printf("%d\n", S(i+j)); }
```

- (8) 下列程序运行后的输出结果是 【8】。
- ```
main()
{ int a=3, b=4, c=5, t=99;
  if(b<a&& a<c)t=a; a=c; c=t;
  if(a<c&& b<c)t=b; b=a; a=t;
  printf("%d%d%d\n", a, b, c); }
```

- (9) 下列程序运行后的输出结果是 【9】。
- ```
main()
{ int a, b, c;
 a=10; b=20; c=(a%b<1) || (a/b>1);
 printf("%d %d %d\n", a, b, c);
}
```

- (10) 下列程序运行后的输出结果是 【10】。
- ```
main()
{ char c1, c2;
  for(c1='0'; c2='9'; c1<c2; c1++, c2--) printf("%c%c", c1, c2);
  printf("\n");
}
```

- (11) 已知字符 A 的 ASCII 代码值为 65，下列程序运行时若从键盘输入：B33<CR>。则输出结果是 【11】。
- ```
#include <stdio.h>
main()
{ char a, b;
 a=getchar(); scanf("%d", &b);
 a=a-'A'+0; b=b*2;
 printf("%c %c\n", a, b);
}
```

- (12) 下列程序中，fun()函数的功能是求 3 行 4 列二维数组每行元素中的最大值。请填写。
- ```
void fun(int, int, int(*)[4], int*);
main()
{ int a[3][4]={ {12,41,36,28}, {19,33,15,27}, {3,27,19,1} };
  fun(3,4,a,b);
  for(i=0; i<3; i++) printf("%4d", b[i]);
}
```

```

printf("n"); }
void fun(int m, int n, int ar[ ][4], int *br)
{ int i, j, x;
  for(i=0; i<m; i++)
  { x=ar[i][0];
    for(j=0; j<n; j++)
    if(x<ar[i][j]) x=ar[i][j];
    【12】=x;
  }
}
(13) 下列程序运行后的输出结果是 【13】。
void swap(int x, int y)
{ int t;
  t=x; x=y; y=t; printf("%d %d", x, y); }
main()
{ int a=3, b=4;
  swap(a, b); printf("%d %d\n", a, b);
}

```

(14) 下列程序运行后的结果是 【14】。

```

#include <string.h>
void fun(char *s, int p, int k)
{ int i;
  for(i=p; i<k-1; i++) s[i]=s[i+2]; }
main()
{ char s[ ]="abcdefg";
  fun(s, 3, strlen(s)); puts(s);
}
(15) 下列程序运行后的输出结果是 【15】。
#include <string.h>
main()
{ char ch[ ]="abc", x[3][4]; int i;
  for(i=0; i<3; i++) strcpy(x[i], ch);
  for(i=0; i<3; i++) printf("%s", &x[i][i]);
  printf("n");
}
(16) 下列程序运行后的输出结果是 【16】。
fun(int a)
{ int b=0; static int c=3;
  b++; c++;
  return(a+b+c);
}

```

```

(17) 下列程序运行后的输出结果是 【17】。
struct NODE
{ int k;
  struct NODE *link; };
main()
{ struct NODE m[5], *p=m, *q=m+4;
  int i=0;
  while(p!=q)
  { p->k++; i++; p++;
    q->k=i++; q--;
  }
  q->k=i;
  for(i=0; i<5; i++) printf("%d", m[i].k);
  printf("n");
}

```

(18) 下列程序中 huiwen() 函数的功能是检查一个字符串是否是回文，当字符串是回文时，函数返回字符串；yes!，否则函数返回字符串：no!，并在主函数中输出。所谓回文即正向与反向的拼写都一样，例如：adga。请填写。

```

#include <string.h>
char* huiwen(char *str)
{ char *p1, *p2; int i, t=0;
  p1=str; p2= 【18】;
  for(i=0; i<strlen(str)/2; i++)
  if(*p1++ != *p2--) { t=1; break; }
  if( 【19】 )
  return("yes!");
  else
  return("no!");
}
main()
{ char str[50];
  printf("Input:"); scanf("%s", str);
  printf("%s\n", 【20】 );
}

```

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)

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) 程序的测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
C) 大量数据存储问题
B) 数据安全问题
D) 简化数据维护
- (9) 设有如下关系表:

R			S			T		
A	B	C	A	B	C	A	B	C
1	1	2	3	1	3	1	1	2
2	2	3				2	2	3
						3	1	3

- 则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) $T=R/S$
- (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 `getchar()` 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) 有下列程序段:
- ```
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  
C) OLD 是一个结构体类型  
B) NODE 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

