

TỔNG ƯỚC SỐ

Với mọi số tự nhiên m ($m \geq 2$) đều có ít nhất hai ước dương khác nhau đó là 1 và chính nó. Ta kí hiệu $s(m)$ là tổng của hai ước dương nhỏ nhất (khác nhau) của m .

Ví dụ: $s(3) = 1+3 = 4$; $s(8) = 1+2 = 3$.

Cho dãy số nguyên dương a_1, a_2, \dots, a_n và cặp chỉ số i, j ($1 \leq i \leq j \leq n$).

Yêu cầu: Tính tổng $s(a_i) + s(a_{i+1}) + \dots + s(a_j)$.

Dữ liệu vào từ file văn bản **SUMDIV.INP** như sau:

- Dòng 1 ghi hai số nguyên dương n và T tương ứng là số số hạng của dãy và số testcase.
- Dòng 2 ghi n số nguyên dương a_1, a_2, \dots, a_n .
- T dòng tiếp theo, mỗi dòng là một testcase tương ứng là một cặp chỉ số i, j ($1 \leq i \leq j \leq n$).

Kết quả ghi ra file văn bản **SUMDIV.OUT** gồm T dòng, mỗi dòng là kết quả của testcase tương ứng.

Ví dụ:

SUMDIV.INP	SUMDIV.OUT
5 3	10
2 3 8 9 100	4
1 3	10
4 4	
3 5	

Giới hạn:

- 30% số test ứng với $n \leq 100$ và $T \leq 10$, $2 \leq a_i \leq 10^7$;
- 30% số test khác ứng $n \leq 100000$, $T \leq 1000$ và $2 \leq a_i \leq 10^4$;
- 40% số test còn lại ứng với $n \leq 100000$, $T \leq 10^4$; $2 \leq a_i \leq 10^7$.

THAO TÁC BẢNG

Cho một bảng các số nguyên dương gồm n dòng m cột ($1 \leq n, m \leq 500$). Một thao tác trên bảng được định nghĩa như sau: chọn một số bất kì trong bảng và tăng giá trị của số đó lên 1. Một số có thể được chọn để thực hiện thao tác trên nhiều hơn 1 lần.

Người ta định nghĩa một bảng là có tính nguyên tố nếu thỏa mãn một trong hai điều kiện sau:

- Trong bảng tồn tại một hàng bất kì mà tất cả các số trong hàng đều là số nguyên tố.
- Trong bảng tồn tại một cột bất kì mà tất cả các số trong cột đều là số nguyên tố.

Yêu cầu: Tìm số thao tác biến đổi ít nhất để bảng ban đầu thành bảng có tính nguyên tố.

Dữ liệu vào cho từ tệp **BNT.INP** gồm:

- Dòng 1 gồm 2 số n, m tương ứng là số dòng và số cột của bảng
- N dòng sau, mỗi dòng gồm m số nguyên dương có giá trị $\leq 10^6$ là các số trong bảng ban đầu. Các số trên một dòng cách nhau một dấu cách.

Dữ liệu ra ghi ra tệp **BNT.OUT** gồm:

- Một số duy nhất là số thao tác ít nhất cần biến đổi để bảng ban đầu có tính nguyên tố.

Ví dụ:

BNT.INP	BNT.OUT
1 1 14	3

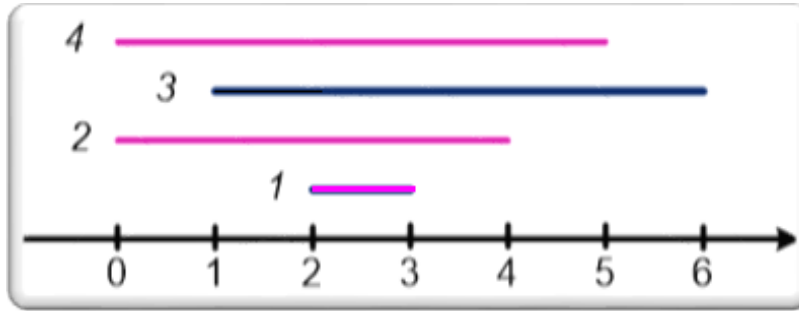
Ràng buộc:

- 50% số test ứng với 50% số điểm của bài có $n \leq 50$

ĐOẠN THẲNG

Cho n đoạn thẳng $[a_i, b_i]$ nằm trên một trục đường thẳng ($a_i < b_i$, $|a_i|, |b_i| \leq 10^6$, $i = 1 \div n$, $1 \leq n \leq 10^5$). Nói đoạn thẳng thứ i nằm trực tiếp trong đoạn thẳng thứ j (hay đoạn thẳng j trực tiếp chứa đoạn i) nếu nó thuộc hoàn toàn đoạn thẳng thứ j , tức là $a_j \leq a_i$ và $b_i \leq b_j$.

ui



Yêu cầu: Tìm một dãy dài nhất các đoạn thẳng khác nhau trong dãy ban đầu sao cho từ đoạn thứ hai trong dãy tìm được, mỗi đoạn chứa trực tiếp một đoạn khác.

