

Mục lục

LISttrace	2
ICBUS	3
COBOX	4
TSPtrace	5
TOWERtrace	6
TELMOV	7
LCSttrace	9
GOLDtrace	10
MARBLEtrace	11
Xếp hàng — LQUEUE	13
Giao hàng — SHIPCOUNT	15
Điểm bán lẻ — RETOUT	16

Nộp bài tại: `scoss.soict.ai/cmslogin/NAN`

Username/Password: nick codeforces

Bài 1. LIStrace

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Dãy con của một dãy là dãy thu được khi xóa đi một vài phần tử của dãy ban đầu và giữ nguyên thứ tự của các phần tử còn lại (có thể không xóa phần tử nào). Một dãy được gọi là tăng dần nếu phần tử sau lớn hơn phần tử trước.

Cho một dãy số nguyên a , hãy tìm dãy con tăng dài nhất của a

Dữ liệu vào

- Dòng đầu chứa số phần tử của dãy a : n
- Dòng tiếp theo chứa dãy a

Kết quả

- Dòng đầu chứa một số nguyên là độ dài dãy con dài nhất tìm được.
- Dòng thứ hai chứa dãy các chỉ số của dãy con đó, theo thứ tự tăng dần.

Ví dụ

stdin	stdout
8 3 1 5 2 6 4 5 7	5 2 4 6 7 8

Hạn chế

- $n \leq 10^5$. $1 \leq a_i \leq 10^5$
- 30% test với $n \leq 20$
- 30% test với $20 < n \leq 1000$

Bài 2. ICBUS

Quốc gia Backoi có N thành phố, mỗi thành phố có một hệ thống xe chạy liên tỉnh khác nhau. Một xe có thể chạy từ thành phố i sang thành phố j nếu như có đường nối trực tiếp giữa hai thành phố này. Các con đường ở đây đều là đường 2 chiều. Mỗi hệ thống xe liên tỉnh có một số luật như sau:

- Hành khách muốn sử dụng hệ thống xe của thành phố i thì bắt buộc phải bắt xe tại thành phố i .
- Giá vé xe của thành phố i là đồng hạng C_i bất kể quãng đường bao xa.
- Hệ thống xe của thành phố i chỉ cho phép chạy tối đa qua D_i thành phố.

Quân là một hành khách muốn đi từ thành phố 1 đến thành phố N . Hãy giúp Quân tìm cách đi sao cho tổng chi phí là thấp nhất.

Dữ liệu vào

Dòng đầu tiên chứa hai số nguyên dương N và K ($2 \leq N \leq 5000$; $N - 1 \leq K \leq 10000$).

N dòng tiếp theo, dòng thứ i chứa 2 số nguyên dương C_i và D_i ($1 \leq C_i \leq 10000$; $1 \leq D_i \leq N$) là 2 thông tin của hệ thống xe của thành phố i .

K dòng tiếp theo mỗi dòng ghi hai số i và j ($1 \leq i < j \leq N$) biểu thị giữa 2 thành phố i và j có đường nối trực tiếp.

Kết quả

Ghi ra duy nhất một số là chi phí Quân phải trả để đi từ thành phố 1 đến thành phố N . Dữ liệu đảm bảo luôn có cách đi từ thành phố 1 đến thành phố N .

Ví dụ

test	answer
6 6 400 2 200 1 500 3 900 1 400 4 200 5 1 2 1 5 2 3 2 4 3 6 4 6	800

Giải thích

Quân sử dụng lần lượt hệ thống xe của thành phố 1 rồi thành phố 5.

Bài 3. COBOX

Cho n chiếc hộp được đánh số từ 1 đến n . Hộp thứ i có chiều dài a_i , chiều rộng b_i . Hộp i có thể đặt vào trong hộp j nếu i chưa bị chứa bởi hộp nào khác, j đang không chứa hộp nào khác và $a_i < a_j, b_i < b_j$. Cần tìm cách lồng các hộp vào nhau sao cho số hộp không bị lồng vào bất kỳ hộp nào là ít nhất. Nếu có nhiều cách lồng các hộp đều là tốt nhất, in ra cách bất kỳ

Dữ liệu vào

- Dòng đầu tiên chứa số nguyên dương n
- Dòng thứ i trong số n dòng tiếp theo ghi $a_i b_i$

