



全国计算机等级考试(二级)

C语言


全国计算机等级考试命题研究组

赵志强 主编

标准化命题预测试卷

内附

- 10套标准化命题预测试卷及解答
- 2套全真试题及解答(2001年4月、9月)
- 答题卡

 考前30天 确保高分
临考入境 真我测试

全国计算机等级考试二级笔试试卷

基础部分和 C 语言程序设计

1

注意事项

- 一、考生应严格遵守考场规则,得到监考人员指令后方可作答。
- 二、考生拿到试卷后应首先将自己的姓名、准考证号等内容涂写在答题卡的相应位置上。
- 三、选择题答案必须用铅笔填涂在答题卡的相应位置上,填空题的答案必须用蓝、黑色钢笔或圆珠笔写在答题卡的相应位置上,答案写在试卷上无效。
- 四、注意字迹清楚,保持卷面整洁。
- 五、考试结束将试卷和答题卡放在桌上,不得带走。待监考人员收毕清点后,方可离场。

* * * 版权所有、任何单位或个人不得保留、复制和出版,违者必究 * * *

全国计算机等级考试二级笔试标准化预测试卷

基础部分和 C 语言程序设计

(考试时间 120 分钟,满分 100 分)

一、选择题((1)~(40)题每题 1 分,(41)~(50)题每题 2 分,共 60 分)

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

- (1)第三代计算机的电子元器件采用的是 B
- A)晶体管
B)中、小规模集成电路
C)大规模集成电路
D)微处理器集成电路
- (2)在微机系统中,硬件与软件的关系是 B
- A)在一定条件下可以相互转化的关系
B)等效关系
C)特有的关系
D)固定不变的关系
- (3)CPU 是由 D 组成的
- A)内存储器和控制器
B)控制器和运算器
C)内存储器和运算器
D)内存储器、控制器和运算器
- (4)计算机软件系统通常分为 A
- A)系统软件和应用软件
B)高级软件和一般软件
C)军用软件和民用软件
D)管理软件和控制软件
- (5)二进制数运算 1110×1101 的结果是 C
- A)10110110 B)00110110 C)01111110 D)10011010
- (6)计算机病毒可以使整个计算机瘫痪,危害极大,计算机病毒是 B
- A)一种芯片
B)一段特制的程序
C)一种生物病毒
D)一条命令
- (7)目前世界上最大的计算机互联网络是 C
- A)ARPA 网 B)IBM 网 C)Internet D)Intranet
- (8)下面关于 DEL 命令的说法中,不正确的说法是 D
- A)利用文件名通配符可以删除一批文件
B)不能删除 IBMDOS.COM、IBMBIO.COM
C)在启动盘的根目录下不能使用命令 DEL *.*
D)可以删除隐藏文件和系统文件
- (9)若 C 盘中已有 DOS 操作系统,并且将 DOS 系统盘插入 A 驱动器中,成功启动系统后,在屏幕上出现的系统提示符是 A
- A)A > B)B > C)D > D)E >
- (10)下列 DOS 命令中,外部命令是 D
- A)COPY B)DIR C)TYPE D)SYS
- (11)分页显示当前文件目录的 DOS 命令是 B
- A)DIR/W B)DIR/P C)DIR D)DIR A:
- (12)当前盘为 A 盘,要显示 C 盘根目录下 AUTOEXEC.BAT 文件的内容,应使用的 DOS 内部命令是 B

A) DIR C: \ AUTOEXEC. BAT

B) DIR \ AUTOEXEC. BAT

C) TYPE \ AUTOEXEC. BAT

D) COPY C: \ AUTOEXEC. BAT CON

(13) 下面四条 DOS 命令中, 错误的一条是

A) REN B: \ * .EXE * .COM

B) REN B: \ ABC. FOR B: \ XYZ. TXT

C) REN B: \ ? AB. TXT ? XYZ. DOC

D) REN B: \ ABC. TXT XYZ. *

(14) 设驱动器 B 中软盘上的目录结构如图 1 所示:

