

# 第十二届全国青少年信息学奥林匹克联赛初赛试题

## ( 提高组 Pascal 语言 二小时完成 )

●● 全部试题答案均要求写在答卷纸上, 写在试卷纸上一律无效 ●●

### 一、单项选择题 (共 10 题, 每题 1.5 分, 共计 15 分。每题有且仅有一个正确答案。)

1. 在以下各项中。( ) 不是 CPU 的组成部分。

- A. 控制器      B. 运算器      C. 寄存器      D. ALU      E. RAM

2. BIOS (基本输入输出系统) 是一组固化在计算机内 ( ) 上一个 ROM 芯片上的程序。

- A. 控制器      B. CPU      C. 主板      D. 内存条      E. 硬盘

3. 在下面各世界顶级的奖项中, 为计算机科学与技术领域作出杰出贡献的科学家设立的奖项是 ( )。

- A. 沃尔夫奖      B. 诺贝尔奖      C. 菲尔兹奖  
D. 图灵奖      E. 南丁格尔奖

4. 在编程时 (使用任一种高级语言, 不一定是 Pascal), 如果要从磁盘文件中输入一个很大的二维数组 (例如  $1000 \times 1000$  的 double 型数组), 按行读 (即外层循环是关于行的) 与按列读 (即外层循环是关于列的) 相比, 在输入效率上 ( )。

- A. 没有区别      B. 有一些区别, 但机器处理速度很快, 可忽略不计  
C. 按行读的方式要高一些      D. 按列读的方式要高一些      E. 取决于数组的存储方式。

5. 在 Pascal 语言中, 表达式  $(21 \text{ xor } 2)$  的值是 ( )

- A. 441      B. 42      C. 23      D. 24      E. 25

6. 在 Pascal 语言中, 判断 a 不等于 0 且 b 不等于 0 的正确的条件表达式是 ( )

- A.  $\text{not } a=0 \text{ or } \text{not } b=0$       B.  $\text{not}((a=0) \text{ and } (b=0))$       C.  $\text{not}(a=0 \text{ and } b=0)$   
D.  $(a <> 0) \text{ or } (b <> 0)$       E.  $(a <> 0) \text{ and } (b <> 0)$

7. 某个车站呈狭长形, 宽度只能容下一台车, 并且只有一个出入口。已知某时刻该车站状态为空, 从这一时刻开始的出入记录为: “进, 出, 进, 进, 进, 出, 出, 进, 进, 进, 出, 出”。假设车辆入站的顺序为 1, 2, 3, …, 则车辆出站的顺序为 ( )。

- A. 1, 2, 3, 4, 5      B. 1, 2, 4, 5, 7      C. 1, 4, 3, 7, 6  
D. 1, 4, 3, 7, 2      E. 1, 4, 3, 7, 5

8. 高度为 n 的均衡的二叉树是指: 如果去掉叶结点及相应的树枝, 它应该是高度为 n-1 的满二叉树。

在这里，树高等于叶结点的最大深度，根结点的深度为 0，如果某个均衡的二叉树共有 2381 个结点，则该树的树高为（ ）。

- A. 10      B. 11      C. 12      D. 13      E.  $2^{10} - 1$

9. 与十进制数 1770.625 对应的八进制数是（ ）。

- A. 3352.5      B. 3350.5      C. 3352.1161  
D. 3350.1151      E. 前 4 个答案都不对

10. 将 5 个数的序列排序，不论原先的顺序如何，最少都可以通过（ ）次比较，完成从小到大的排序。

- A. 6      B. 7      C. 8      D. 9      E. 10

二、不定项选择题（共 10 题，每题 1.5 分，共计 15 分。每题正确答案的个数大于或等于 1。多选或少选均不得分）。

11. 设  $A=B=D=\text{true}$ ,  $C=E=\text{false}$ ，以下逻辑运算表达式值为真的有（ ）。

- A.  $(\neg A \wedge B) \vee (C \wedge D) \vee \neg E$       B.  $\neg(((A \wedge B) \vee C) \wedge D \wedge E)$   
C.  $A \wedge (B \vee C \vee D \vee E)$       D.  $(A \wedge (B \vee C)) \wedge D \wedge E$

