

第十五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、关于图灵机下面的说法哪个是正确的：
A) 图灵机是世界上最早的电子计算机。
B) 由于大量使用磁带操作，图灵机运行速度很慢。
C) 图灵机是英国人图灵发明的，在二战中为破译德军的密码发挥了重要作用。
D) 图灵机只是一个理论上的计算模型。
- 2、关于计算机内存下面的说法哪个是正确的：
A) 随机存储器 (RAM) 的意思是当程序运行时，每次具体分配给程序的内存位置是随机而不确定的。
B) 1MB 内存通常是指 1024*1024 字节大小的内存。
C) 计算机内存严格说来包括主存 (memory)、高速缓存 (cache) 和寄存器 (register) 三个部分。
D) 一般内存中的数据即使在断电的情况下也能保留 2 个小时以上。
- 3、关于 BIOS 下面说法哪个是正确的：
A) BIOS 是计算机基本输入输出系统软件的简称。
B) BIOS 里包含了键盘、鼠标、声卡、显卡、打印机等常用输入输出设备的驱动程序。
C) BIOS 一般由操作系统厂商来开发完成。
D) BIOS 能提供各种文件拷贝、复制、删除以及目录维护等文件管理功能。
- 4、关于 CPU 下面哪个说法是正确的：
A) CPU 全称为中央处理器 (或中央处理单元)。
B) CPU 可以直接运行汇编语言。
C) 同样主频下，32 位的 CPU 比 16 位的 CPU 运行速度快一倍。
D) CPU 最早是由 Intel 公司发明的。
- 5、关于 ASCII，下面哪个说法是正确的：
A) ASCII 码就是键盘上所有键的唯一编码。
B) 一个 ASCII 码使用一个字节的内存空间就能够存放。
C) 最新扩展的 ASCII 编码方案包含了汉字和其他欧洲语言的编码。
D) ASCII 码是英国人主持制定并推广使用的。
- 6、下列软件中不是计算机操作系统的是：
A) Windows B) Linux C) OS/2 D) WPS
- 7、关于互联网，下面的说法哪一个是正确的：

- A) 新一代互联网使用的 IPv6 标准是 IPv5 标准的升级与补充。
B) 互联网的入网主机如果有了域名就不再需要 IP 地址。
C) 互联网的基础协议为 TCP/IP 协议。
D) 互联网上所有可下载的软件及数据资源都是可以合法免费使用的。
- 8、关于 HTML 下面哪种说法是正确的：
A) HTML 实现了文本、图形、声音乃至视频信息的统一编码。
B) HTML 全称为超文本标记语言。
C) 网上广泛使用的 Flash 动画都是由 HTML 编写的。
D) HTML 也是一种高级程序设计语言。
- 9、关于程序设计语言，下面哪个说法是正确的：
A) 加了注释的程序一般会比同样的没有加注释的程序运行速度慢。
B) 高级语言开发的程序不能使用在低层次的硬件系统如：自控机床或低端手机上。
C) 高级语言相对于低级语言更容易实现跨平台的移植。
D) 以上说法都不对。
- 10、已知大写字母A的ASCII编码为65（10进制），则大写字母J的10进制ASCII编码为：
A) 71 B) 72 C) 73 D) 以上都不是
- 11、十进制小数125.125对应的8进制数是
A) 100.1 B) 175.175 C) 175.1 D) 100.175
- 12、有六个元素 FEDCBA 从左至右依次顺序进栈，在进栈过程中会有元素被弹出栈。问下列哪一个**不可能**是合法的出栈序列？
A) EDCFAB B) DECABF C) CDFEBA D) BCDAEF
- 13、表达式 $a*(b+c)-d$ 的后缀表达式是：
A) $abcd*+-$ B) $abc+*d-$ C) $abc*+d-$ D) $-+*abcd$
- 14、一个包含 n 个分支结点（非叶结点）的非空二叉树，它的叶结点数最多为：
A) $2n+1$ B) $2n-1$ C) $n-1$ D) $n+1$
- 15、快速排序最坏情况下的算法时间复杂度为：
A) $O(\log_2 n)$ B) $O(n)$ C) $O(n \log_2 n)$ D) $O(n^2)$
16. 有一个由 4000 个整数构成的顺序表，假定表中的元素已经按升序排列，采用二分查找定位一个元素。则最多需要几次比较就能确定是否存在所查找的元素：
A) 11 次 B) 12 次 C) 13 次 D) 14 次
- 17、排序算法是稳定的意思是关键码相同的记录排序前后相对位置不发生改变，下列哪种排序算法是不稳定的：
A) 冒泡排序 B) 插入排序 C) 归并排序 D) 快速排序

