

1. C – GCD.*

Cho dãy số gồm N số nguyên A_1, A_2, \dots, A_N .

Bạn sẽ chọn một trong số chúng và thay thế nó bằng một số nguyên bạn chọn trong khoảng từ 1 đến 10^9 (bằng 10^9), có thể giống với số nguyên được viết ban đầu.

Yêu cầu: Tìm ước chung lớn nhất có thể có của N số nguyên trên sau khi bạn di chuyển.

Ràng buộc:

- Tất cả các giá trị trong đầu vào là số nguyên.
- $2 \leq N \leq 10^5$; $1 \leq A_i \leq 10^9$

Dữ liệu: Vào từ file GCD.INP gồm:

- Dòng đầu tiên chứa số nguyên N .
- Dòng tiếp theo chứa N số nguyên $A_1 A_2 \dots A_N$ tương ứng, các giá trị cách nhau 1 khoảng trắng.

Kết quả: Ghi ra file GCD.OUT: ước số chung lớn nhất có thể của N số nguyên sau khi bạn di chuyển.

Ví dụ:

GCD.INP	GCD.OUT
3	2
7 6 8	

Giải thích: Nếu chúng ta thay 7 bằng 4, ước chung lớn nhất của ba số nguyên trên bảng đen sẽ là 2, là giá trị lớn nhất có thể.

2. Karate (Karate.*)

Giải thi đấu võ Karate tại Thái Nguyên, cả nam lẫn nữ có tất cả n vận động viên được xếp thành một hàng ngang đánh số từ 1 đến n . Vận động viên thứ i có năng lực chiến đấu là một số nguyên a_i .

Trưởng ban tổ chức muốn chọn ra một số vận động viên đại diện cho tỉnh mình để tham gia giải đấu toàn quốc nhưng ông ta chưa biết chọn thế nào, đành cho các vận động viên của mình đấu với nhau theo nguyên tắc như sau:

- Chỉ thi đấu với nhau khi cùng giới tính.
- Mỗi vận động viên sẽ được đấu với các vận động viên đứng trước họ từ gần đến xa, đến khi nào thua trận thì thôi. (Vận động viên thứ i chỉ đấu thắng vận động viên thứ j nếu $a_i > a_j$).

Yêu cầu: Với mỗi vận động viên hãy cho biết họ thắng bao nhiêu vận động viên khác theo nguyên tắc trên.

Input: Đọc từ tệp karate.inp có cấu trúc như sau:

- Dòng 1: Chứa số nguyên dương n là số lượng võ sỹ nhà để mèn.
- n dòng sau: Mỗi dòng chứa hai số nguyên, dòng thứ $i+1$ chứa số nguyên a_i, b_i là năng lực chiến đấu và giới tính của võ sỹ i ($0 < a_i \leq 10^9, 0 \leq b_i \leq 1$).

Output: Ghi vào tệp karate.out gồm n số nguyên số thứ i là số lượng vận động viên mà vận động viên thứ i sẽ được chiến đấu với vận động viên i thắng trận.

Ví dụ:

karate.inp	karate.out
10	0 0 1 2 0 0 0 0 1 2
9 0	
18 1	
11 0	
12 0	
3 0	
12 1	
2 0	
2 1	
7 1	
6 0	

3. Dãy con có tổng chia hết cho n

Tên file chương trình:

‘SUMDIVN.???’

Cho số nguyên dương n và dãy A gồm n số nguyên dương a_1, a_2, \dots, a_n có thể tạo ra một dãy con gồm các phần tử liên tiếp mà tổng các phần tử chia hết cho n hay không?

Yêu cầu: Hãy tìm dãy con nhỏ nhất chia hết cho n

Dữ liệu vào: Từ file SUMDIVN.INP

- Dòng đầu tiên của mỗi test là một số n ($1 \leq n \leq 10^4$),
- Dòng tiếp theo là các số nguyên dương cách nhau một khoảng trắng

Dữ liệu ra: Ghi vào file SUMDIVN.OUT dãy con tìm được, Nếu không tìm được ghi -1;

Ví dụ:

SUMDIVN.INP	SUMDIVN.OUT
1	1
1	4 2
3	
1 4 2	

4. Đoạn con có tổng lớn nhất trình: 'DOANCON2.???'

Tên File chương

Cho một dãy số nguyên a_1, a_2, \dots, a_n .

Yêu cầu: Hãy tìm đoạn con dài nhất có tổng lớn nhất.

Dữ liệu vào từ file 'DOANCON2.INP':

- Dòng đầu là số nguyên dương n và k ($n \leq 1000$; $k \leq 50$)
- Dòng thứ 2 là các phần tử của dãy (phạm vi mỗi phần tử từ $-2 \cdot 10^9$ đến $2 \cdot 10^9$)

Kết quả ghi vào file 'DOANCON2.OUT' gồm 2 dòng:

- Dòng đầu ghi tổng lớn nhất tìm được.

Ví dụ:

DOANCON2.INP	DOANCON2.OUT
6	11
3 -4 5 6 -8 7	

5. Counting Rooms- cntRooms.*

Cho một bản đồ của một tòa nhà, và nhiệm vụ của bạn là đếm số phòng của nó. Kích thước của bản đồ là $n \times m$ ô vuông và mỗi ô vuông tương ứng là sàn hoặc tường. Bạn có thể đi sang trái, phải, lên và xuống qua các ô vuông trên sàn.

Dữ liệu: vào từ file **CNTROOMS.INP**

- Dòng đầu tiên có hai số nguyên n và m : chiều cao và chiều rộng của bản đồ.
- Tiếp theo có n dòng gồm m ký tự mô tả bản đồ. Mỗi ký tự là một trong hai loại: nếu dấu . là sàn, hoặc dấu # là tường.

Kết quả: ghi ra file **CNTROOMS.OUT**: In ra một số nguyên: số phòng.

Ràng buộc:

- $1 \leq n, m \leq 1000$

Ví dụ:

CNTROOMS.INP	CNTROOMS.OUT
5 8 ##### #..#...# ####.#.# #..#...# #####	3

6. ĐƯỜNG NGUYÊN TỐ - DUONGNT.???

Cho hai số nguyên tố khác nhau có bốn chữ số. Người ta cho rằng hoàn toàn có thể biến đổi từ số này thành số kia sau một số bước theo quy tắc: Tại mỗi bước ta chỉ thay đổi một chữ số trong số trước đó sao cho số tạo được trong mỗi bước đều là một số nguyên tố có bốn chữ số. Một cách biến đổi như vậy gọi là một “đường nguyên tố”.

Bài toán đặt ra là với một cặp số nguyên tố đầu vào, hãy tính ra số bước của đường nguyên tố ngắn nhất. Giả sử đầu vào là hai số 1033 và 8179 thì đường nguyên tố ngắn nhất sẽ có độ dài là 6 với các bước chuyển là: 1033 -> 1733 -> **3733** -> **3739** -> **3779** -> **8779** -> 8179

Dữ liệu vào: Từ tệp văn bản DUONGNT.INP

- Gồm 2 số nguyên tố u và v, mỗi số có đúng 4 chữ số cơ nghĩa.

Dữ liệu ra: ghi vào tệp văn bản DUONGNT.OUT

- Số bước của đường nguyên tố ngắn nhất của 2 số nguyên u và v.

Ví dụ:

DUONGNT.INP	DUONGNT.OUT
1033 8179	6

