

第一题 - 回文数

时间限制: 1 Sec 内存限制: 128 MB

题目描述

回文数的定义为：如果把一个数的各个数位上的数字颠倒过来得到的新数与原数相等，则此数是回文数

例：7, 22, 131, 2112, 31013..... 都是回文数。对任意给出的一个整数 n ，经过一系列的处理，最后都能成为回文数。处理的方法是，该数加上它的颠倒数

例如： $n=176$

第一次处理后 $176+671=847$

第二次处理后 $847+748=1595$

第三次处理后 $1595+5951=7546$

第四次处理后 $7546+6457=14003$

第五次处理后 $14003+30041=44044$

此时成为回文数，共进行5次处理。

给出 n 后，求出使该数按照以上规则进行一系列处理后成为回文数的最少操作次数。

输入描述

n 一个整数 ($1 \leq n \leq 1000000$)。

输出描述

使 n 成为回文数的最少处理次数。若开始给出的 n 是回文数，则输出 0 (即不需任何处理)。

样例输入1

67

样例输出1

2

说明

$67+76=143$

$143+341=484$ (回文数)。

第二题 - 图书管理员

时间限制: 1 Sec 内存限制: 256 MB

题目描述

图书馆中每本书都有一个图书编码，可以用于快速检索图书，这个图书编码是一个正整数。每位借书的读者手中有一个需求码，这个需求码也是一个正整数。如果一本书的图书编码恰好以读者的需求码结尾，那么这本书就是这位读者所需要的。小 D 刚刚当上图书馆的管理员，她知道图书馆里所有书的图书编码，她请你帮她写一个程序，对于每一位读者，求出他所需要的书中图书编码最小的那本书，如果没有他需要的书，请输出 -1。

输入描述

第一行，包含两个正整数 n, q ，以一个空格分开，分别代表图书馆里书的数量和读者的数量。

接下来的 n 行，每行包含一个正整数，代表图书馆里某本书的图书编码。

接下来的 q 行，每行包含两个正整数，以一个空格分开，第一个正整数代表图书馆里读者的需求码的长度，第二个正整数代表读者的需求码。

输出描述

q 行，每行包含一个整数，如果存在第 i 个读者所需要的书，则在第 i 行输出第 i 个读者所需要的书中图书编码最小的那本书的图书编码，否则输出 -1。

样例输入1

```
5 5
2123
1123
23
24
24
2 23
3 123
3 124
2 12
2 12
```

样例输出1

```
23
1123
-1
-1
-1
```

说明

对于 20% 的数据， $1 \leq n \leq 2$ 。

另有 20% 的数据， $q = 1$ 。

另有 20% 的数据，所有读者的需求码的长度均为 1。

另有 20% 的数据，所有的图书编码按从小到大的顺序给出。

对于 100% 的数据， $1 \leq n \leq 1,000$ ， $1 \leq q \leq 1,000$ ，所有的图书编码和需求码均不超过 10,000,000。

第三题 - 标题统计

时间限制: 1 Sec 内存限制: 256 MB

题目描述

凯凯刚写了一篇美妙的作文，请问这篇作文的标题中有多少个字符？注意：标题中可能包含大、小写英文字母、数字字符、空格和换行符。统计标题字符数时，空格和换行符不计算在内。

输入描述

输入文件只有一行，一个字符串 s 。

输出描述

输出文件只有一行，包含一个整数，即作文标题的字符数（不含空格和换行符）。

样例输入1

```
234
```

样例输出1

```
3
```

样例输入2

```
Ca 45
```

样例输出2

```
4
```

说明

【输入输出样例 1 说明】标题中共有 3 个字符，这 3 个字符都是数字字符。

【输入输出样例 2 说明】标题中共有 5 个字符，包括 1 个大写英文字母，1 个小写英文字母和 2 个数字字符，还有 1 个空格。由于空格不计入结果中，故标题的有效字符数为 4 个。

【数据规模与约定】规定 $|s|$ 表示字符串 s 的长度（即字符串中的字符和空格数）。对于 40% 的数据， $1 \leq |s| \leq 5$ ，保证输入为数字字符及行末换行符。对于 80% 的数据， $1 \leq |s| \leq 5$ ，输入只可能包含大、小写英文字母、数字字符及行末换行符。对于 100% 的数据， $1 \leq |s| \leq 5$ ，输入可能包含大、小写英文字母、数字字符、空格和行末换行符。

第四题 - 龙虎斗

时间限制: 1 Sec 内存限制: 128 MB

题目描述

轩轩和凯凯正在玩一款叫《龙虎斗》的游戏，游戏的棋盘是一条线段，线段上有 n 个兵营（自左至右编号 $1 \sim n$ ），相邻编号的兵营之间相隔 1 厘米，即棋盘为长度为 $n-1$ 厘米的线段。 i 号兵营里有 c_i 位工兵。下面图 1 为 $n=6$ 的示例：