12.  $(2010)_{16} + (32)_8$  的结果是（ ）。

- A.  $(8234)_{10}$       B.  $(202A)_{16}$   
C.  $(100000000110)_2$       D.  $(2042)_{16}$

13. 设栈 S 的初始状态为空，元素 a, b, c, d, e 依次入栈，以下出栈序列不可能出现的有（ ）。

- A. a, b, c, e, d      B. b, c, a, e, d  
C. a, e, c, b, d      D. d, c, e, b, a

14. 已知 6 个结点的二叉树的先根遍历是 1 2 3 4 5 6（数字为结点的编号，以下同），后根遍历是 3 2 5 6 4 1，则该二叉树的可能的中根遍历是（ ）

- A. 3 2 1 4 6 5      B. 3 2 1 5 4 6  
C. 2 3 1 5 4 6      D. 2 3 1 4 6 5

15. 在下列各数据库系统软件中，以关系型数据库为主体结构的是（ ）。

- A. ACCESS      B. SQL Server  
C. Oracle      D. Foxpro

16. 在下列各软件中，属于 NOIP 竞赛（复赛）推荐使用的语言环境有（ ）。

- A. gcc/g++      B. Turbo Pascal  
C. Turbo C      D. free pascal

17. 以下断电之后将不能保存数据的有（ ）。

- A. 硬盘      B. ROM      C. 显存      D. RAM

18. 在下列关于计算机语言的说法中，正确的有（ ）。

- A. Pascal和C都是编译执行的高级语言  
B. 高级语言程序比汇编语言程序更容易从一种计算机移植到另一种计算机上  
C. C++是历史上的第一个支持面向对象的计算机语言  
D. 高级语言比汇编语言更高级，是因为它的程序的运行效率更高

19. 在下列关于计算机算法的说法中，正确的有（ ）。

- A. 一个正确的算法至少要有有一个输入  
B. 算法的改进，在很大程度上推动了计算机科学与技术的进步  
C. 判断一个算法的好坏，主要依据它在某台计算机上具体实现时的运行时间  
D. 目前仍然存在许多涉及到国计民生的重大课题，还没有找到能够在计算机上实施的有效算法

20. 在下列关于青少年信息学竞赛的说法中，你赞成的是（ ）（本题不回答为0分，答题一律满分）。

- A. 举行信息学竞赛的目的，是为了带动广大青少年学科学、爱科学，为造就一大批优秀的计算机科学与技术人才奠定良好的基础  
B. 如果竞赛优胜者不能直接保送上大学，我今后就不再参与这项活动了  
C. 准备竞赛无非要靠题海战术，为了取得好成绩，就得拼时间、拼体力  
D. 为了取得好成绩，不光要看智力因素，还要看非智力因素。优秀选手应该有坚韧不拔的意志，有严谨求实的作风，既要努力奋进，又要胜不骄败不馁

### 三. 问题求解（共 2 题，每题 5 分，共计 10 分）

1. 将 2006 个人分成若干不相交的子集，每个子集至少有 3 个人，并且：

- (1) 在每个子集中，没有人认识该子集的所有人。  
(2) 同一子集的任何 3 个人中，至少有 2 个人互不认识。  
(3) 对同一子集中任何 2 个不相识的人，在该子集中恰好只有 1 个人认识这两个人。