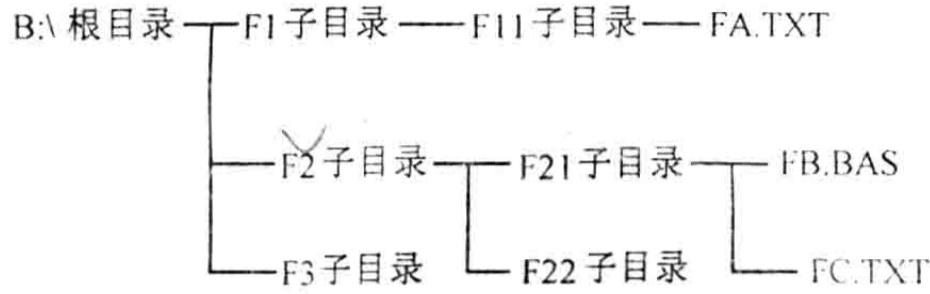


图 1 软盘 B 的目录结构

设当前目录为 F2, 则把 A 盘根目录下的 PROG. BAS 文件复制到 B 盘 F3 子目录中的命令是

A) COPY PROG. BAS B:

B) COPY A: PROG. BAS B: \

C) COPY A: \ PROG. BAS B: \ F3

D) COPY A: PROG. BAS B: \ F3 \

(15) 设 A 盘为启动盘, 其目录结构如图 2 所示

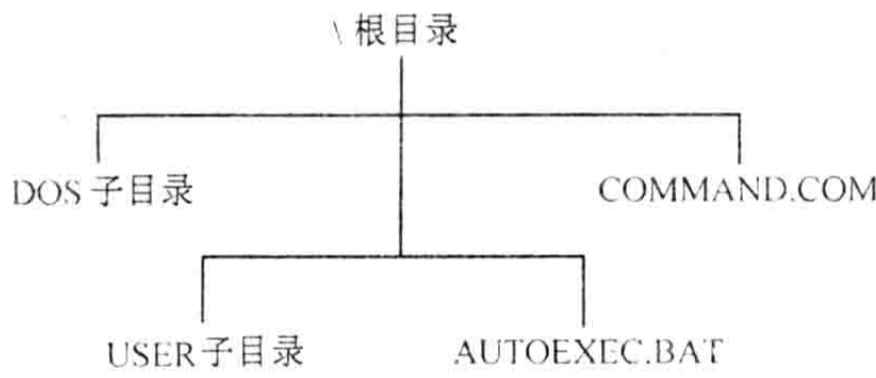


图 2 启动盘 A 的目录结构

其中文件 AUTOEXEC. BAT 的内容为:

MD \ XYZ

CD \ DOS \ USER

则由 A 盘启动成功后, A 盘的当前目录是

A) \ DOS \ USER

B) \ XYZ

C) \ 1

D) \ DOS

(16) 为了要将 A 盘当前目录(不是根目录)中所有扩展名为 .C 的文件同名复制到 A 盘的目录 \ USER 中, 正确的命令是

A) COPY A: * .C USER \ * .C

B) COPY A: * .C \ USER

C) COPY A: * .C A: USER \

D) COPY A: * .C A: USER \ * .C

(17) 以下选项中属于 C 语言的数据类型是

A) 复数型

B) 双精度型

C) 逻辑型

D) 集合型

(18) 在 C 语言中, 不正确的 int 类型的常数是

A) 32768

B) 0

C) 037

D) 0xAF

(19) 语句: printf("%d", (a = 2) && (b = -2)); 的输出结果是

A) 无输出

B) 结果不确定

C) -1

D) 1

(20) 设有定义: int x = 1, y = -1; 则语句: printf("%d \ n", (x - - & + + y)); 的输出结果是

A) 1

B) 0

C) -1

D) 2

(21) 当 c 的值不为 0 时, 能正确将 c 的值赋给变量 a, b

A) c = b = a;

B) (a = c) || (b = c)

C) (a = c) && (b = c)

