

BAO LỒI

Trên mặt phẳng với hệ trục tọa độ Descartes vuông góc Oxy cho n điểm đánh số từ 1 tới n , có thể có những điểm trùng nhau nhưng có ít nhất 3 điểm không thẳng hàng. Điểm thứ i có tọa độ (x_i, y_i) . Hãy tìm một đa giác lồi với diện tích nhỏ nhất mà miền giới hạn bởi đa giác (tính cả đường biên) chứa tất cả n điểm đã cho. (Đa giác lồi được định nghĩa là miền giới hạn bởi một đường gấp khúc khép kín không tự cắt có các đỉnh phân biệt và các góc nhỏ hơn 180 độ).

Dữ liệu: Vào từ file văn bản CONVEXHULL.INP

- ✿ Dòng 1 chứa số nguyên dương n ($3 \leq n \leq 10^5$)
- ✿ n dòng tiếp theo, dòng thứ i chứa hai số nguyên x_i, y_i có giá trị tuyệt đối không quá 10^9

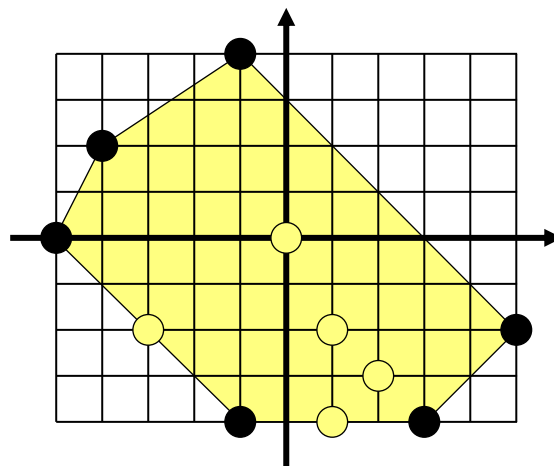
Kết quả: Ghi ra file văn bản CONVEXHULL.OUT

- ✿ Dòng 1 ghi số đỉnh (m) của đa giác tìm được
- ✿ Dòng 2 ghi diện tích đa giác tìm được với đúng 1 chữ số sau dấu chấm thập phân.
- ✿ m dòng tiếp theo, dòng thứ j ghi tọa độ đỉnh thứ j của đa giác tìm được theo thứ tự sau: Đỉnh trái nhất trong số những đỉnh thấp nhất của bao lồi được đánh số 1, các đỉnh còn lại được đánh số theo thứ tự tạo thành đa giác liệt kê theo chiều ngược với chiều kim đồng hồ.

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

Ví dụ

CONVEXHULL.INP	CONVEXHULL.OUT
11	6
-1 4	46.0
-4 2	-1 -4
-5 0	3 -4
0 0	5 -2
-3 -2	-1 4
1 -2	-4 2
5 -2	-5 0
2 -3	
-1 -4	
1 -4	
3 -4	



CHUYỂN NƯỚC

Các bé học sinh trường mầm non SuperKids tỏ ra say mê với các trò chơi đòi hỏi tư duy thuật toán chuyên nghiệp. Nhân dịp đến thăm trường, giáo sư X bày ra một trò chơi cho các bạn nhỏ tại đây.

Ban đầu, người chơi được cho n thùng nước đánh số từ 1 tới n . Thùng thứ i có a_i lít nước. Người chơi được quyền mức một lượng nước bất kỳ từ một thùng chuyển sang thùng liền sau (chuyển từ thùng i sang thùng $i + 1$ với i tùy chọn thỏa mãn $1 \leq i < n$). Năng lượng tiêu tốn cho thao tác này đúng bằng lượng nước được chuyển (có thể không phải là số nguyên)

Nhiệm vụ của người chơi là phải làm cho lượng nước trong các thùng sắp xếp thứ tự không giảm, tức là:

$$a_1 \leq a_2 \leq \dots \leq a_n$$

Yêu cầu: Hãy tìm phương án chơi sao cho tổng năng lượng tiêu tốn là ít nhất

Dữ liệu: Vào từ file văn bản WATERMOV.INP

- ✿ Dòng 1 chứa số nguyên dương $n \leq 10^6$
- ✿ Dòng 2 chứa n số nguyên không âm a_1, a_2, \dots, a_n ($\forall i: a_i \leq 10^6$) cách nhau bởi dấu cách