则满足上述条件的子集最多能有\_\_\_\_\_个？

2. 将边长为  $n$  的正三角形每边  $n$  等分，过每个分点分别做另外两边的平行线，得到若干个正三角形，我们称为小三角形。正三角形的一条通路是一条连续的折线，起点是最上面的一个小三角形，终点是最下面一行位于中间的小三角形。在通路中，只允许由一个小三角形走到另一个与其有公共边的且位于同一行或下一行的小三角形，并且每个小三角形不能经过两次或两次以上（图中是  $n=5$  时一条通路的例子）。设  $n=10$ ，则该正三角形的不同的通路的总数为\_\_\_\_\_。



```

for i:=0 to 4 do
begin
    t:=1;
    for j:=1 to m[i]-1 do t:=t*2;
    t:=(t*2-1)*t;
    write (t,' ');
end;
writeln;
end.

```

输出:\_\_\_\_\_

### 3. Program ex403;

```

Const  NN=7;
Type
    Arr1=array[0..30] of char;
var
    s:arr1;
    k,p:integer;
function fun1(s:arr1; a:char;n:integer):integer;
var
    j:integer;
begin
    j:=n;
    while (a<s[j])and(j>0) do dec(j);
    fun1:=j;
end;
Function fun2(s:arr1; a:char; n:integer):integer;
var
    j:integer;
begin
    j:=1;
    while (a>s[j])and(j<n) do inc(j);
    fun2:=j;
end;
begin
    for k:=1 to NN do
        s[k]:=chr(ord('A')+2*k+1);
    k:=fun1(s,'M',NN)+fun2(s,'M',NN);

```

```
writeln(k);
end.
输出: _____
```

```
4. program ex404;
var
  x,x2:longint;
procedure digit(n,m:longint);
var n2:integer;
begin
  if(m>0) then
  begin
    n2:=n mod 10;
    write(n2:2);
    if(m>1) then digit(n div 10,m div 10);
    n2:=n mod 10;
    write(n2:2);
  end;
end;
begin
  writeln('Input a number:');
  readln(x);
  x2:=1;
  while(x2<x) do x2:=x2*10;
  x2:=x2 div 10;
  digit(x,x2);
  writeln;
end.
输入: 9734526
输出: _____
```

## 五. 完善程序 (前 5 空, 每空 2 分, 后 6 空, 每空 3 分, 共 28 分)

1. (选排列) 下面程序的功能是利用递归方法生成从 1 到  $n$  ( $n < 10$ ) 的  $n$  个数中取  $k$  ( $1 \leq k \leq n$ ) 个数的全部可能的排列 (不一定按升序输出)。例如, 当  $n=3$ ,  $k=2$  时, 应该输出 (每行输出 5 个排列):

```
12 13 21 23 32
31
```

程序:

```

Program ex501;
Var
  i,n,k:integer;
  a:array[1..10] of integer;
  count:longint;
Procedure perm2(j:integer);
  var i,p,t:integer;
  begin
    if ① then
      begin
        for i:=k to n do
          begin
            inc(count);
            t:=a[k]; a[k]:=a[i]; a[i]:=t;
            for ② do write(a[p]:1);
            write(' ');
            t:=a[k];a[k]:=a[i];a[i]:=t;
            if (count mod 5=0) then writeln;
          end;
        exit;
      end;
    for i:=j to n do
      begin
        t:=a[j];a[j]:=a[i];a[i]:=t;
        ③ ;
        t:=a[j]; ④ ;
      end;
    end;
  begin
    writeln('Entry n,k (k<=n):'); read(n,k);
    count:=0;
    for i:=1 to n do a[i]:=i;
    ⑤ ;
  end.