D) a = c = b

(22) 以下程序的输出结果是

```
main( )
{ int x = 10, y = 10, i;
  for(i = 0; x > 8; y = ++i)
    printf("%d %d", x--, y);
}
```

- A) 10 1 9 2 B) 9 8 7 6 C) 10 9 9 0 D) 10 10 9 1

(23) 以下程序的输出结果是

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

- A) 5 2 B) 5 1 C) 5 3 D) 9 7

(24) 以下程序的运行结果是

```
main()
{ char a[10] = {'1','2','3','4','5','6','7','8','9','0'}, *p;
  int i;
  i = 8;
  p = a + i;
  printf("%s \n", p - 3);
}
```

- A) 6 B) 6789 C) '6' D) 789

(25) 以下程序的运行结果是

```
#include <stdio.h>
main( )
{
  int a[] = {1,2,3,4,5,6,7,8,9,10,11,12};
  int *p = a + 5, *q = NULL;
  *q = *(p + 5);
  printf("%d %d \n", *p, *q);
}
```

- A) 运行后报错 B) 6 6 C) 6 12 D) 5 5

(26) 以下说法正确的是

- A) #define 和 printf 都是 C 语句 B) define 是 C 语句, 而 printf 不是
 C) printf 是 C 语句, 但 #define 不是 D) #define 和 printf 都不是 C 语句

(27) 语句: printf("%d \n", 12 & 012); 的输出结果是

- A) 12 B) 8 C) 6 D) 012

(28) 能正确表示 a 和 b 同时为正或同时为负的逻辑表达式是

- A) (a >= 0 || b >= 0) && (a < 0 || b < 0) B) (a >= 0 && b >= 0) && (a < 0 && b < 0)
 C) (a + b > 0) && (a + b <= 0) D) a * b > 0

(29) 以下程序的输出结果是

```
main( )
{ int n = 4;
  while (n --) printf("%d", --n);
}
```

- A) 2 0 B) 3 1 C) 3 2 1 D) 2 1 0

(30)若已定义: int a[9], *p = a; 并在以后的语句中未改变 p 的值, 不能表示 a[1]地址的表达式为

- A) p + 1 B) a + 1 C) a + + D) + + p

(31)以下程序的输出结果是

```
main ( )
{ int k = 17;
  printf ("%d, %o, %x \n", k, k, k);
}
```

- A) 17, 021, 0x11 B) 17, 17, 17 C) 17, 0x11, 021 D) 17, 21, 11

117
00014001
21

(32)以下程序的输出结果是

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

- A) -1 1 B) 0 1 C) 1 0 D) 0 0

(33)若有说明: long *p, a; 则不能通过 scanf 语句正确给输入项读入数据的程序段是

- A) *p = &a; scanf ("%ld", p); B) p = (long *) malloc(8); scanf ("%ld", p);
C) scanf ("%ld", p = &a); D) scanf ("%ld", &a);

(34)定义 s 为合法的结构体变量中有错误, 其中修改正确的是

```
struct
{ int a = 1;
  char b[10];
} s;
```

A) "struct" 改为 "typedef"
C) "char" 改为 "int"

- B) "char" 改为 "typedef"
D) "int + a = 1;" 改为 "int a;"

(35)下面程序的输出是

```
main( )
{ int x = 2;
  while ( x - - );
  printf ("%d \n", x); }
```

- A) -1 B) 1 C) 0 D) 出错信息

(36)下面程序的输出是

```
main( )
{ int a[ ] = {2, 4, 6}, *ptr = &a[0], x = 8, y, z;
  for(y = 0; y < 3; y + + )
    z = (* (ptr + y) < x) ? * (ptr + y) : x;
  printf ("%d \n", z); }
```

- A) 2 B) 4 C) 6 D) 8

(37)以下程序的输出结果是

```
# include < string . h >
main( )
{ char * a = "abcdefghi"; int k;
```

```

    fun(a); puts(a);
}
fun ( char * s)
{ int x, y; char c;
  for ( x=0, y= strlen(s) - 1; x < y; x+ + ,y- - )
    { c = s[y];s[y] = s[x];s[x] = c;}
}