Kết quả

- Dòng đầu chứa số nguyên dương k là số hộp còn lại
- k dòng tiếp theo, mỗi dòng mô tả một dãy hộp: Số đầu tiên là số lượng hộp bị lồng vào nhau và theo sau là chỉ số của các hộp bị lồng vào nhau, theo thứ tự từ ngoài vào trong của các hộp

Các dãy hộp có thể in ra theo thứ tự tùy ý

Ví dụ

test	answer
4 1 1 2 2 3 3 2 4	2 1 4 3 3 2 1

Hạn chế

- $1 \leq n \leq 10^5, 1 \leq a_i \leq 10^9$
- Subtask 1: $n \leq 5000$
- Subtask 2: $a_i = b_i$
- Subtask 3: Ràng buộc gốc

Bài 4. TSPtrace

Một người du lịch xuất phát từ thành phố thứ nhất muốn đi thăm quan tất cả $n - 1$ thành phố khác. mỗi thành phố đúng một lần, rồi quay trở lại thành phố xuất phát.

Yêu cầu: Cho biết chi phí đi lại giữa các thành phố, hãy giúp người du lịch tìm hành trình với tổng chi phí là nhỏ nhất.

Dữ liệu vào

Dòng đầu tiên chứa hai số nguyên dương n, m cách nhau bởi dấu cách ($n \leq 20, m < 400$).

m dòng tiếp theo mỗi dòng chứa ba số nguyên dương i, j, c ($i, j \leq n, c \leq 10^6$) biểu thị chi phí đi trực tiếp từ thành phố i đến thành phố j là c .

Lưu ý: nếu từ thành phố i đến thành phố j nào không mô tả chi phí đi lại thì có nghĩa là không có đường đi trực tiếp từ i đến j .

Kết quả

- Dòng đầu chứa một số nguyên là tổng chi phí hành trình nhỏ nhất tìm được;
- Dòng thứ hai chứa n số tương ứng với n đỉnh trên chu trình tìm được.

Ví dụ

test	answer
2 2	5
1 2 3	1 2
2 1 2	

Bài 5. TOWERtrace

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Có lẽ bạn đã nghe nói về truyền thuyết về Tháp Babylon. Ngày nay nhiều chi tiết của câu chuyện này đã bị lãng quên. Vì vậy, bây giờ, phù hợp với tính chất giáo dục của cuộc thi này, chúng tôi sẽ kể cho bạn toàn bộ câu chuyện:

Người babylon có n loại khối, và nguồn cung cấp không giới hạn của từng loại. Mỗi loại i là một khối hình chữ nhật có kích thước tuyến tính (x_i, y_i, z_i) . Một khối có thể được định hướng (xoay) lại để bất kỳ hai trong ba chiều của nó xác định kích thước của cơ sở và kích thước khác là chiều cao. Họ muốn xây dựng tòa tháp cao nhất có thể bằng cách xếp chồng các khối. Vấn đề là, trong việc xây dựng một tòa tháp, một khối chỉ có thể được đặt trên đỉnh của một khối khác, miễn là hai kích thước cơ sở của khối trên đều nhỏ hơn so với kích thước cơ sở tương ứng của khối dưới. Điều này có nghĩa các khối có kích thước bằng nhau ở một trong hai cạnh sẽ không thể được xếp chồng lên nhau.

Công việc của bạn là viết một chương trình xác định chiều cao của tòa tháp cao nhất mà người babylon có thể xây dựng với một tập hợp các khối nhất định.

Dữ liệu vào

- Dòng đầu chứa: n ($1 \leq n \leq 1000$)
- n dòng tiếp theo chứa: $x_i y_i z_i$ ($1 \leq x_i, y_i, z_i \leq 10^9$)

Có 50% test với $n \leq 7$

Kết quả

- Dòng đầu chứa hai số nguyên: M k là chiều cao lớn nhất tìm được và số khối đá của tháp.
- k dòng tiếp theo mỗi dòng chứa một khối đá: Chiều dài, chiều rộng, chiều cao (đã được xoay). Các khối đá cần được liệt kê theo thứ tự từ chân tháp đến đỉnh tháp.

