

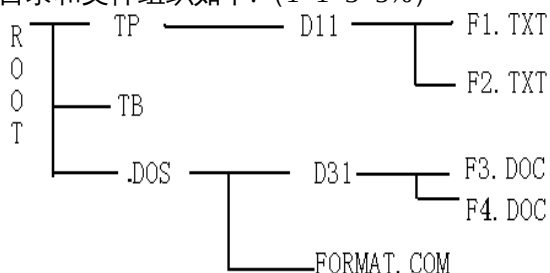
**第二届全国青少年信息学（计算机）奥林匹克分区联赛初赛试题**  
**（初中组）**

**（BASIC 语言 竞赛用时：2 小时）**

●●全部试题答案均要求写在答卷纸上，写在试卷纸上—律无效●●

一、基础知识部分：(44 分)

1. 已知 A 盘上的目录和文件组织如下：(1+1+3=5%)



其中 TP、TB、DOS、D11、D31 都是子目录名。

设当前命令提示符为 A:\TB>,请写出完成如下操作的 DOS 命令：

① 将 F1.TXT 移到 D31 子目录中去；

② 删除子目录 TB；

③ 在 DOS 运行中，没有执行过 PATH 命令，现要用 DOS 子目录中 FORMAT 命令，对插入在 B 驱动器（5.25 英寸高密）中的 360KB 软盘进行格式化工作,请写出相应的操作命令。

2. 执行命令时，屏幕上显示如下出错信息：(1+1=2%)

```
WRITE PROTECT ERROR WRITING DRIVE B
ABORT, RETRY, FALL ?
```

请说明这是什么错误？应如何校正？

3. 请用等号或不等号联接表示下列不同进位制数值的大小。(3%)

例如：(3)<sub>10</sub> < (4)<sub>4</sub> = (100)<sub>2</sub> < (A)<sub>16</sub>

其中圆括号外右下角的下标，表示圆括号内数的进位制。

(21)<sub>10</sub> (15)<sub>16</sub> (25)<sub>8</sub> (10101)<sub>2</sub>

4. 阅读下列程序段，写出程序段运行后变量 X 的值。(4%)

```
100 X1=3
110 X2=8
120 FOR =1 TO 5
130 X= (X1+X2) *2
140 X1=X2
150 X2=X
160 NEXT I
```

```
170 PRINT "X=";X
180 END
```

5. 阅读下列程序段，写出程序运行后数组元素 A1, A2, ..., A11 中的值。 (6%)

```
100 A (1) =1
110 A (2) =1
120 FOR K=2 TO 10
130 A (K+1) =1
140 FOR I=K TO 2 STEP -1
150 A (I) =A (I) +A (I-1)
160 NEXT I
170 NEXT K
```

6. 已知：ACK (M, N) 函数的计算公式如下： (4%)

$$\text{ACK}(M, N) = \begin{cases} N+1 & M=0 \\ \text{ACK}(M-1, 1) & N=0 \\ \text{ACK}(M-1, \text{ACK}(M, N-1)) & M \neq 0 \text{ 且 } N \neq 0 \end{cases}$$

请计算：ACK (1, 2) 与 ACK (2, 2) 的值。

7. 请写出对应计算如下算式的程序段： (4%)

$$Y = A_N X^N + A_{N-1} X^{N-1} + \dots + A_1 X + A_0$$

8. 有 N×N 个数据组成如下方阵： (7%)

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} & \dots & A_{1N} \\ A_{21} & A_{22} & A_{23} & \dots & A_{2N} \\ A_{31} & A_{32} & A_{33} & \dots & A_{3N} \\ \dots & \dots & \dots & \dots & \dots \\ A_{N1} & A_{N2} & A_{N3} & \dots & A_{NN} \end{pmatrix}$$

并已知： $A_{ij} = A_{ji}$  现将  $A_{11}$ ,  $A_{21}$ ,  $A_{22}$ ,  $A_{31}$ ,  $A_{32}$ ,  $A_{33}$  ... 存储在一维数组 A (1), A (2), ..., A ((N\*(N+1))/2) 中。

试问：任给 i, j 怎样求出 K 来，使得 A[K] 的值正好是  $A_{ij}$ ，请写出由 i, j 计算 K 值的表达式。

9. 已知： $A_1, A_2, \dots, A_{81}$  共有 81 个数，其中只有一个数比其它数大，要用最少的比较运算次数，把这个值大的数找出来（假设两个数比较一次能决定出大于、小于或等于这三种情况）请将以下算法补充完整： (9%)

第一步： $S1 = A_1 + A_2 + \dots + A_{27}$

$S2 = A_{28} + A_{29} + \dots + A_{54}$

第一次比较 (S1, S2) :

$S1 > S2$  取  $K=0$

$S1 < S2$  取  $K=27$

$S1 = S2$  取  $K=54$

第二步：  $S1 = A_{K+1} + A_{K+2} + \dots + A_{K+9}$

$S2 = A_{K+10} + A_{K+11} + \dots + A_{K+18}$

第二次比较 ( $S1, S2$ )：

$S1 > S2$  取  $K = \underline{\hspace{1cm}}$

$S1 < S2$  取  $K = \underline{\hspace{1cm}}$

$S1 = S2$  取  $K = \underline{\hspace{1cm}}$

第三步：  $S1 = A_{K+1} + A_{K+2} + A_{K+3}$

$S2 = A_{K+4} + A_{K+5} + A_{K+6}$

第三次比较 ( $S1, S2$ )：

$S1 > S2$  取  $K = \underline{\hspace{1cm}}$

$S1 < S2$  取  $K = \underline{\hspace{1cm}}$

$S1 = S2$  取  $K = \underline{\hspace{1cm}}$

第四步：  $S1 = A_{K+1}$

$S2 = A_{K+2}$

第四次比较 ( $S1, S2$ )：

$S1 > S2$          为最大数

$S1 < S2$          为最大数，

$S1 = S2$          为最大数。

二、根据题目要求，补充完善以下程序：(56%)