- 18、已知 n 个顶点的有向图，若该图是强连通的（从所有顶点都存在路径到达其他顶点），则该图中最少有多少条有向边？
A) n B) $n+1$ C) $n-1$ D) $n*(n-1)$
- 19、全国信息学奥林匹克的官方网站为参与信息学竞赛的老师同学们提供相关的信息和资源，请问全国信息学奥林匹克官方网站的网址是：
A) <http://www.noi.com/> B) <http://www.noi.org/>
C) <http://www.noi.cn/> D) <http://www.xinxixue.com/>
- 20、在参加 NOI 系列竞赛过程中，下面哪一种行为是 不 被严格禁止的：
A) 携带书写工具，手表和不具有通讯功能的电子词典进入赛场。
B) 在联机测试中通过手工计算出可能的答案并在程序里直接输出答案来获取分数。
C) 通过互联网搜索取得解题思路。
D) 在提交的程序中启动多个进程以提高程序的执行效率。

二. 问题求解（共 2 题，每空 5 分，共计 10 分）

1. 小陈现有 2 个任务 A, B 要完成，每个任务分别有若干步骤如下：A=a1->a2->a3, B=b1->b2->b3->b4->b5。在任何时候，小陈只能专心做某个任务的一个步骤。但是如果愿意，他可以在做完手中任务的当前步骤后，切换至另一个任务，从上次此任务第一个未做的步骤继续。每个任务的步骤顺序不能打乱，例如……a2->b2->a3->b3……是合法的，而……a2->b3->a3->b2……是不合法的。小陈从 B 任务的 b1 步骤开始做，当恰做完某个任务的某个步骤后，就停工回家吃饭了。当他回来时，只记得自己已经完成了整个任务 A，其他的都忘了。试计算小陈饭前已做的可能的任务步骤序列共有_____种。

2. 有如下的一段程序：

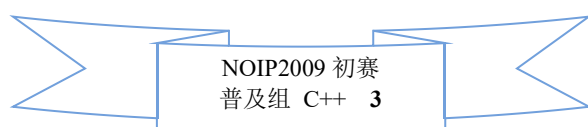
```
1.  a=1;
2.  b=a;
3.  d=-a;
4.  e=a+d;
5.  c=2*d;
6.  f=b+e-d;
7.  g=a*f+c;
```

现在要把这段程序分配到若干台（数量充足）用电缆连接的 PC 上做并行执行。每台 PC 执行其中的某几个语句，并可随时通过电缆与其他 PC 通讯，交换一些中间结果。假设每台 PC 每单位时间可以执行一个语句，且通讯花费的时间不计。则这段程序最快可以在_____单位时间内执行完毕。注意：任意中间结果只有在某台 PC 上已经得到，才可以被其他 PC 引用。例如若语句 4 和 6 被分别分配到两台 PC 上执行，则因为语句 6 需要引用语句 4 的计算结果，语句 6 必须在语句 4 之后执行。

三. 阅读程序写结果（共 4 题，每题 8 分，共计 32 分）

1.

```
#include <iostream>
```



```

using namespace std;

int a,b;

int work(int a,int b){
    if (a%b)
        return work(b,a%b);
    return b;
}

int main(){
    cin >> a >> b;
    cout << work(a,b) << endl;
    return 0;
}

```

输入: 20 12
输出: _____

2.

```

#include <iostream>
using namespace std;
int main()
{
    int a[3],b[3];
    int i,j,tmp;
    for (i=0;i<3;i++)
        cin >> b[i];
    for (i=0;i<3;i++)
    {
        a[i]=0;
        for (j=0;j<=i;j++)
        {
            a[i]+=b[j];
            b[a[i]%3]+=a[j];
        }
    }
    tmp=1;
    for (i=0;i<3;i++)
    {

```

```

        a[i]%=10;
        b[i]%=10;
        tmp*=a[i]+b[i];
    }
    cout << tmp << endl;
    return 0;
}