Dữ liệu vào từ tệp văn bản **DOANTHANG.INP** gồm:

- Dòng đầu tiên chứa số nguyên n
- Dòng thứ i trong n dòng sau chứa 2 số nguyên a_i và b_i .

Dữ liệu ra ghi ra tệp văn bản **DOANTHANG.OUT** gồm:

- Một số nguyên duy nhất là độ dài dãy tìm được.

Ví dụ:

DOANTHANG.INP	DOANTHANG.OUT	Giải thích
4 2 3 0 4 1 6 0 5	3	<i>Giải thích ví dụ: Dãy con tìm được gồm 3 đoạn: (2,3); (0,4) và (0,5).</i>
6	3	<i>Giải thích ví dụ: Dãy con tìm</i>

2 10		được gồm 3 đoạn: (7,8); (6,9) và (2,10).
6 9		
1 2		
7 8		
1 8		
8 10		

Ràng buộc:

- Có 30% số test tương ứng 30% số điểm có $n \leq 20$
- Có 30% số test tương ứng 30% số điểm có $n \leq 10^4$
- Có 40% số test tương ứng 40% số điểm có $n \leq 10^5$

TẬP SỐ

Cho S là tập hợp các số có $n(n \geq 2)$ chữ số thỏa mãn:

- Hai chữ số bất kì liên tiếp luôn có ít nhất một chữ số là số nguyên tố.
- Tổng hai chữ số bất kì liên tiếp lớn hơn hoặc bằng 8.

Ví dụ: với $n = 2$, tập hợp $S = \{17, 26, 27, 28, 29, 35, 36, 37, 38, 39, 45, 47, 53, 54, 55, 56, 57, 58, 59, 62, 63, 65, 67, 71, 72, 73, 74, 75, 76, 77, 78, 79, 82, 83, 85, 87, 92, 93, 95, 97\}$. Như vậy tập S có 40 phần tử.

Yêu cầu: Gọi k là số phần tử của tập S . Tính k .

Dữ liệu cho trong file văn bản **NUMSET.INP** gồm số nguyên dương n .

Kết quả ghi ra file văn bản **NUMSET.OUT** Là số dư khi chia k cho 10^9 .

Ví dụ:

NUMSET.INP	NUMSET.OUT
2	40

Giới hạn:

- 10% số test ứng với $n \leq 7$;

- 10% số test khác ứng với $n \leq 10^6$;
- 80% số test còn lại ứng với $n \leq 10^{18}$.

HÀNH TRÌNH

Ở một nơi rất xa có một quần đảo gồm có n hòn đảo được đánh số từ 1 đến n . Có m tuyến đường biển để đi lại hai chiều giữa các đảo, tuyến đường thứ i nối hai đảo khác nhau u_i, v_i ($1 \leq u_i, v_i \leq n$) và mất t_i đơn vị thời gian để đi hết nó.

Để đi lại giữa các đảo, chúng ta sử dụng tàu thủy có độ dày thân tàu là k cm. Mỗi khi tàu đi hết tuyến đường i , thân tàu bị làm mòn h_i cm. Như vậy, để đi từ đảo A đến đảo B với tàu thủy có độ dày thân tàu là k cm bằng các tuyến đường thì tổng độ làm mòn thân tàu trên các tuyến đường mà nó đi qua phải nhỏ hơn k .

Yêu cầu: cho trước hai hòn đảo A và B ($1 \leq A, B \leq n, A \neq B$). Tìm đường đi cho con tàu khi đi từ đảo A tới đảo B có tổng thời gian là ít nhất (thỏa mãn có tổng độ làm mòn thân tàu khi đi trên đường đi này nhỏ hơn k).

