

# 第十二届全国青少年信息学奥林匹克联赛初赛试题

（普及组 C++ 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 20 题，每题 1.5 分，共计 30 分。每题有且仅有一个正确答案。）。

1. 在下面各世界顶级的奖项中，为计算机科学与技术领域做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

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

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

3. 以下断电之后仍能保存数据的有（ ）。

- A. 寄存器      B. ROM      C. RAM      D. 高速缓存

4. Linux 是一种（ ）。

- A. 绘图软件      B. 程序设计语言      C. 操作系统      D. 网络浏览器

5. CPU 是（ ）的简称。

- A. 硬盘      B. 中央处理器      C. 高级程序语言      D. 核心寄存器

6. 在计算机中，防火墙的作用是（ ）。

- A. 防止火灾蔓延      B. 防止网络攻击  
C. 防止计算机死机      D. 防止使用者误删除数据

7. 在下列关于计算机语言的说法中，不正确的是（ ）。

- 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. 与汇编语言相比，高级语言程序更容易阅读

8. 在下列关于计算机算法的说法中，不正确的是（ ）。

- 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的主要标准是算法的时间复杂性与空间复杂性

- 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
9. 在下列各种排序算法中，不是以“比较”作为主要操作的算法是（ ）。
- A. 选择排序      B. 冒泡排序      C. 插入排序      D. 基数排序
10. 在编程时（使用任一种高级语言，不一定是 C++），如果需要从磁盘文件中输入一个很大的二维数组（例如 1000\*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. 按行读的方式要高一些  
C. 按列读的方式要高一些      D. 取决于数组的存储方式。
11. 在 C++ 中，表达式  $21^2$  的值是（ ）
- A. 441      B. 42      C. 23      D. 24
12. 在 C++ 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.  $!a==0 \ || \ !b==0$       B.  $!((a==0)\&\&(b==0))$   
C.  $!(a==0\&\&b==0)$       D.  $a \ \&\& \ b$
13. 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
14. 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
15. 与十进制数 1770 对应的八进制数是（ ）。
- A. 3350      B. 3351      C. 3352      D. 3540
16. 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
17. 设  $A=B=D=true$ ,  $C=false$ ，以下逻辑运算表达式值为真的有（ ）。
- A.  $(\neg A \wedge B) \vee (C \wedge D)$       B.  $\neg((A \vee B \vee D) \wedge C)$

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

18.  $(2010)_{16} + (32)_8$ 的结果是 ( )。

- A.  $(8234)_{10}$       B.  $(202B)_{16}$   
C.  $(20056)_8$       D.  $(100000000110)_2$

19. 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

20. 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 1 3 5 4 6      D. 2 3 1 4 6 5

## 二. 问题求解 (共 2 题, 每题 5 分, 共计 10 分)

1. (寻找假币) 现有 80 枚硬币, 其中有一枚是假币, 其重量稍轻, 所有真币的重量都相同, 如果使用不带砝码的天平称重, 最少需要称几次, 就可以找出假币? 你还要指出第 1 次的称重方法。请写出你的结果: \_\_\_\_\_。

2. (取石子游戏) 现有 5 堆石子, 石子数依次为 3, 5, 7, 19, 50, 甲乙两人轮流从任一堆中任取 (每次只能取自一堆, 不能不取), 取最后一颗石子的一方获胜。甲先取, 问甲有没有获胜策略 (即无论乙怎样取, 甲只要不失误, 都能获胜)? 如果有, 甲第一步应该在哪一堆里取多少? 请写出你的结果: \_\_\_\_\_。

## 三. 阅读程序写结果 (共 4 题, 每题 8 分, 共计 32 分)

```
1. #include <iostream.h>
void main()
{int i,u[4],a,b,x,y=10;
  for(i=0;i<=3;i++)
    cin >>u[i];
  a=(u[0]+u[1]+u[2]+u[3])/7;
  b=u[0]/((u[1]-u[2])/u[3]);
  x=(u[0]+a+2)-u[(u[3]+3)%4];
  if(x>10)
    y+= (b*100-u[3])/(u[u[0]%3]*5);
```

```

else
    y+=20+(b*100-u[3])/(u[u[0]%3]*5);
    cout <<x<<","<<y<<endl;
} // 注：本例中，给定的输入数据可以避免分母为 0 或下标越界。
输入：9 3 9 4
输出：_____