```

- A) ihgfedcba B) abcdefghi C) abcdedcba D) ihgfefghi

(38) 以下程序的输出结果是

```

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) 值不确定

(39) 以下程序的输出结果是

```

main()
{ union { char i[2];
         int k;
         } r;
  r.i[0] = 2; r.i[1] = 0;
  printf("%d \ n", r.k);
}

```

- A) 2 B) 1 A) 0 D) 不确定

(40) 设有以下定义和语句, 则输出的结果是(用 small 模式编译, 指针变量占 2 个字节)

```

struct date
{ long * cat;
  struct date * next;
  double dog;
} too;
printf ("%d", sizeof(too));

```

- A) 20 B) 16 C) 14 D) 12

(41) 下面程序段的输出结果是

```

int a, b, c;
a = 10, b = 50, c = 30;
if(a > b) a = b,
b = c;
c = a;
printf("a = %d, b = %d, c = %d", a, b, c);

```

- A) a = 10, b = 50, c = 10 B) a = 10, b = 30, c = 10
 C) a = 50, b = 30, c = 10 D) a = 50, b = 30, c = 50

(42) 下面程序的输出结果是

```
# include < stdio. h >
main()
{int x = 1, y = 31;
char z = 'A';
printf("%d", x&15&&z < 'a');
}
```

- A)0 B)1 C)2 D)3

(43)下面程序的输出结果是

```
# include < stdio. h >
main()
{int a = 0, b = 0, c = 0;
if( ++a > 0 || ++b > 0)
++c; /
printf(" \ na = %d, b = %d, c = %d", a, b, c);
}
```

- A)a = 0, b = 0, c = 0 B)a = 1, b = 1, c = 1
C)a = 1, b = 0, c = 1 D)a = 0, b = 1, c = 1

(44)下面程序的输出结果是

```
# include < stdio. h >
main()
{int a, b, c;
int x = 5, y = 10;
a = ( --y == x++ ) ? --y : ++x;
b = y + +;
c = x;
printf(" \ na = %d, b = %d, c = %d", a, b, c);
```

- A)a = 6, b = 9, c = 7 B)a = 6, b = 9, c = 6
C)a = 7, b = 9, c = 6 D)a = 7, b = 9, c = 7

(45)下面程序段的输出结果是

```
int a = 2, b = 3, c = 4, d = 5;
int m = 2, n = 2;
a = (m = a > b) && (n = c > d) + 5;
printf("%d", n);
```

- A)0 B)1 C)2 D)5

(46)当 a = 1, b = 3, c = 5, d = 4 时, 执行完下面一段程序后 x 的值是

```
if(a < b)
if(c < d) x = 1;
else
if(a < c)
if(b < d) x = 2;
else x = 3;
else x = 6;
else x = 7;
```

- A)1 B)2 C)3 D)6

(47)下面程序段

二、填空题(每空 2 分,共 40 分)

请将每一个空的正确答案写在答题卡【1】~【20】序号的横线上,答在试卷上不得分。

- (1)在 DOS 系统中,自动批处理文件的文件名为 【1】AUTO EXEC. BAT
- (2)设当前盘为 A 盘,当前 DOS 系统提示符为“A>”,在不改变当前盘的情况下,显示 C 盘的当前目录路径,应使用的命令是 【2】CD C:。
- (3)DOS 命令分为内部命令与外部命令,COMP 命令属于 【3】外 命令。
- (4)要将当前盘当前目录下的子目录 XYZ 中所有扩展名为 .BAK 的文件名改成扩展名为 .FOR,应使用的 DOS 命令为 【4】REN XYZ * .BAK * .FOR
- (5)在 Windows 中,为了终止一个应用程序的运行,首先单击该应用程序窗口中的控制菜单框,然后在控制菜单中单击 【5】关闭或 close
- (6)语句 `int i = 0x80; printf("%d", i >> = 1);` 的执行结果是 【6】0。
- (7)若 a、b 和 c 均是整型变量,则计算表达式后,a 值为 【7】1,b 值为 【8】1,c 值为 【9】2。

$a = (b = 4) + (c = 2)$

- (8)以下程序运行后的输出结果是 【10】。