1. [题 目] 15分 (每空5分)

设有  $N$  个不同整数的数列：例如  $N=4$  时，有 4 个不同整数的数列为 17，4，16，5。数列中的第 1 个数 17，比它后面的三个数都大，则称数 17 的逆数为 3。数列中的第 2 个数 4 比它后面的数都小，则称数 4 的逆数为 0。同时记数列中全部逆数的和称为数列的逆数。上例中，数列 17，4，16，5 的逆数：为  $3+0+1+0=4$ 。

**[程序要求]** 当给出  $N$  个不同整数的数列后，求出此数列的逆数。

**[算法描述]** 为求得上面问题的解，设置数组  $A$  和逆数计数器  $S$ ，然后用一个二重循环求出数列的逆数。

**[程 序]** 10 DIM A (10)

20 N=10

30 FOR I=1 TO N : INPUT A (I) : NEXT I

40 S=0

50 FOR I=1 TO ①

60 FOR J= ② TO N

70 IF A (I) > A (J) THEN ③

80 NEXT J, I

90 PRINT "S=" ; S

100 END

2. [题 目] 20分 (每空4分)

装球：设有  $n$  个盒子 ( $n$  足够大，可装入任何数量的球)，分别编号 1，2，……。同时有  $k$  个小球 ( $k>0$ )，今将  $k$  个小球装入到盒子中去。装入规则如下：

(1) 第一个盒子不能为空。

- (2) 装入必须严格按递增顺序进行。
- 例如，当  $k=8$ ， $n=6$  时，装入方法有：1, 2, 5 或 1, 3, 4
- (3) 在满足上面的两个条件下，要求有球的盒子尽可能多。
- (4) 装完后，相邻盒子中球个数差的绝对值之和最小（未装的盒子不计）。

如上例中：

装入法 1, 2, 5，则差的绝对值之和为  $2-1+5-2=4$

装入法 1, 3, 4，则差的绝对值之和为  $3-1+4-3=3$

**[程序要求]** 给出  $k$  ( $k$  表示小球的个数) 之后，求出满足上述四个条件的装入方法。

**[算法描述]** 设计一个数组  $A$  用数组元素代表盒子，然后依次装入小球。

**[程 序]**

```

5 N=20
10 DIM A (N)
20 INPUT K
30 ①
40 J=1
50 IF ② THEN 80
60 A (J) =J : ③ : J=J+1
70 GOTO 50
80 L=J-1
90 IF K=0 THEN 120
100 ④ : K=K-1 : L=L-1
110 GOTO 90
120 FOR I=1 TO ⑤
130 PRINT A (I) ;
140 NEXT I
150 END

```

### 3. [题 目] 21 分 (3+4+3+3+4+4)

积木游戏：设有  $n$  个小木块排成一排，如下图：

□□□……□

游戏开始时，每个小木块向下的一面涂有红、黄、蓝三种颜色之中的一种（约定：0 表示红色，1 表示黄色，2 表示蓝色）。要求通过翻看与交换方式对小木块重新排列（翻看的规则为每个小木块只能看一次），最终成为下面的形状：

□□□……□    □□……□    □□……□  
           红                蓝                黄

即相同颜色的木块排列在一起，设计一个翻看与交换的方案，使得用最少的交换次数实现上面的要求。

**[算法描述]**：翻看小木块时，可以从两端进行。例如，设中间状态如下：

□□□……□ A□□……B □□□……C □□□……□  
           红                未翻过                蓝                黄

此时，可以从两个方向看，即从 A 或 B 处开始：

- (1) 若看 A 则有三种可能性：
- 为红色，则不用交换
  - 为蓝色，交换一次，即 A 与 B 交换

为黄色，交换两次，即 C 与 B 交换一次，然后 A 与 C 交换一次  
此时，平均交换次数为 1。

(2) 若看 B，也有三种可能性：

为蓝色，则不用交换

为红色，交换一次，即 B 与 A 交换一次。

为黄色，交换一次，即 B 与 C 交换一次。

此时，平均交换次数为 2/3。

由此可见，从 B 处翻看直到游戏结束，次数最少符合题目要求。

#### [程 序]

```
5 INPUT "N=" ; N
10 DIM A (N)
20 FOR I=1 TO N
30 INPUT A (I)
40 NEXT I
50 R=1 : ① : Y=N
60 IF ② THEN 160
70 IF ③ THEN 140
80 IF ④ THEN 110
90 TEM=A (R) : A (R) =A (B) : A (B) =TEM : R=R+1
100 GOTO 60
110 TEM=A (B) : A (B) =A (Y) : A (Y) =TEM
120 ⑤ : ⑥
130 GOTO 660
140 B=B-1
150 GOTO 60
160 FOR I=1 TO N : PRINT A (I) ; " " ; : NEXT I
170 END
```