Nếu có nhiều phương án đều tốt nhất, in ra phương án bất kỳ trong số đó.

Ví dụ

stdin	stdout
7	28 7
1 1 1	7 7 7
2 2 2	6 6 6
3 3 3	5 5 5
4 4 4	4 4 4
5 5 5	3 3 3
6 6 6	2 2 2
7 7 7	1 1 1

Bài 6. TELMOV

Cô kỹ sư Alice đang sống ở trong thiên hà VNOI2020. Trong thiên hà này có N hành tinh khác nhau và M kênh vận chuyển hai chiều dạng (x, y, t) cho phép bạn di chuyển từ hành tinh x đến hành tinh y (hoặc ngược lại) trong t giây.

Nhưng Alice nhận thấy phương pháp vận chuyển này rất kém hiệu quả nên đã phát triển một thiết bị cho phép bạn dịch chuyển từ hành tinh x đến bất kỳ hành tinh y nào khác trong P giây với điều kiện bạn có thể đến hành tinh y đó từ hành tinh x chỉ sử dụng tối đa L kênh vận chuyển.

Thiết bị này hiện mới là bản thử nghiệm nên không thể được sử dụng quá K lần. Alice đang ở hành tinh 1 và muốn biết thời gian tối thiểu để đến hành tinh N .

Yêu cầu: Viết chương trình tính thời gian tối thiểu cần thiết để đến được hành tinh N bắt đầu từ hành tinh 1.

Dữ liệu vào

Dòng đầu tiên chứa 5 giá trị N, M, P, L, K cách nhau một dấu cách.

Mỗi dòng trong số M dòng sau chứa 3 giá trị X_i, Y_i, T_i mô tả một kênh vận chuyển. Dữ liệu đảm bảo có nhiều nhất một kênh giữa hai hành tinh.

Kết quả

Kết quả ghi ra một giá trị duy nhất là thời gian tối thiểu cần thiết để đến hành tinh N bắt đầu từ hành tinh 1. Dữ liệu đảm bảo luôn có đáp án.

Ví dụ

test	answer	Giải thích
6 7 3 2 1 1 2 2 1 3 5 2 3 4 2 4 23 3 4 6 5 4 7 5 6 9	14	Thiết bị có thể được sử dụng một lần. Để đến hành tinh 6 trong thời gian tối thiểu, chúng ta sẽ đi qua kênh 1 \rightarrow 2 sau đó sẽ dịch chuyển đến hành tinh 5 từ đó sẽ đi qua kênh 5 \rightarrow 6. Chi phí cuối cùng là $2 + 3$ (dịch chuyển bởi thiết bị) $+ 9 = 14$.
6 7 3 2 0 1 2 2 1 3 5 2 3 4 2 4 23 3 4 6 5 4 7 5 6 9	27	Thiết bị hoàn toàn không thể sử dụng được. Để đến hành tinh 6 từ hành tinh 1 trong thời gian tối thiểu, cần đi qua các kênh theo thứ tự 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 và với thời gian $5 + 6 + 7 + 9 = 27$ giây.

Hạn chế

- $1 < N, \leq 10000, 1 < M \leq 20000$;
- $0 \leq L, K \leq 10$;
- $1 < T_i, P \leq 100000$;

- $1 < X_i, Y_i \leq N$;
- 24% số điểm ứng với các test có $K = 0$ và tất cả các kênh vận chuyển đều có $T_i = 1$;
- 16% số điểm ứng với các test khác có $K = 0$;
- 16% số điểm ứng với các test khác đảm bảo $N \leq 300$;

Bài 7. LCStrace

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Dãy con của một dãy là dãy thu được khi xóa đi một vài phần tử của dãy ban đầu và giữ nguyên thứ tự của các phần tử còn lại (có thể không xóa phần tử nào). Cho hai dãy a và b . Tìm dãy c vừa là dãy con của a , vừa là dãy con của b và có độ dài lớn nhất có thể.

Dữ liệu vào

- Dòng đầu chứa số phần tử của dãy a : n ($1 \leq n \leq 1000$).
- Dòng tiếp theo chứa dãy a : $a_1 a_2 \dots a_n$ ($1 \leq a_i \leq 10^9$).
- Dòng tiếp theo chứa số phần tử của dãy b : m ($1 \leq m \leq 1000$).
- Dòng tiếp theo chứa dãy b : $b_1 b_2 \dots b_m$ ($1 \leq b_i \leq 10^9$).

