

# 省选模拟试题

ExfJoe

March 2, 2017

竞赛时长: 240min

试题名称	冒泡排序	字符串匹配	阅读
可执行文件名	bubble	match	reading
输入文件名	bubble.in	match.in	reading.in
输出文件名	bubble.out	match.out	reading.out
时间限制	1s	1s	1s
空间限制	256M	256M	256M
测试点数目	10	10	10
测试点分数	10	10	10
是否有 SPJ	否	否	否
是否有部分分	否	否	否
题目类型	传统	传统	传统

- 认真独立完成试题，不与他人交流讨论
- 最终评测在 Win10 下使用 Lemon，默认栈空间限制为 8M，不开启 O2 优化
- 试题按英文名称字典序排序

# 冒泡排序

## 题目描述

冒泡排序是一种简单的排序方式，其过程如下：

对  $1 \sim n$  的排列  $A$  进行冒泡排序：

```
bubble sort
counter = 0
While A 不是有序的
    counter = counter + 1
    For i = 1 to (n - 1)
        If A[i] > A[i + 1]
            Then Swap(A[i], A[i + 1])
        End If
    End For
End While
```

现在给定一个排列  $A$ ，请你求出冒泡排序几轮后  $A$  会有序 (即变量  $counter$  的值)。

## 输入格式

仅一行五个整数  $n, S, B, C, D$ ， $n$  表示排列长度。 $A$  数组用以下方式生成：

```
gen
For i = 1 to n
    A[i] = i
    S = (S * B + C) mod D
    Swap(A[i], A[(S mod i) + 1])
End For
```

## 输出格式

仅一行一个整数表示答案。

## 样例

```
Input
4 5 7 9 13
```

```
Output
2
```

## 约定

30% 的数据： $n \leq 10^3$

50% 的数据： $n \leq 10^5$

70% 的数据： $n \leq 10^6$

100% 的数据： $1 \leq n \leq 3 \times 10^7$ ， $0 \leq S, B, C < D \leq 10^9 + 7$

# 字符串匹配

## 题目描述

对于一个字符集大小为  $C$  的字符串  $P$ ，我们可以将任意两种字符在  $P$  中的位置进行互换，例如  $P = abcba$ ，我们交换  $a, b$  就变为  $bacab$ ，交换  $a, d$  就变为  $dbcb d$ ，交换可以进行任意次。若交换后  $P$  变为了字符串  $Q$ ，则我们称  $Q$  与  $P$  是匹配的。

现在给定两个字符集大小为  $C$  的字符串  $S, T$ ，请你求出  $S$  中有多少个连续子串与  $T$  是匹配的。

## 输入格式

第一行两个整数  $Case, C$  表示数据组数与字符集大小。字符用  $1 \sim C$  的整数来表示。

每组数据第一行两个整数  $n, m$  表示  $S$  的长度与  $T$  的长度。

第二行  $n$  个正整数表示  $S$ 。

第三行  $m$  个正整数表示  $T$ 。

## 输出格式

对于每组数据输出两行。

第一行一个正整数  $ans$ ，表示  $S$  中有多少个连续子串与  $T$  匹配。

接下来一行从小到大输出  $ans$  个整数，表示  $S$  中与  $T$  匹配的连续子串的首位置 (从 1 开始编号)。

## 样例

Input
3 3
6 3
1 2 1 2 3 2
3 1 3
6 3
1 2 1 2 1 2
3 1 3
6 3
1 1 2 1 2 1
3 1 3
Output
3
1 2 4
4
1 2 3 4
3
2 3 4

## 约定

10% 的数据： $n, m, C \leq 1000$

另有 20% 的数据： $n, m \leq 10^5, C \leq 40$

另有 30% 的数据： $n, m, C \leq 10^5$

100% 的数据： $1 \leq n, m, C \leq 10^6, Case = 3$

# 阅读

## 题目描述

A 君喜欢阅读，现在他准备读一本书，他会从第  $K$  页开始看，然后看到第  $M$  页。

书中的内容并不一定都让 A 君愉悦，或者说，A 君更喜欢看书中的精华。更具体地，书中有  $N$  页能让 A 君感到愉悦，阅读第  $T_i$  页可以获得  $B_i$  的愉悦度。

由于书的页数实在太多，因此 A 君会选择跳着看，但是他一次最多跳  $D$  页（两页页码差不大于  $D$ ），然后阅读跳到的那一页的内容，每次翻页他将会丧失  $A$  的愉悦度。

现在 A 君想知道他阅读完这本书，能得到的愉悦度之和最大能是多少。

## 输入格式

第一行五个非负整数  $K, M, D, A, N$ ，意义见题目描述。

接下来  $N$  行每行两个正整数  $T_i, B_i$ ，表示能让 A 君感到愉悦的一页。

## 输出格式

仅一行一个整数，表示最大的愉悦度之和。

## 样例

	Input	
0	10	4
10	2	
3	10	
8	5	
	Output	
-20		

## 样例解释

从第 0 页开始看。

跳到第 3 页并阅读，先丧失 10 点愉悦度再得到 10 点愉悦度。

跳到第 7 页再跳到第 10 页，丧失 20 点愉悦度。

-20 点愉悦度为最优策略。

## 约定

20% 的数据： $N \leq 1000$

另有 30% 的数据： $D \leq 100$

100% 的数据： $1 \leq B_i, A, D \leq 10^9$ ， $1 \leq N \leq 10^5$ ， $0 \leq K < T_1 < T_2 < \dots < T_N < M \leq 10^9$