

# Russia contest 5

Created: 4/27/2025 23:00

Updated: 4/27/2025 23:02

Exported: 4/27/2025 23:02

## Đề bài

### Bài A. Thẩm vấn nghi phạm

Khi thám tử Benoit Blanc nhận được một phong bì chứa tiền và thư nặc danh đề nghị điều tra một vụ án, ông lập tức đến hiện trường để thẩm vấn các nghi phạm.

Có  $n$  nghi phạm, mỗi nghi phạm thứ  $i$  có độ *nhàm chán* là  $a_i$ . Tại mọi thời điểm, độ *nhàm chán* của vụ án là tổng độ *nhàm chán* của tất cả nghi phạm đã được thẩm vấn.

Tuy nhiên, Benoit Blanc rất ghét các vụ án nhàm chán. Có  $m$  *điểm tới hạn* (critical points), mỗi điểm là một giá trị  $b_j$ . Nếu tổng độ *nhàm chán* vượt quá  $b_j$ , thì mức độ hứng thú của Blanc với vụ án sẽ giảm xuống một lượng cố định.

Nhiệm vụ của bạn là sắp xếp thứ tự thẩm vấn các nghi phạm sao cho số lượng *điểm tới hạn* bị vượt qua là ít nhất, để giữ cho thám tử hứng thú nhất với vụ án.

---

## Input:

- Dòng 1: Số nguyên  $n$  — số lượng nghi phạm ( $1 \leq n \leq 2 \cdot 10^5$ ).
  - Dòng 2:  $n$  số nguyên  $a_1, a_2, \dots, a_n$  — độ *nhàm chán* của từng nghi phạm ( $-10^9 \leq a_i \leq 10^9$ ).
  - Dòng 3: Số nguyên  $m$  — số lượng *điểm tới hạn* ( $1 \leq m \leq 2 \cdot 10^5$ ).
  - Dòng 4:  $m$  số nguyên  $b_1, b_2, \dots, b_m$  — giá trị các *điểm tới hạn* ( $0 \leq b_j \leq 10^9$ ).
-

## Output:

- Dòng 1: Số lượng *điểm tới hạn* tối thiểu có thể bị vượt qua.
- Dòng 2: Thứ tự thăm vấn **n** nghi phạm (in ra chỉ số từ 1 đến n).

Nếu có nhiều cách tối ưu, in ra bất kỳ.

## Ví dụ:

Input :

```
3
2 3 1
4
1 7 10 5
```

Output :

```
2
1 2 3
```

Input :

```
4
10 -10 20 -20
5
11 12 3 24 15
```

Output :

```
0
2 4 1 3
```

## Tóm tắt:

- Có **n** nghi phạm với độ nhàm chán **a<sub>i</sub>**.
- Có **m** ngưỡng nhàm chán **b<sub>j</sub>**.
- Mỗi khi tổng độ nhàm chán vượt quá **b<sub>j</sub>**, hứng thú của thám tử giảm xuống.
- Yêu cầu: sắp xếp thứ tự thẩm vấn để **số ngưỡng bị vượt qua ít nhất**.

## Phân tích Subtask:

| Subtask | Điểm | Ràng buộc thêm                     | Yêu cầu qua subtask |
|---------|------|------------------------------------|---------------------|
| 1       | 20   | $n, m \leq 10$                     | Không có lỗi        |
| 2       | 10   | Tất cả <b>a<sub>i</sub> &lt; 0</b> | Không có lỗi        |
| 3       | 10   | Tất cả <b>a<sub>i</sub> &gt; 0</b> | Không có lỗi        |
| 4       | 25   | $n, m \leq 1000$                   | Qua subtask 1       |
| 5       | 35   | Không giới hạn (full constraint)   | Qua subtask 1-4     |

## Đề bài

### Bài B. Điều tra chứng cứ

Thám tử Benoit Blanc đang điều tra một vụ án bí ẩn và thu thập được **n** chứng cứ, mỗi chứng cứ thứ **i** có độ *nhẹ* là **a<sub>i</sub>**.

Blanc tin rằng những chứng cứ *nhẹ* nhất có thể là quan trọng nhất, nên ông sử dụng một phương pháp điều tra như sau:

- Chọn một chứng cứ ban đầu (vị trí **x**).

- Di chuyển sang bên trái, nếu chứng cứ bên trái có độ nặng **nhỏ hơn hoặc bằng** hiện tại thì tiếp tục di chuyển.
- Tuy nhiên, với các chứng cứ có **độ nặng bằng nhau**, Blanc chỉ di chuyển tối đa **k lần** liên tiếp (ông dễ chán!).

Ví dụ: Với dãy trọng số chứng cứ `[3, 3, 3, 4, 4, 5]`, bắt đầu từ cuối, với **k = 2**:

- Bắt đầu từ chứng cứ 6 (giá trị 5).
- Sang trái (giá trị 4), tiếp tục sang trái (4), sang trái (3), sang trái (3), nhưng **dừng lại** ở đây vì đã di chuyển qua **2 chứng cứ bằng nhau (3)**.

