

BÁN HÀNG

Chip quyết định mở một cửa hàng bán kẹo và lập ra kế hoạch bán hàng đặc biệt tên là Chip Selling Plan (CSP). Giá hàng của Chip có vô hạn gói kẹo đánh số bằng các số nguyên dương từ 1 trở đi. Chip đã bí mật giấu m quà tặng vào các gói kẹo mang số hiệu $b_1, b_2, \dots, b_m$.

Trong ngày, có n khách hàng đánh số từ 1 tới n theo thứ tự đến mua hàng. Khi một khách hàng thứ i vào cửa hàng, Chip hỏi số gói kẹo người đó muốn mua (a_i) sau đó chọn đúng a_i gói kẹo còn lại trên giá có số hiệu nhỏ nhất chia hết cho a_i để bán cho người khách đó.

Ví dụ:

Khách hàng thứ nhất đến mua $a_1 = 4$ gói kẹo, Chip sẽ bán cho các gói số hiệu 4, 8, 12 và 16

Khách hàng thứ hai đến mua $a_2 = 2$ gói kẹo, Chip sẽ bán tiếp các gói số hiệu 2 và 6.

Cuối ngày, Chip muốn biết có bao nhiêu gói kẹo chứa quà tặng đã được bán. Việc bóc các gói kẹo để kiểm kê tỏ ra rất mất thời gian, hãy giúp Chip tính con số đó.

Dữ liệu: Vào từ file văn bản GIFTS.INP

Dòng 1 chứa số nguyên dương $m \leq 10^6$ là số quà tặng

Dòng 2 chứa m số nguyên dương $b_1, b_2, \dots, b_m$ hoàn toàn phân biệt là số hiệu những gói kẹo chứa quà tặng
($\forall i: b_i \leq 10^6$)

Dòng 3 chứa số nguyên dương $n \leq 10^6$ là số khách hàng

Dòng 4 chứa n số nguyên dương $a_1, a_2, \dots, a_n$ là số kẹo muốn mua của các khách hàng ($\forall i: a_i \leq 10^6$)

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách

Kết quả: Ghi ra file văn bản GIFTS.OUT một số nguyên duy nhất là số gói kẹo chứa quà tặng đã được bán

Ví dụ

GIFTS.INP	GIFTS.OUT
4	3
1 6 8 16	
3	
4 2 4	

KIỆN

Một đàn kiến phân bố trên đoạn $[0; L]$. Đàn kiến có hai kho rỗng ở điểm 0 và điểm L . Có tất cả n con kiến đánh số từ 1 tới n . Con kiến thứ i đứng ở tọa độ x_i ($0 < x_i < L$) và đang công một lượng hàng có khối lượng là w_i . Không có hai con kiến nào đứng cùng vị trí.

Mục đích của các con kiến là chuyển hàng về một trong hai kho. Tất cả các con kiến ban đầu đều di chuyển với vận tốc như nhau: 1 đơn vị độ dài trong 1 đơn vị thời gian. Tuy nhiên hướng di chuyển có thể là về phía kho 0 hoặc về phía kho L tùy theo ý định ban đầu của mỗi con kiến.

Vì đoạn đường rất hẹp nên để tránh giẫm lên nhau, khi hai con kiến va vào nhau trong quá trình di chuyển, chúng sẽ cùng đổi hướng đi ngược lại với vận tốc không đổi. Khi con kiến về một trong hai kho, nó sẽ đưa hàng vào kho và không di chuyển nữa.

Giả sử các con kiến bắt đầu di chuyển ở thời điểm 0, Kiến Chúa muốn kiểm soát tiến độ chuyển hàng tại các thời điểm $t_1, t_2, \dots, t_m$, hãy cho biết ở mỗi thời điểm t_k (tính cả thời điểm đó), tổng lượng hàng tại hai kho là bao nhiêu.

Dữ liệu: Vào từ file văn bản ANTMOVE.INP

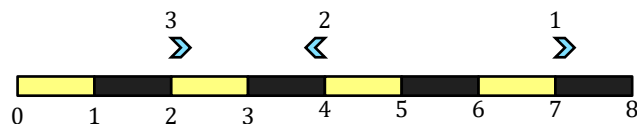
- ✿ Dòng 1 chứa ba số nguyên dương n, L, m ($1 \leq n, m \leq 10^5; n < L \leq 10^9$)
- ✿ n dòng tiếp theo, mỗi dòng chứa ba số nguyên x_i, w_i, d_i là thông tin ban đầu của một con kiến: x_i là vị trí con kiến, w_i là khối lượng hàng nó công ($1 \leq w_i \leq 10^9$), $d_i \in \{0, 1\}$, trong đó $d_i = 0$ cho biết con kiến đi về phía kho 0 còn $d_i = 1$ cho biết con kiến đi về phía kho L .
- ✿ Dòng cuối ghi m số nguyên $t_1, t_2, \dots, t_m$ ($0 \leq t_j \leq 10^9; \forall j: 1 \leq j \leq m$)

Kết quả: Ghi ra file văn bản ANTMOVE.OUT một dòng m số nguyên ứng với tổng lượng hàng đã về kho tại thời điểm $t_1, t_2, \dots, t_m$ theo đúng thứ tự đó

Các số trên một dòng của input/output được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ

ANTMOVE . INP	ANTMOVE . OUT
3 8 5	0 1 4 6 6
2 3 1	
4 2 0	
7 1 1	
0 2 4 6 8	



THỨ HẠNG

Trường học của giáo sư X có n lớp đánh số từ 1 tới n . Hàng tuần mỗi lớp được thống kê những việc làm tốt và những lỗi đã phạm để xếp thứ hạng thi đua.

Trong giờ tổng kết hàng tuần, hệ số thưởng phạt được đưa ra một cách tùy hứng bởi hai hệ số dương a, b . Cụ thể là nếu một lớp làm được x việc tốt và phạm y lỗi thì điểm thi đua của lớp đó sẽ được tính bởi công thức:

$$a \times x - b \times y$$

Thứ hạng thi đua của một lớp được tính bằng số lớp khác có điểm thi đua cao hơn lớp đó cộng thêm 1. Những lớp có thứ hạng tốt nhất (nhỏ nhất) sẽ được thưởng những chuyến dã ngoại vui vẻ, trong khi đó những lớp thứ hạng tệ nhất (lớn nhất) sẽ phải đi lao động công ích.

Chính vì sự hồi hộp trong buổi tổng kết hàng tuần nên các học sinh quyết định nhờ những thành viên đội tuyển tin học viết một chương trình hỗ trợ cộng đồng (Community Support Program – CSP) nhằm xác định thứ hạng tốt nhất và tệ nhất có thể của mỗi lớp trên tất cả các cặp hệ số dương (a, b) .

Dữ liệu: Vào từ file văn bản RANKING.INP

- ✿ Dòng 1 chứa số nguyên dương $n \leq 1000$
- ✿ n dòng tiếp theo, mỗi dòng chứa hai số nguyên x_i, y_i lần lượt là số việc tốt và số lỗi đã phạm của lớp thứ i ($\forall i: 0 \leq x_i, y_i \leq 10^6$)

Kết quả: Ghi ra file văn bản RANKING.OUT n dòng, dòng thứ i ghi hai số nguyên lần lượt là thứ hạng tốt nhất và thứ hạng tệ nhất có thể của lớp thứ i .

Các số trên một dòng của input/output file được phải ghi cách nhau bởi dấu cách

Ví dụ

RANKING.INP	RANKING.OUT
5	5 5
0 1000	2 3
1 100	2 3
1 100	2 4
2 200	1 1
100 0	