Có 50% test với $n \leq 20$

Kết quả

- Dòng đầu chứa một số nguyên là độ dài dãy c
- Dòng thứ hai chứa giá trị các phần tử trong dãy c theo thứ tự trên dãy.

Ví dụ

stdin	stdout
7 3 7 2 5 1 4 9 10 4 3 2 3 6 1 5 4 9 7	5 3 2 1 4 9

Hạn chế

- $n \leq 10^5$. $1 \leq a_i \leq 10^5$
- 30% test với $n \leq 20$
- 30% test với $20 < n \leq 1000$

Bài 8. GOLDtrace

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Vương quốc ALPHA có n kho vàng nằm trên một đường thẳng và được đánh số $1, 2, \dots, n$. Kho thứ i có số vàng là a_i (a_i là số nguyên không âm) và được đặt tại tọa độ i ($\forall i = 1, \dots, n$). Vua của ALPHA muốn tìm một tập hợp các kho vàng có tổng số vàng lớn nhất với điều kiện khoảng cách giữa hai kho được chọn phải lớn hơn hoặc bằng L_1 và nhỏ hơn hoặc bằng L_2 .

Dữ liệu vào

- Dòng đầu chứa: n , L_1 , and L_2 ($1 \leq n \leq 100000, 1 \leq L_1 \leq L_2 \leq n$).
- Dòng 2 chứa: a_1, a_2, \dots, a_n .

Kết quả

- Dòng đầu chứa hai số nguyên: M k là tổng số vàng lớn nhất tìm được và số kho vàng trong cách chọn.
- Dòng thứ 2 chứa k số nguyên là chỉ số của các kho hàng được chọn.

Nếu có nhiều cách chọn đều tốt nhất, in ra cách chọn bất kỳ trong số đó.

Ví dụ

stdin	stdout
6 2 3 3 5 9 6 7 4	19 3 1 3 5

Bài 9. MARBLEtrace

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Phong là một nhà điêu khắc, ông có một tấm đá cẩm thạch hình chữ nhật kích thước $W \times H$. Ông ta muốn cắt tấm đá thành các miếng hình chữ nhật kích thước $W_1 \times H_1, W_2 \times H_2, \dots, W_N \times H_N$. Ông ta muốn cắt đến tối đa các mẫu kích thước có thể. Tấm đá có những vân đá cho nên không thể xoay khi sử dụng, có nghĩa là không thể cắt ra miếng $B \times A$ thay cho miếng $A \times B$ trừ khi $A = B$. Các miếng phải được cắt tại các điểm nguyên trên hàng cột và mỗi nhát cắt phải cắt đến hết hàng hoặc hết cột. Sau khi cắt sẽ còn lại những mẫu đá còn thừa bỏ đi, nghĩa là những mẫu đá không thể cắt thành miếng kích thước cho trước nào.

Yêu cầu: Hãy tìm cách cắt sao cho còn ít nhất diện tích đá thừa bỏ đi.

Hình dưới minh họa cách cắt các phiến đá trong ví dụ với diện tích thừa ít nhất tìm được là 10.

10×4			10×4	
	6×2		6×2	6×2
7×5		7×5		7×5

Dữ liệu vào

Dòng đầu tiên chứa hai số nguyên: W và H .

Dòng thứ hai chứa một số nguyên N . N dòng tiếp theo mỗi dòng chứa hai số nguyên W_i và H_i .

Kết quả

Dòng đầu chứa hai số nguyên là diện tích thừa và số lần cắt

Các dòng tiếp theo mỗi dòng mô tả một lát cắt: $x \ y \ t \ k$ có nghĩa là cắt hình chữ nhật $x \times y$, $t = 0$ nghĩa là cắt cạnh bằng x thành k và $x - k$, $t = 1$ nghĩa là cắt cạnh y thành k và $y - k$

Chỉ cần đưa ra một phương án cắt tốt nhất tùy ý.