Dữ liệu vào cho trong tệp **HANHTRINH.INP** gồm:

- Dòng thứ nhất gồm 3 số nguyên k, n, m ($1 \leq k \leq 200, 2 \leq n \leq 10000, 1 \leq m \leq 100000$), mỗi số cách nhau một dấu cách.
- Dòng thứ i trong m dòng tiếp theo, mỗi dòng chứa 4 số nguyên u_i, v_i, t_i và h_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i, 1 \leq t_i \leq 10^5, 0 \leq h_i \leq 200$) thể hiện đi lại trên tuyến đường thứ i nối hai đảo u_i, v_i mất t_i đơn vị thời gian và độ làm mòn thân tàu khi đi trên nó là h_i , mỗi số cách nhau một dấu cách.
- Dòng cuối cùng chứa hai số nguyên A và B ($1 \leq A, B \leq n, A \neq B$)

Dữ liệu ra ghi ra tệp **HANHTRINH.OUT** gồm:

- Một số nguyên duy nhất là thời gian ít nhất khi đi từ đảo A tới đảo B thỏa mãn yêu cầu trên hoặc đưa ra -1 nếu không có cách đi từ đảo A tới đảo B.

Ví dụ:

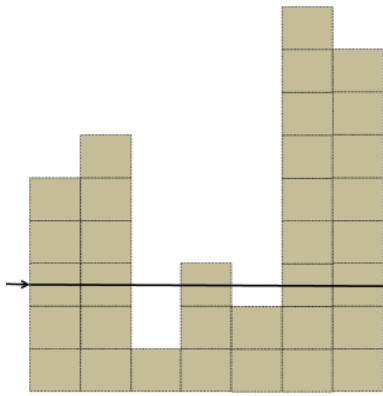
HANHTRINH.INP	HANHTRINH.OUT
10 4 7	7
1 2 4 4	
1 3 7 2	
3 1 8 1	
3 2 2 2	
4 2 1 6	
3 4 1 1	
1 4 6 12	
1 4	

Ràng buộc:

- Có 50% số test tương ứng 50% số điểm có $K = 1, N \leq 2000$
- Có 25% số test tương ứng 25% số điểm có $N \leq 2000$
- Có 25% số test tương ứng 25% số điểm có $N \leq 10000$

KHỐI BÊ TÔNG

Một công trình xây dựng có n thanh bờ tường chiều rộng bằng 1, chiều cao a_i được dựng kề nhau. Các thanh bê tông được đánh số từ 1 đến n , khối bê tông thứ i có chiều cao a_i (a_i là các số nguyên dương). Lúc đầu các thanh bê tông này tạo thành một khối, nhà thầu xây dựng muốn dùng một tấm che chiều cao h (h là một số thực) để che các thanh bê tông từ dưới đất lên và các thanh bê tông cao hơn h sẽ nhìn thấy được, các thanh bê tông có độ cao nhỏ hơn hoặc bằng h sẽ không nhìn thấy được. Các thanh nhìn thấy được liên tiếp nhau sẽ tạo thành một khối. Nhà thầu xây dựng muốn nhờ bạn tính độ cao h bằng bao nhiêu để số khối bê tông nhìn thấy là nhiều nhất.



Ví dụ: 5 6 1 3 2 9 8

Có 7 thanh bê tông với các chiều cao tương ứng là:

5, 6, 1, 3, 2, 9, 8.

- Ban đầu các thanh này tạo thành một khối.
- Nếu $h = 1$ thì ta sẽ thấy được 2 khối (khối thứ nhất gồm hai cột bên trái; khối thứ 2 gồm 4 cột bên phải).
- Nếu $h = 2$ thì ta thấy được 3 khối.
- Nếu $h = 3$ thì ta thấy được 2 khối.

Với ví dụ này thì số khối có thể tạo ra nhiều nhất là

Dữ liệu cho trong file **BETONG.INP** như sau:

- Dòng đầu ghi số nguyên dương n .
- n dòng tiếp theo, mỗi dòng ghi một số nguyên a_i chiều cao của thanh bê tông thứ i .

Kết quả ghi ra file **BETONG.OUT** gồm một số duy nhất là số khối bê tông nhiều nhất có thể tạo ra.

Ví dụ:

BETONG.INP	BETONG.OUT
7	3
5	
6	
1	
3	
2	
9	
8	

Giới hạn:

- 20% số test ứng với $n \leq 10^3, a_i \leq 10^9$;
- 20% số test khác ứng với $n \leq 10^5$ và $a_i \leq 20$;
- 20% số test khác ứng với $n \leq 10^5, a_i \neq a_j$ với $i \neq j, a_i \leq 10^9$;
- 20% số test khác ứng với $n \leq 10^6, a_i \neq a_j$ với $i \neq j, a_i \leq 10^9$;
- 20% số test còn lại ứng với $n \leq 10^6, a_i \leq 10^9$;

