

## 11.13 模拟赛

StudyingFather

2020 年 11 月 12 日

题目名称	生成数	单词	喷泉	考试
题目类型	传统	传统	传统	传统
目录	number	word	fountain	exam
输入文件名	number.in	word.in	fountain.in	exam.in
输出文件名	number.out	word.out	fountain.out	exam.out
时间限制	2.0 秒	1.0 秒	1.5 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
子任务数目	1	2	3	6
子任务是否等分	是	否	否	否

注意事项:

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中函数 `main()` 的返回值类型必须是 `int`，程序正常结束时的返回值必须是 `0`。
3. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
4. 题目栈空间限制和内存限制一致。
5. 评测在 NovaOJ 上进行，比赛采用 OI 赛制，即每道题取最后一次提交计分。
6. 对于采用子任务捆绑测试的题目，你在该题上的得分等于各子任务的得分之和，而各子任务的得分，等于该子任务下每个测试点的最低得分。
7. 赛后可以在组题人的洛谷博客<sup>1</sup>上找到题解。

---

<sup>1</sup><https://studyingfather.blog.luogu.org/simulation-contests-log>

# 1 生成数

## 1.1 题目描述

小 W 最初有一个整数  $x$ ，满足  $1 \leq x \leq N$ 。在小 W 执行零次或多次下列操作后，这个整数变成了  $N$ 。

- 设  $x$  各十进制位上的数字和为  $p$ ，将  $x$  加上  $p$  形成新的整数  $x$ 。

现在你要求出有多少个  $x$  满足前述条件。

## 1.2 输入格式

输入仅包含一个整数  $N$ 。

## 1.3 输出格式

输出一个整数，代表满足条件的  $x$  的个数。

## 1.4 样例

### 1.4.1 样例输入 1

1 13

### 1.4.2 样例输出 1

1 4

### 1.4.3 样例解释 1

假如小 W 最初有的整数为 5，则可以通过如下操作序列变为 13:  $5 \rightarrow 10 \rightarrow 11 \rightarrow 13$ 。  
容易发现，10, 11, 13 也满足条件。

### 1.4.4 样例输入 2

1 20

### 1.4.5 样例输出 2

1 1

### 1.4.6 样例输入 3

1 2019

#### 1.4.7 样例输出 3

1 449

#### 1.5 子任务

对于 100% 的数据， $1 \leq N \leq 10^6$ 。

## 2 单词

### 2.1 题目描述

小 X 在读书的时候，遇到了一个令他感到害怕的单词。为了战胜恐惧，他决定玩一个游戏。

他害怕的单词可以用一个长度为  $n$  的字符串表示，其中只包含小写字母。游戏开始时，他需要将自己的手指指到这个单词的任意一个位置，并写下该位置对应的字母。接下来，他可以执行如下两个操作中的任意一个：

- 将自己的手指向左移动到一个字母或向右移动一个字母（要求该位置存在）。接下来，小 X 将会接着之前写下的内容，写下自己手指指向的字母。
- 将自己的手指移动到任意一个与当前位置字母相同的位置，这种情况下，小 X 将不会写下任何东西。

若刚开始手指位于字符串的第  $x$  个字符，目标位置位于第  $y$  个字符，则一次操作的花费为  $|x - y|$ 。

当小 X 写下他最喜欢的单词时，他就战胜了恐惧。现在他想知道，他要战胜恐惧所需的最小花费是多少。

### 2.2 输入格式

第一行包含两个整数  $n, m$ 。

第二行一个长度为  $n$  的字符串，代表令小 X 感到害怕的单词  $S$ 。

第三行一个长度为  $m$  的字符串，代表小 X 最喜欢的单词  $T$ 。

### 2.3 输出格式

若小 X 无法战胜恐惧，输出 -1。

否则，输出一个整数，代表他战胜恐惧所需的最小花费。

### 2.4 样例

#### 2.4.1 样例输入 1

```
1 2 2
2 wa
3 ac
```

#### 2.4.2 样例输出 1

```
1 -1
```

### 2.4.3 样例输入 2

```
1 7 7
2 monolog
3 nogolom
```

### 2.4.4 样例输出 2

```
1 10
```

### 2.4.5 样例输入 3

```
1 14 5
2 niskoobrazovan
3 boook
```

### 2.4.6 样例输出 3

```
1 5
```

### 2.4.7 样例解释 3

小 X 的操作序列如下：

- 刚开始小 X 将手指指向  $S$  中的第七个位置，并写下该位置的字母  $b$ 。
- 接下来他将手指向左移动一次，写下  $o$ 。
- 接下来他将手指再向左移动一次，写下  $o$ 。
- 接下来他用第二种移动方式将手指移动到第六个位置。
- 接下来他将手指向左移动一次，写下  $o$ 。
- 接下来他将手指再向左移动一次，写下  $k$ 。

总花费为 5。可以证明没有更优的方案。

## 2.5 子任务

所有数据均满足： $1 \leq n, m \leq 300$ ， $S, T$  中均只有小写字母。

各子任务的约束条件如下：

子任务编号	约束	分值
1	$S$ 中的所有字符两两不同	25
2	无特殊约束	75

## 3 喷泉

### 3.1 题目描述

一个由  $N$  层储水装置竖直排列的人造喷泉如下图所示，从上到下分别给储水装置编号为 1 到  $N$ ：

每个储水装置都有其直径，容量和一个阀门，阀门可以向装置内注入任意体积的水。每当装置内水的体积超出了装置容量，多余的水就会向下流向离这个装置最近且直径严格大于这个装置的某装置中，如果没有这样的装置，则水将直接流向最下层的水路中。