```

**2. (TSP 问题的交叉算子)** TSP 问题(Traveling Salesman Problem)描述如下: 给定  $n$  个城市, 构成一个完全图, 任何两城市之间都有一个代价 (例如路程、旅费等), 现要构造遍历所有城市的环路, 每个城市恰好经过一次, 求使总代价达到最小的一条环路。

遗传算法是求解该问题的一个很有效的近似算法。在该算法中，一个个体为一条环路，其编码方法之一是 1 到 n 这 n 个数字的一个排列，每个数字为一个城市的编号。例如当 n=5 时，“3 4 2 1 5”表示该方案实施的路线为 3→4→2→1→5→3。遗传算法的核心是通过两个个体的交叉操作，产生两个新的个体。下面的程序给出了最简单的一种交叉算法。具体过程如下：

- (1) 选定中间一段作为互换段，该段的起止下标为 t1, t2，随机生成 t1, t2 后，互换两段。
- (2) 互换后，在每个新的排列中可能有重复数字，因而不能作为新个体的编码，一般再做两步处理：
  - (2.1) 将两个互换段中，共同的数字标记为 0，表示已处理完。
  - (2.2) 将两个互换段中其余数字标记为 1，按顺序将互换段外重复的数字进行替换。

例如：n=12，两个个体分别是：

a1: 1 3 5 4 \* 2 6 7 9 \* 10 12 8 11

a2: 3 2 1 12 \* 6 7 10 11 \* 8 5 4 9

t1=5, t2=8。上述每一行中，两个星号间的部分为互换段。假定数组的下标从 1 开始，互换后有：

a1: 1 3 5 4 \* 6 7 10 11 \* 10 12 8 11

a2: 3 2 1 12 \* 2 6 7 9 \* 8 5 4 9

然后，将数字 6,7 对应的项标记为 0，星号内数字 2,9,10,11 对应的项标记为 1，并且按顺序对应关系为：10↔2, 11↔9。于是，将 a1[9]=10 替换为 a1[9]=2，将 a2[2]=2 替换为 a2[2]=10，类似再做第 2 组替换。这样处理后，就得到了两个新个体：

a1: 1 3 5 4 6 7 10 11 2 12 8 9

a2: 3 10 1 12 2 6 7 9 8 5 4 11

- (3) 输出两个新个体的编码。

程序：

```
program ex502;
type arr1=array[1..20] of integer;
var
  a1,a2,kz1,kz2:arr1;
  n,k,t1,t2:integer;
function rand1(k:integer):integer;
var t:integer;
begin
  t:=0;
  while (t<2) or(t>k) do
    t:=random(k+1)-2;
  rand1:=t;
end;
procedure read1(var a:arr1;m:integer);
{读入数组元素 a[1]至 a[m]，a[0]=0，略。}
procedure wrt1(var a:arr1;m:integer);
{输出数组元素 a[1]至 a[m]，略。}
```



```

procedure cross(var a1,a2:arr1;t1, t2,n:integer);
var
  i,j,t,kj:integer;
begin
  for i:=t1 to t2 do
    begin
      t:=a1[i]; ① ;
    end;
    for i:=1 to n do
      if (i<t1)or(i>t2) then
        begin
          kz1[i]:=-1;kz2[i]:=-1;
        end
      else
        begin ② ; end;
    for i:=t1 to t2 do
      for j:=t1 to t2 do
        if(a1[i]=a2[j]) then
          begin ③ ; break; end;
    for i:=t1 to t2 do
      if(kz1[i]=1) then
        begin
          for j:=t1 to t2 do
            if(kz2[j]=1) then
              begin kj:=j; break; end;
          for j:=1 to n do
            if ④ then
              begin a1[j]:=a2[kj];break; end;
          for j:=1 to n do
            if ⑤ then
              begin a2[j]:=a1[i]; break; end;
              kz1[i]:=0;kz2[kj]:=0;
            end;
          end;
        begin
          writeln('input (n>5):');
          readln(n);
          writeln('input array 1:'); readl(a1,n);

```

```

writeln('input array 2:'); readl(a2,n);
t1:=randl(n-1);
repeat
    t2:=randl(n-1);
until(t1<>t2);
if(t1>t2) then
    begin k:=t1; t1:=t2; t2:=k; end;
    ⑥;
writeln(a1,n); writeln(a2,n);
end.

```