```
void main()
{
    int a, b, c = 241;
    a = c / 100 % 9;
    b = -1 && -1;
    printf("%d, %d", a, b);
}
```

- (9)以下程序运行后的输出结果是 【11】。

```
输入 The Programming #
main()
{
    int j = 0, i = 0;
    char ch;
    while((ch = getchar()) != "#")
    {
        switch (ch)
        {
            case 'a': case 'A':
            case 'e': case 'E':
            case 'o': case 'O':
            default: i++;
            case 'i': case 'I':
            case 'u': case 'U':
            j++;
        }
    }
    printf("%d, %d", i, j);
}
```

- (10)以下程序运行后的输出结果是 【12】。

```
main()
{
    int i, j;
    for(i = 0; i < 3; i++, i++)
        for(j = 4; j >= 0; j--)
            if((j + i) % 2)
                j--;
    printf("%d", j);
}
```

```

        continue; }
        -- i;
        j -- ; printf("%d", j);
    }
}

```

(11) 以下程序运行后的输出结果是 【13】。

```

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

```

(12) 以下程序运行后的输出结果是 【14】。

```

#include <stdio.h>
main()
{int x = 1, y = 0, a = 0, b = 0;
switch(x)
{case 1:
switch(y)
{case 0: a++; break;
case 1: b++; break;
}
case 2:
a++; b++; break;
case 3:
a++; b++;
}
printf("\na = %d, b = %d", a, b);
}

```

(13) 在下述程序中, 判断 $i > j$ 共执行了 【15】 次。

```

#include <stdio.h>
main()
{int i = 0, j = 10, k = 2, s = 0;
for(;;)
{i += k;
if(i > j)
{printf("%d \n", s);
break;
}
s += i;
}
}

```

(14) 以下程序运行后的输出结果是 【16】。

```

main()
{int m = 37, n = 13;

```

```

while(m! = n)
    { while(m > n)
        m = m - n;
      while(n > m)
        n = m;
    }
printf("m = %d \ n", m);
}

```

(15) 以下程序运行后的输出结果是 【17】。

```

#include <stdio.h>
main()
{ int x = 9;
  for( ; x > 0; x - - )
  { if(x % 3 == 0)
    { printf("%d", - - x);
      continue;
    }
  }
}

```

(16) 以下程序从读入的整数数据中,统计大于零的整数个数和小于零的整数个数。用输入零来结束输入,程序中用变量 i 统计大于零的整数个数,用变量 j 统计小于零的整数个数。但程序不完整,请填空。

```

#include <stdio.h>
main()
{ int n, i, j;
  printf("输入非零的整数(输入零结束程序)");
  i = j = 0;
  scanf("%d", &n);
  while 【18】 n
  { if(n > 0)
    { i = i + 1;
      if(n < 0)
        j = j + 1;
      【19】 scanf("%d", &n)
    }
  }
  printf("i = %4d, j = %4d \ n", i, j);
}

```

(17) 以下程序运行后的输出结果是 【20】。

从键盘输入 3 和 4