Blanc sẽ thực hiện **m** thí nghiệm, mỗi lần bắt đầu từ một vị trí khác nhau. Nhiệm vụ của bạn là giúp Blanc biết sau mỗi thí nghiệm, ông dừng lại ở chứng cứ nào.

## Input:

- Dòng 1: Số nguyên **n** — số lượng chứng cứ ( $1 \leq n \leq 4 \cdot 10^5$ ).
- Dòng 2: **n** số nguyên **a<sub>1</sub>, a<sub>2</sub>, ..., a<sub>n</sub>** — trọng số của từng chứng cứ ( $1 \leq a_i \leq 10^9$ ).
- Dòng 3: Hai số nguyên **m** và **k** — số thí nghiệm và số lần tối đa được di chuyển qua các chứng cứ bằng nhau ( $1 \leq m \leq 4 \cdot 10^5$ ,  $0 \leq k \leq n$ ).
- Dòng 4: **m** số nguyên **x<sub>1</sub>, x<sub>2</sub>, ..., x<sub>m</sub>** — vị trí bắt đầu mỗi thí nghiệm ( $1 \leq x_i \leq n$ ).

## Output:

- In ra **m** số nguyên — vị trí cuối cùng Blanc dừng lại sau mỗi thí nghiệm.

## Ví dụ:

Input :

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

Output :

```
1 1 2 2
```

Input :

```
7
1 5 7 2 10 10 6
7 0
1 2 3 4 5 6 7
```

Output :

```
1 1 1 4 4 6 7
```

## Tóm tắt:

- Có **n** chứng cứ với trọng số **a<sub>i</sub>**.
- Thực hiện **m** lần thí nghiệm:
  - Mỗi lần bắt đầu từ chứng cứ thứ **x**.
  - Di chuyển sang trái khi trọng số không tăng, nhưng chỉ **k lần** qua các chứng cứ bằng nhau.

## Phân tích Subtask:

| Subtask | Điểm | Ràng buộc thêm                   | Yêu cầu qua subtask |
|---------|------|----------------------------------|---------------------|
| 1       | 13   | $n, m \leq 10$                   | Không có lỗi        |
| 2       | 12   | $n, m \leq 1000, k = 0$          | Không có lỗi        |
| 3       | 15   | $n, m \leq 1000, k = 1$          | Không có lỗi        |
| 4       | 18   | $n, m \leq 1000$                 | Qua subtask 1-3     |
| 5       | 19   | $k = 0$                          | Qua subtask 2       |
| 6       | 23   | Không giới hạn (full constraint) | Qua subtask 1-5     |

## Đề bài

### Bài C. Thông điệp mã hóa

Khi điều tra hiện trường vụ án, thám tử Benoit Blanc tìm thấy một mẫu giấy ghi chú khả nghi. Văn bản trên ghi chú là một chuỗi các chữ cái **t**, thoạt nhìn không có ý nghĩa gì. Tuy nhiên, sau khi phân tích kỹ, Blanc nhận ra đây là một thông điệp đã được mã hóa.

Phương pháp mã hóa như sau:

- Có **n** từ mã hóa: **w<sub>1</sub>, w<sub>2</sub>, ..., w<sub>n</sub>**.
- Các từ này được nối lại thành một chuỗi: **w<sub>1</sub> + w<sub>2</sub> + ... + w<sub>n</sub>**.
- Sau đó, chuỗi này được **đảo ngược** và nối tiếp vào sau (tức là thêm **w<sub>n</sub> + ... + w<sub>2</sub> + w<sub>1</sub>**).
- Kết quả thu được chuỗi **t**, không có dấu cách hay ngăn cách nào.

Công việc của bạn là giúp Blanc khôi phục lại các từ **w<sub>1</sub>, w<sub>2</sub>, ..., w<sub>n</sub>** sao cho:

- Số lượng từ **n** là **lớn nhất có thể**.

Nếu có nhiều cách khôi phục thỏa mãn điều kiện trên, in ra bất kỳ cách nào.

---

## Input:

- Một chuỗi **t** gồm các chữ cái thường ( $1 \leq |t| \leq 10^6$ ).
- 

## Output:

- Dòng 1: Số nguyên **n** — số lượng từ tối đa.
  - Tiếp theo **n** dòng, mỗi dòng in ra một từ.
- 

## Ví dụ:

Input:

abaccaba

Output:

4

a

b

a

c

Input:

guesswhoitisisitwhoguess

Output:

4

guess  
who  
it  
is

**Input:**

xaabaababaaaabxaa

**Output:**

5  
xaa  
b  
a  
a  
ba

## Tóm tắt:

- Chuỗi **t** gồm các từ  $w_1 + \dots + w_n + w_n + \dots + w_1$  (nối xuôi rồi nối ngược).
- Cần chia **t** thành các từ ban đầu sao cho **số từ lớn nhất**.