B) k%2==1

C) (k%2)!=0

D) !k%2==1

(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)

B) sin(0.5)\*sin(0.5)/3

C) pow(sin(0.5),2)/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 nl=n2=10;

B) char c=32;

C) float f=f+1.1;

D) double x=12.3E2.5;

(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= * π *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

B) -k++

C) k%int(f)

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)

B) (a<b)&&!c || 1

C) a && b

D) a || (b+b)&&(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

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

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

B) 任意负偶数

C) 任意正偶数

D) 任意负奇数

(27) 设变量已正确定义,则以下能正确计算 f=n! 的程序段是 ( )。

A) f=0;

B) f=1;

C) for(i=1;i<=n;i++) f\*=i;

D) for(i=1;i<=n;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;

B) p=q;

C) \*p=&n1;

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

B) 0

C) x 的地址

D) p 的地址

(30) 设 fun() 函数的定义形式为 ( )。

A) fun(char ch, float x) { ... }

B) fun('abc', 3.0);

C) fun('65', 2.8);

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 ( )。

```
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]);
}
```

```
main()
{ int i1,i2;
 i1=f(8)/f(4); i2=(f(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%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;
```

```
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 3 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", "aabdfg", "abbd", "dcdbe", "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) abedcfg

(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);
}
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=i+【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+'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 strcat() 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。请填空。

```

#include <string.h>
void strcat(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]="abcdefgh";
 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->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

```

请填空。

```

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][N], int b[N][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方法的是 ( )。
- 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
- ```
graph TD
    A[A] --> B[B]
    A --> C[C]
    B --> D[D]
    B --> E[E]
    C --> F[F]
```
- 进行后序遍历的结果为 ()。
- A) ABCDEF B) DBEAF C) ABDEC D) DEBFC
- (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

(11) 下列不合法的用户标识符是 ()。

- A) j2_KEY
C) 4d
B) Double
D) _8_

(12) 下列不合法的数值常量是 ()。

- A) 011
C) 8.0E0.5
B) 1e1
D) 0xabcd

(13) 下列不合法的字符常量是 ()。

- A) '018'
C) '\'
B) '\n'
D) 'xccc'

(14) 表达式 $3.6-5/2+1.2+5\%2$ 的值是 ()。

- A) 4.3
C) 3.3
B) 4.8
D) 3.8

(15) 下列能正确定义字符串的语句是 ()。

- A) char str[] = {'064'};
C) char str = "
B) char str[] = "064";
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() C) getchar()
B) gets() D) scanf()

(18) 下列 4 个程序中, 完全正确的是 ()。

- A) #include <stdio.h>
main()
{
/* programming */
printf("programming! \n");
}
C) #include <stdio.h>
main()
{
/* programming */
printf("programming! \n");
}
B) #include <stdio.h>
main()
{
/* programming */
printf("programming! \n");
}
D) include <stdio.h>
main()
{
/* programming */
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 ()。

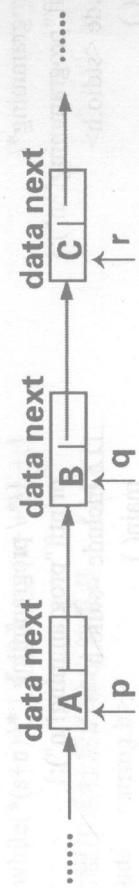
- A) switch(x)
{
case 1.0: printf("x\n");
case 2.0: printf("x\n");
}
C) switch(a+b)
{
case 1: printf("x\n");
case 2+1: printf("x\n");
}
B) switch((int)x)
{
case 1: printf("x\n");
case 2: printf("x\n");
}
D) switch(a+b)
{
case 1: printf("x\n");
case c: printf("x\n");
}

(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 语句正确读入数据的程序段是 ( )。
A) *p = &a; scanf("%f",p); B) *p = &a; scanf("%f",p);
C) p = &a; scanf("%f",*p); D) p = &a; scanf("%f",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;
(26) 现有将 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->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;
(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 feof(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,&c6);
    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()

```

```

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 feof(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,&c6);
    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()

```