```

```

2. #include <iostream.h>

void main()
{int i,j,m[]={2,3,5,7,13};
long t;
for (i=0;i<=4;i++)
    {t=1;
    for(j=1;j<m[i];j++)    t*=2;
    cout <<(t*2-1)*t<<"    ";
    }
cout <<endl;
}
输出：_____

```

```

3. #include "iostream.h"
#define N 7
int fun(char s[],char a,int n)
{int j;
j=n;
while(a<s[j] && j>0) j--;
return j;
}

void main()
{char s[N+1];
int k;
for(k=1;k<=N;k++)
    s[k]='A'+2*k+1;
cout <<fun(s,'M',N)<<endl;
}
输出：_____

```

```

4. #include <iostream.h>

```

```

#include <iomanip.h>
void digit(long n,long m)
{if(m>0)
    cout <<setw(2)<<n%10;
    if(m>1)
        digit(n/10,m/10);
    cout <<setw(2)<<n%10;
}
void main()
{long x,x2;
    cout <<"Input a number:"<<endl;
    cin >>x;
    x2=1;
    while(x2<x)  x2*=10;
    x2/=10;
    digit(x,x2);
    cout <<endl;
}

```

输入: 9734526

输出: \_\_\_\_\_

#### 四. 完善程序 (前 4 空, 每空 2.5 分, 后 6 空, 每空 3 分, 共 28 分)

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

```

123 132 213 231 321
312

```

程序:

```

#include <iostream.h>
#include <iomanip.h>
int n,a[10]; // a[1],a[2],...,a[n]构成n个数的一个排列
long count=0; // 变量count 记录不同排列的个数, 这里用于控制换行
void perm(int k)
{int j,p,t;
    if( ① )
    {count++;
        for(p=1;p<=n;p++)
            cout <<setw(1)<<a[p];
    }
}

```

```

        cout <<" ";
        if( ② ) cout <<endl;
        return;
    }
    for(j=k;j<=n;j++)
        {t=a[k];a[k]=a[j];a[j]=t;
          ③ ;
          t=a[k]; ④ ;
        }
    }
}

void main()
{int i;
  cout <<"Entry n:"<<endl;
  cin >>n;
  for(i=1;i<=n;i++) a[i]=i;
  ⑤ ;
}

```

2. 由键盘输入一个奇数  $P$  ( $P < 100,000,000$ ), 其个位数字不是 5, 求一个整数  $S$ , 使  $P \times S = 1111\dots 1$  (在给定的条件下, 解  $S$  必存在)。要求在屏幕上依次输出以下结果:

(1)  $S$  的全部数字。除最后一行外, 每行输出 50 位数字。 (2) 乘积的数字位数。

例 1: 输入  $p=13$ , 由于  $13 \times 8547 = 111111$ , 则应输出 (1) 8547, (2) 6

例 2: 输入  $p=147$ , 则输出结果应为 (1) 755857898715041572184429327286470143613

(2) 42, 即等式的右端有 42 个 1。

程序:

```

#include <iostream.h>
#include <iomanip.h>
void main()
{long p,a,b,c,t,n;
  while (1)
  { cout <<"输入 p, 最后一位为 1 或 3 或 7 或 9:"<<endl;
    cin >>p;
    if ((p%2!=0)&&(p%5!=0)) // 如果输入的数符合要求, 结束循环
      ⑥ ;
    }
  a=0; n=0;
  while (a<p)

```

```

{a=a*10+1; n++; // 变量 a 存放部分右端项, n 为右端项的位数
}
t=0;
do
{b=a/p;
cout <<setw(1)<<b;
t++;
if ( ⑦ )
    cout <<endl;
c= ⑧ ; a= ⑨ ; n++;
} while (c>0);
cout<<endl<<"n="<< ⑩ <<endl;
}

```