## Phân tích Subtask:

| Subtask | Điểm | Ràng buộc thêm           | Yêu cầu qua subtask |
|---------|------|--------------------------|---------------------|
| 1       | 13   |                          | t                   |
| 2       | 18   |                          | t                   |
| 3       | 23   |                          | t                   |
| 4       | 20   | t chỉ gồm chữ 'a' và 'b' | Không có lỗi        |



| Subtask | Điểm | Ràng buộc thêm                   | Yêu cầu qua subtask |
|---------|------|----------------------------------|---------------------|
| 5       | 26   | Không giới hạn (full constraint) | Qua subtask 1-4     |

## Đề bài

### Bài D. Mạng lưới tội phạm

Thám tử Benoit Blanc đang điều tra một vụ án liên quan đến một **băng đảng mafia** lớn. May mắn thay, lần này mafia là nạn nhân và muốn hợp tác điều tra.

- Băng đảng có  **$n$**  thành viên trong ban lãnh đạo, được tổ chức theo dạng **cây có gốc tại người đứng đầu** (người số 1).
- Mỗi thành viên thứ  **$i$**  (từ 2 đến  $n$ ) có **sếp trực tiếp** là  **$p_i$** .
- Mỗi thành viên lãnh đạo thứ  **$i$**  có  **$a_i$**  lính dưới quyền (các lính này không trùng nhau giữa các lãnh đạo).

Thông tin truyền tin:

- Khi một người nhận được tin nhắn, họ **lập tức truyền tin** cho toàn bộ lính dưới quyền của mình.
- Nhưng truyền tin giữa các lãnh đạo (từ sếp sang cấp dưới) tốn  **$t_i$**  phút.

Blanc chỉ có  **$T$  phút** và **1 cuộc gọi duy nhất** để bắt đầu truyền tin cho một lãnh đạo bất kỳ. Nhiệm vụ của bạn là chọn **người lãnh đạo phù hợp nhất** để gọi, sao cho:

- Trong  **$T$  phút**, **số người nhận được tin là lớn nhất** (gồm lãnh đạo, lính của họ, và những người cấp dưới khác truyền tin đủ thời gian).

## Input:

- Dòng 1: Hai số nguyên **n** và **T** ( $1 \leq n \leq 10^5$ ,  $0 \leq T \leq 10^9$ ).
- Dòng 2: **n-1** số nguyên **p<sub>2</sub>, ..., p<sub>n</sub>** — sắp trực tiếp của lãnh đạo từ 2 đến n ( $1 \leq p_i \leq n$ ).
- Dòng 3: **n-1** số nguyên **t<sub>2</sub>, ..., t<sub>n</sub>** — thời gian truyền tin từ sếp đến lãnh đạo tương ứng ( $0 \leq t_i \leq 10^9$ ).
- Dòng 4: **n** số nguyên **a<sub>1</sub>, ..., a<sub>n</sub>** — số lính dưới quyền của mỗi lãnh đạo ( $0 \leq a_i \leq 10^9$ ).

## Output:

- Hai số nguyên: **người lãnh đạo nên gọi** và **tổng số người nhận được tin nhắn**.

Nếu có nhiều phương án tối ưu, in ra bất kỳ.

## Ví dụ:

Input:

```
3 10
1 1
9 11
100 49 51
```

Output:

```
1 151
```

Input:

```
7 15
1 1 2 2 3 3
```

```
9 8 6 9 16 5
400 100 200 50 1000 1100 300
```

```
Output:
2 1153
```

## Tóm tắt:

- Cây **nút gốc 1**, có  **$n$**  lãnh đạo.
- Truyền tin **tốn thời gian** từ sếp đến cấp dưới ( **$t_i$** ).
- Mỗi lãnh đạo có một số lượng **lính dưới quyền** ( **$a_i$** ).
- Chọn một lãnh đạo để bắt đầu truyền tin, tối đa  **$T$  phút**, sao cho **nhiều người nhận được tin nhất**.

## Phân tích Subtask:

| Subtask | Điểm | Ràng buộc thêm   | Yêu cầu qua subtask |
|---------|------|--|---------------------|
| 1       | 16   | $n \leq 1000$  | Không có lỗi        |
| 2       | 16   | <b><math>p_i = i - 1</math></b> với mọi $i$                            | Không có lỗi        |
| 3       | 10   | <b><math>T = 10</math>; <math>t_i = 1</math>, <math>a_i = 0</math></b> | Không có lỗi        |
| 4       | 16   | <b><math>T = 10^3</math></b>   | Qua subtask 3       |
| 5       | 20   | <b><math>a_i = 0</math></b> với mọi $i$                                | Qua subtask 3       |
| 6       | 22   | Không giới hạn (full constraint)                                       | Qua subtask 1-5     |