```

#include <stdio.h>
main()
{ int a, b, s;
  scanf("%d %d", &a, &b);
  s = a;
  if(a < b)
  s = b; s = s * s;
  printf("%d", s);
}

```

全国计算机等级考试二级笔试标准化预测试卷
基础部分和 C 语言程序设计

答案及评分标准

一、选择题((1) ~ (40)题每题 1 分,(41) ~ (50)题每题 2 分,共 60 分)

- | | | | | |
|-------|-------|-------|-------|-------|
| (1)B | (2)A | (3)B | (4)A | (5)A |
| (6)B | (7)C | (8)D | (9)A | (10)D |
| (11)B | (12)D | (13)B | (14)C | (15)C |
| (16)B | (17)B | (18)A | (19)D | (20)B |
| (21)C | (22)D | (23)C | (24)B | (25)A |
| (26)D | (27)B | (28)D | (29)A | (30)C |
| (31)D | (32)C | (33)A | (34)D | (35)A |
| (36)C | (37)A | (38)C | (39)A | (40)D |
| (41)A | (42)B | (43)C | (44)D | (45)C |
| (46)B | (47)B | (48)A | (49)B | (50)C |

二、填空题(每空 2 分,共 40 分)

- (1)【1】AUTOEXEC. BAT
(2)【2】CD C:
(3)【3】外部
(4)【4】REN XYZ \ * .BAK * .FOR
(5)【5】关闭(或 Close)
(6)【6】164
(7)【7】6 【8】4 【9】2
(8)【10】2,1
(9)【11】14,15
(10)【12】3 1 - 1 3 1 - 1
(11)【13】3
(12)【14】a = 2, b = 1
(13)【15】6
(14)【16】m = 1
(15)【17】852
(16)【18】(n) 【19】scanf("%d",&n);
(17)【20】16

全国计算机等级考试二级笔试试卷

基础部分和 C 语言程序设计

2

注意事项

- 一、考生应严格遵守考场规则,得到监考人员指令后方可作答。
- 二、考生拿到试卷后应首先将自己的姓名、准考证号等内容涂写在答题卡的相应位置上。
- 三、选择题答案必须用铅笔填涂在答题卡的相应位置上,填空题的答案必须用蓝、黑色钢笔或圆珠笔写在答题卡的相应位置上,答案写在试卷上无效。
- 四、注意字迹清楚,保持卷面整洁。
- 五、考试结束将试卷和答题卡放在桌上,不得带走。待监考人员收毕清点后,方可离场。

* * * 版权所有、任何单位或个人不得保留、复制和出版,违者必究 * * *

全国计算机等级考试二级笔试标准化预测试卷

基础部分和 C 语言程序设计

(考试时间 120 分钟, 满分 100 分)

一、选择题((1)~(40)题每题 1 分,(41)~(50)题每题 2 分,共 60 分)

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

(1)通常人们所说的一个完整的计算机系统应包括

- A)主机、键盘、显示器
B)计算机及其外部设备
C)系统硬件和系统软件
D)硬件系统和软件系统

(2)下列一组数据中的最大数是

- A) $(227)_8$ B) $(1FF)_{16}$ C) $(1010001)_8$ D) $(789)_{10}$

(3)5.25 英寸软盘的写保护口已经封上时,则

- A)只能读盘、不能写盘
B)既能读盘、又能写盘
C)只能写盘、不能读盘
D)不能读盘,也不能写盘

(4)在微机中与 VGA 密切相关的设备是

- A)针式打印机 B)鼠标 C)显示器 D)键盘

(5)在微型计算机中,通用寄存器的位数是

- A)8 位 B)16 位 C)计算机字长 D)32 位

(6)用高级语言编写的源程序,要转换为与其等价的目标程序,必须经过

- A)编译 B)编辑 C)解释 D)汇编

(7)下面列出的四项中,不属于计算机病毒特征的是

- A)潜伏性 B)激发性 C)传播性 D)免疫性

(8)下列四组 DOS 命令中,功能完全等价的一组是

- A)DEL 和 ERASE B)COPY 和 XCOPY
C)DISKCOPY 和 BACKUP D)RECOVER 和 RESTORE

(9)建立一个新子目录的 DOS 命令是

- A)CREATE B)RD C)MD D)CD

(10)电子邮件是

- A)网络信息检索服务
B)通过 Web 网页发布的公告信息
C)通过网络实时交互的信息传递方式
D)一种利用网络交换信息的非交互式服务

(11)设 C 盘目录结构如图 1 所示:

C 盘根目录下文件 AUTOEXEC.BAT 的内容为:

CD FORTRAN

CD \ TC



图1 C盘目录结构

如果在 A 驱动器中插入一张根目录中只有扩展名为 .C 的若干个文件(即 A 盘中无 DOS 系统文件)的软盘片,则由 C 盘启动 DOS 系统后,第一次键入的下列命令中,正确的是

- A) COPY A: \ * .C
B) COPY A: \ * .C TC
C) COPY A: \ * .C FORTRAN
D) CD USER

(12) 下列命令中合法的是

- A) DISKCOPY A: A:
B) DISKCOPY A: C:
C) DISKCOPY C: A:
D) DISKCOPY C: D:

(13) 设当前盘中某一目录路径为 \ A \ B \ C \ D \ XU, 当前目录为 \ A。要将根目录下扩展名为 .C 的所有文件复制到当前目录下,并将扩展名改为 .FOR, 应使用命令

- A) COPY \ * .C * .FOR
B) COPY * .C \ A \ * .FOR
C) COPY * .FOR \ * .C
D) COPY \ A \ * .C * .FOR

(14) 在用 C 盘启动后,把 A 盘格式化系统盘命令为

- A) FORMAT A:
B) \ FORMAT A: /S
C) \ DOS \ FORMAT A: /S /F
D) \ DOS \ FORMAT A: \ S

(15) 设当前盘为 C 盘, C 盘当前目录为 \ DOS。把 JD1 .FOR 和 JD2 .FOR 连接起来后存入 A 盘根目录下,命名为 JD3 .FOR, 应使用命令

- A) COPY JD1 .FOR + JD2 .FOR A: JD3 .FOR
B) COPY \ F4 \ JD1 .FOR + JD2 .FOR A: \ JD3 .FOR
C) COPY \ F4 \ JD1 .FOR + \ F4 \ JD2 .FOR A: \ JD3 .FOR
D) COPY \ F4 \ JD1 .FOR + \ F4 \ JD2 .FOR JD3 .FOR

(16) DOS 系统的热启动与冷启动的不同之处为热启动时

- A) 不检测键盘、外设接口与内存
B) 不提示用户输入日期与时间
C) 不用装入 COMMAND.COM 文件
D) 不能从软盘启动

(17) 以下程序的输出结果是