Kết quả: Ghi ra file văn bản WATERMOV.OUT một số thực duy nhất với 1 chữ số sau dấu chấm thập phân là tổng năng lượng tiêu tốn nếu các bé chơi theo phương án của bạn

Ví dụ

WATERMOV.INP	WATERMOV.OUT
6 1 3 0 0 3 0	4.5

Giải thích:

Ta sẽ chuyển nước để được lượng nước trong các thùng là 1.0,1.0,1.0,1.0,1.5,1.5

Chuyển 2 lít từ thùng 2 sang thùng 3

Chuyển 1 lít từ thùng 3 sang thùng 4

Chuyển 1.5 lít từ thùng 5 sang thùng 6

THỬA ĐẤT LỚN NHẤT

Bờm lại thắng Phú ông trong một cuộc đánh cược và theo thỏa thuận từ trước, Phú ông buộc phải cho Bờm một thửa đất trong phần đất đai rộng lớn của mình. Bản đồ phần đất của Phú ông có thể coi là một mặt phẳng với hệ trục tọa độ Descartes vuông góc Oxy trên đó đánh dấu n ($n \geq 3$) cột mốc hoàn toàn phân biệt và không đồng thời thẳng hàng, cột mốc thứ i có tọa độ (x_i, y_i) . Bờm được chọn ba cột mốc trong số đó để nhận thửa đất có dạng hình tam giác có ba đỉnh là vị trí ba cột mốc được chọn.

Yêu cầu: Hãy giúp Bờm chọn ba cột mốc để nhận được thửa đất có diện tích lớn nhất.

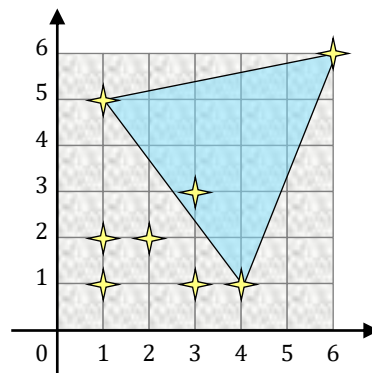
Dữ liệu: Vào từ file văn bản TRILAND.INP

- ✿ Dòng 1 chứa số nguyên dương n ($3 \leq n \leq 3000$)
- ✿ n dòng tiếp theo, dòng thứ i chứa hai số nguyên x_i, y_i ($\forall i: |x_i|, |y_i| \leq 10^9$) cách nhau bởi dấu cách

Kết quả: Ghi ra file văn bản TRILAND.OUT diện tích của thửa đất Bờm sẽ nhận theo phương án tìm được. Diện tích này phải ghi dưới dạng số thực với đúng 1 chữ số sau dấu chấm thập phân.

Ví dụ:

TRILAND.INP	TRILAND.OUT
8 1 1 1 2 1 5 2 2 3 1 3 3 4 1 6 6	11.5
4 1 1 1 5 3 3 4 1	6.0



ĐƯỜNG PHỐ MÙA LỄ HỘI

Con đường Vạn Hoa dài m km mà giáo sư X thường đi ngắm cảnh trong kỳ nghỉ đang vào mùa lễ hội, ngày nào cũng có m lễ hội trên đường đánh số từ 1 tới m . Lễ hội thứ i diễn ra tại điểm cách đầu đường i km và tiến hành từ đầu ngày (thời điểm 0) cho tới hết thời điểm t_i trong ngày, trong thời gian lễ hội tổ chức không xe nào được đi qua điểm diễn ra lễ hội mà phải đợi tới khi lễ hội kết thúc mới được đi qua.

Giáo sư X không quan tâm lắm tới các lễ hội mà ông chỉ đam mê tốc độ trong khung cảnh thiên nhiên hoang dã, trong mỗi ngày đi dạo (bằng mô-tô) từ đầu tới cuối con đường Vạn Hoa, ông muốn tính toán xem mình có thể đi với tốc độ tối đa là bao nhiêu mà không phải dừng lại chờ bất cứ lễ hội nào.