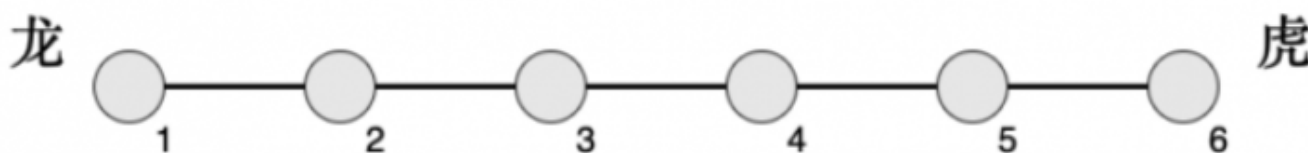


图 1. $n = 6$ 的示例

轩轩在左侧，代表“龙”；凯凯在右侧，代表“虎”。他们以 m 号兵营作为分界，靠左的工兵属于龙势力，靠右的工兵属于虎势力，而第 m 号兵营中的工兵很纠结，他们不属于任何一方。

一个兵营的气势为：该兵营中的工兵数 \times 该兵营到 m 号兵营的距离；参与游戏一方的势力定义为：属于这一方所有兵营的气势之和。

下面图 2 为 $n = 6, m = 4$ 的示例，其中红色为龙方，黄色为虎方：

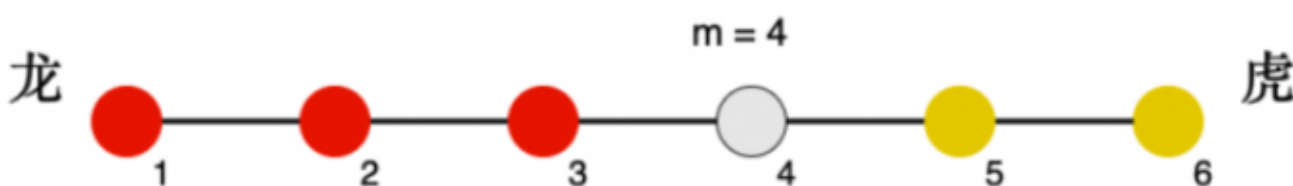


图 2. $n = 6, m = 4$ 的示例

游戏过程中，某一刻天降神兵，共有 s_1 位工兵突然出现在了 p_1 号兵营。作为轩轩和凯凯的朋友，你知道如果龙虎双方气势差距太悬殊，轩轩和凯凯就不愿意继续玩下去了。为了让游戏继续，你需要选择一个兵营 p_2 ，并将你手里的 s_2 位工兵全部派往兵营 p_2 ，使得双方气势差距尽可能小。

注意：你手中的工兵落在哪个兵营，就和该兵营中其他工兵有相同的势力归属（如果落在 m 号兵营，则不属于任何势力）。

输入描述

输入文件的第一行包含一个正整数 n ，代表兵营的数量。

接下来的一行包含 n 个正整数，相邻两数之间以一个空格分隔，第 i 个正整数代表编号为 i 的兵营中起始时的工兵数量 ci 。

接下来的一行包含四个正整数，相邻两数间以一个空格分隔，分别代表 $m, p1, s1, s2$ 。

输出描述

输出文件有一行，包含一个正整数，即 $p2$ ，表示你选择的兵营编号。如果存在多个编号同时满足最优，取最小的编号。

样例输入1

```
6
2 3 2 3 2 3
4 6 5 2
```

样例输出1

```
2
```

样例输入2

```
6
1 1 1 1 1 16
5 4 1 1
```

样例输出2

```
1
```

说明

【输入输出样例 1 说明】见问题描述中的图 2。

双方以 $m=4$ 号兵营分界，有 $s1=5$ 位工兵突然出现在 $p1=6$ 号兵营。龙方的气势为：

$$2 \times (4 - 1) + 3 \times (4 - 2) + 2 \times (4 - 3) = 14$$

虎方的气势为： $2 \times (5 - 4) + (3 + 5) \times (6 - 4) = 18$

当你将手中的 $s2=2$ 位工兵派往 $p2=2$ 号兵营时，龙方的气势变为： $14 + 2 \times (4 - 2) = 18$

此时双方气势相等。

【输入输出样例 2 说明】

双方以 $m=5$ 号兵营分界，有 $s1=1$ 位工兵突然出现在 $p1=4$ 号兵营。

龙方的气势为： $1 \times (5 - 1) + 1 \times (5 - 2) + 1 \times (5 - 3) + (1 + 1) \times (5 - 4) = 11$