Hạn chế

- $1 \leq W \leq 600, 1 \leq H \leq 600, 0 < N \leq 200, 1 \leq W_i \leq W$, and $1 \leq H_i \leq H$.
- Có 50% số test ứng với $W \leq 20, H \leq 20$ và $N \leq 5$.

Ví dụ

stdin	stdout
21 11	10 17
4	21 11 1 2
10 4	21 9 1 4
6 2	21 5 0 7
7 5	14 5 0 7
15 10	21 4 0 1
	20 4 0 10
	1 4 1 1
	1 3 1 1
	1 2 1 1
	21 2 0 1
	20 2 0 1
	19 2 0 1
	18 2 0 6
	12 2 0 6
	1 2 1 1
	1 2 1 1
	1 2 1 1

Bài 10. Xếp hàng — LQUEUE

An là nhân viên bảo vệ ngân hàng. Hôm nay công việc khá nhàn rỗi nên An bắt đầu quan sát hàng những người đang đứng đợi trước quầy phục vụ. Ban đầu trong hàng chỉ có n người. An đánh số những người trong hàng theo thứ tự bắt đầu từ 0. Như vậy số thứ tự của mỗi người chính bằng số người đứng trước họ trong hàng đợi.

An có khả năng đánh giá tâm trạng người khác rất tốt. Tâm trạng của người thứ i được An mô tả bởi số nguyên không âm a_i . An cho rằng tâm trạng của người này tốt nếu $a_i \geq x$. Ngược lại, nếu $a_i < x$ thì tâm trạng của người này không tốt.

Hàng đợi thường có người mới tới xếp hàng và có người rời khỏi hàng sau khi được phục vụ xong. Nếu có người mới tới, An sẽ ngay lập tức đánh giá tâm trạng của người đó và tâm trạng của người này không thay đổi theo thời gian.

An đặt ra câu hỏi thú vị: tại thời điểm nào đó, An chọn một người trong hàng đợi và muốn đếm xem có bao nhiêu người có tâm trạng tốt hiện đang đứng trước anh ta trong hàng đợi.

Hãy giúp An!

Dữ liệu vào

Dòng đầu tiên chứa hai số nguyên n, x ($1 \leq n \leq 100\,000, 0 \leq x \leq 10^9$).

Dòng tiếp theo chứa n số nguyên a_i mô tả tâm trạng của n người trong hàng đợi ($0 \leq a_i \leq 10^9$).

Dòng thứ ba chứa số nguyên m ($1 \leq m \leq 100\,000$) là số sự kiện xảy ra đối với hàng đợi.

Trong m dòng tiếp theo mô tả các sự kiện xảy ra đối với hàng đợi. Mỗi sự kiện được mô tả bởi 1 trong 3 trường hợp sau:

- 1 a ($0 \leq a \leq 10^9$) — có người vừa tới xếp cuối hàng với tâm trạng là a .
- 2 — người đầu hàng (có số thứ tự là 0) rời khỏi hàng. Khi đó An sẽ giảm số thứ tự của tất cả những người trong hàng xuống 1 đơn vị.
- 3 i — An muốn biết, tại thời điểm này có bao nhiêu người có trạng thái tốt đứng trước người thứ i .

Đảm bảo rằng các mô tả sự kiện đều chính xác: nếu hàng đợi rỗng thì sự kiện dạng 2 không được thực hiện; số người trong hàng đợi luôn lớn hơn i trong các sự kiện dạng 3.

Kết quả

Đối với mỗi sự kiện dạng 3, hãy ghi ra một dòng chứa số lượng người có tâm trạng tốt hiện đang đứng trước người được chỉ định số thứ tự trong sự kiện.

Ví dụ

test	answer
1 2	0
3	1
5	2
1 2	
1 1	
3 0	
3 1	
3 2	
2 2	0
1 2	0
7	0
3 0	0
3 1	1
2	
3 0	
1 3	
3 0	
3 1	