```

输入: 2 3 5

输出: _____

3.

```

#include <iostream>
using namespace std;

const int c=2009;

int main()
{
    int n,p,s,i,j,t;
    cin >> n >> p;
    s=0;t=1;
    for(i=1;i<=n;i++)
    {
        t=t*p%c;
        for(j=1;j<=i;j++)
            s=(s+t)%c;
    }
    cout << s << endl;
    return 0;
}

```

输入: 11 2

输出: _____

4.

```

#include <iostream>
using namespace std;

const int maxn=50;

```

```

void getnext(char str[])
{
    int l=strlen(str),i,j,k,temp;
    k=l-2;
    while(k>=0&&str[k]>str[k+1]) k--;
    i=k+1;
    while(i<l&&str[i]>str[k]) i++;
    temp=str[k];
    str[k]=str[i-1];
    str[i-1]=temp;
    for(i=l-1;i>k;i--)
        for(j=k+1;j<i;j++)
            if(str[j]>str[j+1])
            {
                temp=str[j];
                str[j]=str[j+1];
                str[j+1]=temp;
            }
    return ;
}

int main()
{
    char a[maxn];
    int n;
    cin >> a >> n;
    while(n>0)
    {
        getnext(a);
        n--;
    }
    cout << a << endl;
    return 0;
}

```

输入: NOIP 3

输出: _____

四. 完善程序 (前 8 空, 每空 3 分, 后 2 空, 每空 2 分, 共 28 分)

1. (**最大连续子段和**) 给出一个数列 (元素个数不多于 100), 数列元素均为负整数、正整数、0。请找出数列中的一个连续子数列, 使得这个子数列中包含的所有元素之和最大, 在和最大的前提下还要求该子数列包含的元素个数最多, 并输出这个最大和以及该连续子数列中元素的个数。例如数列为 4, -5, 3, 2, 4 时, 输出 9 和 3; 数列为 1 2 3 -5 0 7 8 时, 输出 16 和 7。

```

#include <iostream>
using namespace std;

int a[101];
int n,i,ans,len,tmp,beg;

int main(){
    cin >> n;
    for (i=1;i<=n;i++)
        cin >> a[i];
    tmp=0;
    ans=0;
    len=0;
    beg= ①;
    for (i=1;i<=n;i++){
        if (tmp+a[i]>ans){
            ans=tmp+a[i];
            len=i-beg;
        }
        else if (② && i-beg>len)
            len=i-beg;
        if (tmp+a[i] ③){
            beg=④;
            tmp=0;
        }
        else
            ⑤;
    }
    cout << ans << " " << len << endl;
    return 0;
}

```

2. **(国王放置)** 在 $n*m$ 的棋盘上放置 k 个国王，要求 k 个国王互相不攻击，有多少种不同的放置方法。假设国王放在第 (x,y) 格，国王的攻击的区域是： $(x-1,y-1)$ ， $(x-1,y)$ ， $(x-1,y+1)$ ， $(x,y-1)$ ， $(x,y+1)$ ， $(x+1,y-1)$ ， $(x+1,y)$ ， $(x+1,y+1)$ 。读入三个数 n,m,k ，输出答案。题目利用回溯法求解。棋盘行标号为 $0\sim n-1$ ，列标号为 $0\sim m-1$ 。

```

#include <iostream>
using namespace std;

int n,m,k,ans;
int hash[5][5];
void work(int x,int y,int tot){

```

```

int i,j;
if (tot==k){
    ans++;
    return;
}
do{
    while (hash[x][y]){
        y++;
        if (y==m){
            x++;
            y=①;
        }
        if (x==n)
            return;
    }
    for (i=x-1;i<=x+1;i++)
        if (i>=0&&i<n)
            for (j=y-1;j<=y+1;j++)
                if (j>=0&&j<m)
                    ②;
    ③;
    for (i=x-1;i<=x+1;i++)
        if (i>=0&&i<n)
            for (j=y-1;j<=y+1;j++)
                if (j>=0&&j<m)
                    ④;

    y++;
    if (y==m){
        x++;
        y=0;
    }
    if (x==n)
        return;
}
while (1);
}
int main(){
    cin >> n >> m >> k;
    ans=0;
    memset(hash,0,sizeof(hash));
    ⑤;
    cout << ans << endl;
    return 0;
}

```