```
main( )
{ int w = 5; fun(w); printf(" \ n"); }
fun (int k)
{ if (k > 0) fun(k - 1);
  printf("%d", k);
}
```

- A) 5 4 3 2 1
B) 0 1 2 3 4 5
C) 1 2 3 4 5
D) 5 4 3 2 1 0

(18) 以下程序的输出结果是

```

int d = 1;
fun(int p)
{ static int d = 5;
  d + = p;
  printf("%d ", d);
  return(d);
}
main( )
{ int a = 3; printf("%d \n", fun(a + fun(d))); }

```

A)6 9 9 B)6 6 9 C)6 15 15 D)6 6 15

(19)以下程序的输出结果是

```

main( )
{ char ch[3][4] = {"123", "456", "78"}, * p[3]; int i;
  for(i = 0; i < 3; i + + ) p[i] = ch[i];
  for(i = 0; i < 3; i + + ) printf("%s", p[i]);
}

```

A)123456780 B)123 456 780 C)12345678 D)147

(20)以下程序的输出结果是

```

#include < stdlib.h >
int a[3][3] = {1,2,3,4,5,6,7,8,9}, * p;
main( )
{ p = (int * )malloc(sizeof(int));
  f(p, a);
  printf("%d \n", * p);
}
f(int * s, int p[ ][3])
{ * s = p[1][1]; }

```

A)1 B)4 C)7 D)5

(21)以下程序的输出结果是

```

#define f(x) x * x
main( )
{ int a = 6, b = 2, c;
  c = f(a)/f(b);
  printf( "%d \n", c);
}

```

A)9 B)6 C)36 D)18

(22)以下程序的输出结果是

```

main( )
{ char * s = "12134211"; int v[4] = {0,0,0,0}, k, i;
  for (k = 0; s[k]; k + + )
  { switch (s[k])

```