Yêu cầu: Cho biết tốc độ tối đa có thể của giáo sư X trong mỗi ngày, biết rằng kỳ nghỉ của giáo sư diễn ra trong n ngày và vào ngày thứ j giáo sư bắt đầu đi vào thời điểm s_j

Dữ liệu: Vào từ file văn bản RIDER.INP

- Dòng 1 chứa số nguyên dương $m \leq 10^5$
- Dòng 2 chứa m số nguyên dương $t_1, t_2, \dots, t_m \leq 10^{12}$ cách nhau ít nhất một dấu cách
- Dòng 3 chứa số nguyên dương $n \leq 10^5$
- Dòng 4 chứa n số nguyên không âm s_1, s_2, \dots, s_n cách nhau ít nhất một dấu cách ($\forall j: s_j < \max_{i=1,2,\dots,m} \{t_i\}$)

Kết quả: Ghi ra file văn bản RIDER.OUT n dòng, dòng thứ j ghi tốc độ tối đa (số km/1 đơn vị thời gian) của giáo sư trong ngày thứ j dưới dạng một số thực làm tròn lấy đúng 12 chữ số sau dấu chấm thập phân

RIDER.INP	RIDER.OUT
4	0.333333333333
3 5 6 1	1.000000000000
3	3.000000000000
0 3 5	

ĐỌC TRUYỆN

An có đủ bộ truyện Doraemon mới tái bản gồm n tập đánh số từ 1 tới n , tập thứ i có độ dày là t_i trang. Vì các bạn trong lớp đều muốn đọc nên An muốn tạo ra một kế hoạch đọc truyện cho các bạn sao cho hợp lý nhất. Có m bạn muốn đọc truyện, họ phải bốc thăm và đánh số từ 1 tới m theo thứ tự từ người sẽ được đọc đầu tiên tới người sẽ được đọc sau cùng. Nếu một bạn có tốc độ đọc C giây/trang thì để đọc tập thứ i , bạn đó sẽ mất thời gian là $C \times t_i$. Tất cả các bạn đều muốn đọc bộ truyện theo đúng thứ tự từ tập 1 tới tập n , hơn thế nữa khi đọc xong một tập, họ muốn có thể đọc ngay tập tiếp theo mà không mất thời gian chờ đợi:

- Đầu tiên An cho bạn thứ nhất mượn từng tập, đọc xong tập nào trả lại ngay cho An tập đó và mượn tập kế tiếp...
- Tới bạn thứ hai, An cũng cho mượn theo cách như vậy. Nhưng nhờ biết tốc độ đọc của bạn thứ nhất, An phải tính toán thời điểm bắt đầu cho bạn thứ hai đọc tập 1 để khi bạn thứ hai đọc xong mỗi tập i thì tập $i + 1$ đã được bạn thứ nhất trả để An cho bạn thứ hai mượn.
- Tương tự như vậy với các bạn thứ 3, 4, ..., n . An phải tính toán thời điểm mỗi người bắt đầu đọc tập 1 để không có bạn nào phải chờ đợi tập kế tiếp mỗi khi đọc xong một tập...

Yêu cầu: Biết tốc độ đọc của m bạn là c_1, c_2, \dots, c_m . Tính thời điểm sớm nhất mà bạn cuối cùng đọc xong bộ truyện. Biết rằng người 1 bắt đầu đọc từ thời điểm 0.

Ví dụ với $n = 3$ tập, $m = 3$ bạn, $t = (1, 2, 1)$, $c = (10, 10, 2)$. Cách đọc kết thúc sớm nhất và các khoảng thời gian đọc truyện trong lịch có thể cho trong bảng sau:

	Tập 1 (số trang = 1)	Tập 2 (số trang = 2)	Tập 3 (số trang = 1)
Người đọc 1 (10)	0 ... 10	10 ... 30	30 ... 40
Người đọc 2 (10)	20 ... 30	30 ... 50	50 ... 60
Người đọc 3 (2)	54 ... 56	56 ... 60	60 ... 62

Dữ liệu: Vào từ file văn bản READERS.INP

- Dòng 1 chứa hai số nguyên dương $n, m \leq 10^5$
- Dòng 2 chứa n số nguyên dương $t_1, t_2, \dots, t_n \leq 10^5$
- Dòng 3 chứa m số nguyên dương $c_1, c_2, \dots, c_m \leq 10^5$

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

Kết quả: Ghi ra file văn bản READERS.OUT một số nguyên duy nhất là thời điểm người cuối cùng đọc xong bộ truyện

Ví dụ

READERS.INP	READERS.OUT
3 3 1 2 1 10 10 2	62