虎方的气势为： $16 \times (6 - 5) = 16$

当你将手中的 $s_2=1$ 位工兵派往 $p_2=1$ 号兵营时，龙方的气势变为： $11 + 1 \times (5 - 1) = 15$

此时可以使双方气势的差距最小。

【数据规模与约定】

$1 < m < n, 1 \leq p_1 \leq n。$

对于 20% 的数据， $n = 3, m = 2, c_i = 1, s_1, s_2 \leq 100。$

另有 20% 的数据， $n \leq 10, p_1 = m, c_i = 1, s_1, s_2 \leq 100。$

对于 60% 的数据， $n \leq 100, c_i = 1, s_1, s_2 \leq 100。$

对于 80% 的数据， $n \leq 100, c_i, s_1, s_2 \leq 100。$

对于 100% 的数据， $n \leq 10^5, c_i, s_1, s_2 \leq 10^9。$

第五题 - 玩具谜题

时间限制: 1 Sec 内存限制: 500 MB

题目描述

小南有一套可爱的玩具小人, 它们各有不同的职业。

有一天, 这些玩具小人把小南的眼镜藏了起来。 小南发现玩具小人们围成了一个圈，它们有的面朝圈内，有的面朝圈外。如下图：



这时 `singer` 告诉小南一个谜题：“眼镜藏在我左数第 3 个玩具小人的右数第 1 个玩具小人的左数第 2 个玩具小人那里。”

小南发现，这个谜题中玩具小人的朝向非常关键，因为朝内和朝外的玩具小人的左右方向是相反的：面朝圈内的玩具小人，它的左边是顺时针方向，右边是逆时针方向；而面向圈外的玩具小人，它的左边是逆时针方向，右边是顺时针方向。

小南一边艰难地辨认着玩具小人，一边数着：

`singer` 朝内，左数第 3 个是 `archer`。

`archer` 朝外，右数第 1 个是 `thinker`。

`thinker` 朝外，左数第 2 个是 `writer`。

所以眼镜藏在 `writer` 这里！

虽然成功找回了眼镜，但小南并没有放心。如果下次有更多的玩具小人藏他的眼镜，或是谜题的长度更长，他可能就无法找到眼镜了。所以小南希望你写程序帮他解决类似的谜题。这样的谜题具体可以描述为：

有 `n` 个玩具小人围成一圈，已知它们的职业和朝向。现在第 `1` 个玩具小人告诉小南一个包含 `m` 条指令的谜题，其中第 `z` 条指令形如“左数 / 右数第 `s` 个玩具小人”。你需要输出依次数完这些指令后，到达的玩具小人的职业。

输入描述

输入的第一行包含两个正整数 n, m ，表示玩具小人的个数和指令的条数。

接下来 n 行，每行包含一个整数和一个字符串，以逆时针为顺序给出每个玩具小人的朝向和职业。其中 0 表示朝向圈内， 1 表示朝向圈外。保证不会出现其他的数。字符串长度不超过 10 且仅由小写字母构成，字符串不为空，并且字符串两两不同。整数和字符串之间用一个空格隔开。

接下来 m 行，其中第 i 行包含两个整数 a_i, s_i ，表示第 i 条指令。若 $a_i=0$ ，表示向左数 s_i 个人；若 $a_i=1$ ，表示向右数 s_i 个人。保证 a_i 不会出现其他的数， $i \leq s_i < n$ 。

输出描述

输出一个字符串，表示从第一个读入的小人开始，依次数完 m 条指令后到达的小人的职业。

样例输入1

```
7 3
0 singer
0 reader
0 mengbier
1 thinker
1 archer
0 writer
1 magician
0 3
1 1
0 2
```

样例输出1

```
writer
```

样例输入2

```
10 10
1 C
0 r
0 P
1 d
1 e
1 m
1 t
1 y
1 u
0 v
1 7
1 1
1 4
0 5
```



```
0 3
0 1
1 6
1 2
0 8
0 4
```

样例输出2

```
y
```

说明

【样例1说明】

这组数据就是【题目描述】中提到的例子。

【子任务】

子任务会给出部分测试数据的特点。如果你在解决题目中遇到了困难，可以尝试只解决一部分测试数据。

每个测试点的数据规模及特点如下表：

测试点	n	m	全朝内	全左数	$s_i = 1$	职业长度为1
1	$= 20$	$= 10^3$	√	√	√	√
2			×			
3			√	×		
4			×			
5			√	√	×	
6			×			
7			√	×		
8			×			
9			√	√	√	
10			×			
11			√	×		
12			×			
13			√	√	×	
14			×			
15			√	×		
16			×			
17	$= 10^5$	$= 10^5$	√	√	×	×
18			×			
19			√	×		
20			×			

其中一些简写的列意义如下：

1. 全朝内：若为“√”，表示该测试点保证所有的玩具小人都朝向圈内；
2. 全左数：若为“√”，表示该测试点保证所有的指令都向左数，即对任意的 $1 \leq z \leq m, a_i = 0$ ；
3. $s=1$ ：若为“√”，表示该测试点保证所有的指令都只数 1 个，即对任意的 $1 \leq z \leq m, s_i = 1$ ；
4. 职业长度为 1：若为“√”，表示该测试点保证所有玩具小人的职业一定是一个长度为 1 的字符串。