你需要回答  $Q$  个独立的询问。对于每个询问，你需要求出：如果向  $R_i$  容器中注入  $V_i$  升水，最后的水流会在编号为多少的装置处停止？如果水流止于最下层的水路，请输出 0。

### 3.2 输入格式

第一行包含两个整数  $N, Q$ 。

接下来  $N$  行，每行包含两个整数  $D_i, C_i$ ，分别表示第  $i$  个装置的直径和容量。

接下来  $Q$  行，每行包含两个整数  $R_i, V_i$ 。

### 3.3 输出格式

输出  $Q$  行，每行按顺序输出对于每个询问的回答。

### 3.4 样例

#### 3.4.1 样例输入 1

```
1 6 5
2 4 10
3 6 8
4 3 5
5 4 14
6 10 9
7 4 20
8 1 25
9 6 30
10 5 8
11 3 13
12 2 8
```

#### 3.4.2 样例输出 1

```
1 5
2 0
3 5
4 4
5 2
```

### 3.4.3 样例解释 1

前两个询问的图示如题目描述中图片所示。

因为每个询问互相独立，对于第三个询问，5 号装置中的水不会溢出。

### 3.4.4 样例 2

见附加文件下的 *fountain/fountain2.in* 和 *fountain/fountain2.ans*。

该样例满足子任务 2 的限制。

### 3.4.5 样例 3

见附加文件下的 *fountain/fountain3.in* 和 *fountain/fountain3.ans*。

该样例满足子任务 3 的限制。

## 3.5 子任务

对于全部数据， $2 \leq N \leq 10^5$ ,  $1 \leq Q \leq 2 \times 10^5$ ,  $1 \leq C_i \leq 1000$ ,  $1 \leq D_i, V_i \leq 10^9$ ,  $1 \leq R_i \leq N$ 。

详细子任务附加限制与分值如下表所示：

子任务编号	约束	分值
1	$N \leq 1000, Q \leq 2000$	30
2	$\forall i \in [1, n - 1], D_i \leq D_{i+1}$	30
3	无附加约束	40

## 4 考试

### 4.1 题目描述

$N$  名学生正坐成一排参加考试，他们从左到右编号为 1 到  $N$ 。目前学生做的情况都已经知道了，第  $i$  名学生将会打恰好  $A_i$  分。

某些时刻监考人员休息，他们会离开一会儿。就在监考离开的时候，学生们就可以作弊：任意连续且不少于两名学生可以聚在一起，抄他们当中打分最高的学生的卷子。最后，他们的分数就等于那段学生中的最高分。作弊可以发生任意次（也可能不发生）。

为了通过考试，第  $i$  名学生需要打恰好  $B_i$  分。请求出最多会有多少学生通过考试。

### 4.2 输入格式

第一行包含一个整数  $N$ 。

接下来一行  $N$  个整数，分别为  $A_1, A_2, \dots, A_N$ 。

接下来一行  $N$  个整数，分别为  $B_1, B_2, \dots, B_N$ 。

### 4.3 输出格式

输出一行一个整数，表示最多通过考试的学生数。

### 4.4 样例

#### 4.4.1 样例输入 1

```
1 3
2 1 2 3
3 2 2 2
```

#### 4.4.2 样例输出 1

```
1 2
```

#### 4.4.3 样例解释 1

在第一组样例中，前两个学生可以互相抄袭，最后分数变为 2, 2, 3，1, 2 两名学生就会通过考试。

#### 4.4.4 样例输入 2

```
1 4
2 10 1 9 1
3 10 9 10 9
```

#### 4.4.5 样例输出 2

```
1 3
```



#### 4.4.6 样例解释 2

在第二组样例中，学生 2 或 3 可以通过考试，但是他们不可能同时通过考试。  
注意这组数据不可能在子任务 2, 3, 4 中出现。

#### 4.4.7 样例 3

见附加文件下的 *exam/exam3.in* 和 *exam/exam3.ans*。  
该样例满足子任务 2 的限制。

#### 4.4.8 样例 4

见附加文件下的 *exam/exam4.in* 和 *exam/exam4.ans*。  
该样例满足子任务 3 的限制。

#### 4.4.9 样例 5

见附加文件下的 *exam/exam5.in* 和 *exam/exam5.ans*。  
该样例满足子任务 4 的限制。

#### 4.4.10 样例 6

见附加文件下的 *exam/exam6.in* 和 *exam/exam6.ans*。  
该样例满足子任务 6 的限制。

### 4.5 子任务

对于所有数据， $2 \leq N, 1 \leq A_i, B_i \leq 10^9$ 。

详细子任务附加限制及分值如下表：

子任务编号	约束	分值
1	$N \leq 10$	14
2	$N \leq 10^5$ , $B$ 中所有元素均相等	12
3	$N \leq 5000$ , $A$ 中所有元素单调递增	13
4	$N \leq 10^5$ , $A$ 中所有元素均不同	23
5	$N \leq 200$	16
6	$N \leq 5000$	22