Bài 11. Giao hàng — SHIPCOUNT

Tại kho hàng (điểm 0), điều phối viên phải lập lộ trình vận chuyển hàng hoá cho K xe khác nhau đến N khách hàng (điểm $1, \dots, N$). Lộ trình của mỗi xe sẽ xuất phát từ kho và đi đến 1 số khách hàng nào đó (mỗi khách hàng đúng 1 lần) và quay về kho. Mỗi khách hàng chỉ thuộc về đúng 1 lộ trình của 1 xe nào đó. Thứ tự các khách hàng trên mỗi lộ trình là quan trọng, ví dụ lộ trình $0 \rightarrow 1 \rightarrow 3 \rightarrow 0$ và lộ trình $0 \rightarrow 3 \rightarrow 1 \rightarrow 0$ là hai lộ trình khác nhau. Có thể có xe không được sử dụng (không được lập lộ trình). Để tìm ra phương án tối ưu, điều phối viên quyết định dùng phương pháp liệt kê hết tất cả các phương án. Tuy nhiên, sau một hồi ngẫm nghĩ và thử, điều phối viên cảm thấy số lượng phương án có vẻ là rất lớn. Yêu cầu: Hãy giúp điều phối viên tính số lượng phương án có thể có.

Dữ liệu vào

Dữ liệu đầu vào bao gồm 1 dòng chứa 2 số nguyên dương K và N

Kết quả

Ghi ra một số nguyên là số dư trong phép chia số lượng phương án cho $10^9 + 7$.

Ví dụ

test	answer
2 2	6

Giải thích

Có tất cả 6 phương án lộ trình được liệt kê trong Bảng 1

Phương án 1	xe 1: $0 \rightarrow 1 \rightarrow 2 \rightarrow 0$	xe 2: 0
Phương án 2	xe 1: $0 \rightarrow 2 \rightarrow 1 \rightarrow 0$	xe 2: 0
Phương án 3	xe 1: $0 \rightarrow 1 \rightarrow 0$	xe 2: $0 \rightarrow 2 \rightarrow 0$
Phương án 4	xe 1: $0 \rightarrow 2 \rightarrow 0$	xe 2: $0 \rightarrow 1 \rightarrow 0$
Phương án 5	xe 1: 0	xe 2: $0 \rightarrow 1 \rightarrow 2 \rightarrow 0$
Phương án 6	xe 1: 0	xe 2: $0 \rightarrow 2 \rightarrow 1 \rightarrow 0$

Bảng 1: Các phương án lộ trình với 2 xe và 2 khách hàng

Hạn chế

- Subtask 1: $K, N \leq 10$
- Subtask 2: $K, N \leq 200$
- Subtask 3: $K, N \leq 2000$

Bài 12. Điểm bán lẻ — RETOUT

Một công ty phân phối hàng hóa đến M điểm bán lẻ $1, 2, \dots, M$. Có N chi nhánh $1, 2, \dots, N$, chi nhánh i có a_i nhân viên bán hàng. Công ty phải giao M điểm bán lẻ cho N chi nhánh sao cho mỗi chi nhánh chịu trách nhiệm phân phối hàng hoá cho một số điểm bán lẻ, mỗi điểm bán lẻ do đúng một chi nhánh phân phối. Để cân bằng giữa các nhân viên bán hàng, số điểm bán lẻ được giao cho mỗi chi nhánh i phải là số dương và chia hết cho a_i .

Yêu cầu: Hãy tính tổng số Q các cách gán như vậy.

Ví dụ, $N = 2, M = 20, a_1 = 3, a_2 = 2$. Có 3 cách:

- Chi nhánh 1 được chỉ định cho 6 điểm bán lẻ, chi nhánh 2 được chỉ định cho 14 điểm
- Chi nhánh 1 được chỉ định cho 12 điểm bán lẻ, chi nhánh 2 được chỉ định cho 8 điểm
- Chi nhánh 1 được chỉ định cho 18 điểm bán lẻ, chi nhánh 2 được chỉ định cho 2 điểm

Dữ liệu vào

- Dòng 1: N và M
- Dòng 2: N số nguyên dương a_1, \dots, a_N

Kết quả

Ghi ra duy nhất một số nguyên Q là phần dư của số cách tìm được trong phép chia cho $(10^9 + 7)$

Ví dụ

test	answer
2 20 3 2	3

Hạn chế

- Subtask 1: $N \leq 100, M \leq 500$
- Subtask 2: $N \leq 10^6, M \leq 10^{18}, a_i = 1, \forall i$
- Subtask 3: $N \leq 1000, M \leq 5000$.