

Bài A. TWOLAYER

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

Huy đang muốn mua tặng Giang một món quà nhân ngày sinh nhật. Nơi Huy sống có n thành phố đánh số 1 đến n . Món quà mà Huy muốn mua, ở thành phố i được bán với giá c_i đồng. Hệ thống giao thông công cộng ở đây chủ yếu là tàu điện ngầm. Mỗi tuyến có hai định mức giá vé, là giá vé đi lần đầu của tuyến này và giá vé đi không phải lần đầu của tuyến này. Có m tuyến tàu điện như vậy, tuyến thứ i di chuyển hai chiều giữa thành phố u_i và v_i , hai định mức giá vé là w_i và l_i đồng ($l_i \leq w_i$). Điều này có nghĩa, nếu bạn mua vé tuyến thứ i (bất kỳ chiều nào) thì bạn phải trả w_i đồng, và trong ngày đó nếu muốn mua tiếp các vé của tuyến thứ i (bất kỳ chiều nào), bạn sẽ chỉ mất l_i đồng cho mỗi vé

Nhà Huy ở thành phố 1, nhà Giang ở thành phố n . Huy muốn tìm một lộ trình trong ngày, xuất phát từ nhà mình, mua quà ở một thành phố bất kỳ nào đó và kết thúc ở nhà Giang - tất nhiên rồi. Hãy giúp anh ấy tìm lộ trình sao cho tổng số tiền phải dùng là ít nhất có thể.

Dữ liệu vào

- Dòng đầu chứa n m
- Dòng tiếp theo chứa n số tự nhiên: c_1, c_2, \dots, c_n
- m dòng tiếp theo, dòng thứ i chứa: u_i v_i w_i l_i

Kết quả

Ghi tổng số tiền nhỏ nhất có thể

Ví dụ

stdin	stdout
5 6 10 10 1 10 10 1 2 2 2 2 3 1 1 3 4 1 1 4 5 2 2 1 5 1 1 2 5 2 0	6

Giải thích

Huy sẽ đi 1 -> 5 -> 2 -> (3) -> 2 -> 5

Hạn chế

- $n, m \leq 10^5$. $0 \leq w_i, l_i, c_i \leq 10^9$
- 20% test với $l_i = w_i$
- 20% test tiếp theo có $l_i = 0$
- 30% test tiếp theo có $n \leq 1000$
- 30% test tiếp theo không có ràng buộc bổ sung

Bài C. TGAME

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

Cây là một đồ thị vô hướng liên thông không có chu trình. Cho một cây có n đỉnh, các đỉnh được đánh số từ 1 đến n với gốc là đỉnh 1. Nút thứ i của cây được đặt a_i đồng xu. Ta nói cây có tính thống trị nếu với mọi nút không phải nút lá, số đồng xu ở nút đó phải lớn hơn hoặc bằng tổng số đồng xu ở các nút con trực tiếp của nó. Ban đầu cây được cho thỏa mãn tính thống trị.

Xét một trò chơi thú vị trên cây này như sau:

- Có hai người chơi. Hai người chơi này sẽ luân phiên thực hiện lượt chơi của mình
- Mỗi lượt chơi, người chơi chọn ra một nút không phải là nút gốc của cây và lấy đi một số đồng xu ở đó (ít nhất là một và nhiều nhất là toàn bộ), sao cho cây vẫn đảm bảo tính thống trị
- Ai không thực hiện được lượt chơi nữa thì thua cuộc

Bạn được mời tham gia trò chơi này với máy tính. Cụ thể, cho trước cây và số đồng xu trên mỗi nút, bạn được quyền chọn ai sẽ chơi trước và nhiệm vụ là phải giành phần thắng trong trò chơi.

Tương tác: Hệ thống cung cấp thư viện "tgamelib.h" có sẵn các hàm sau:

- `int get_n()`: trả về số nút của cây
- `int get_parent(int i)`: trả về nút cha của nút i ($2 \leq i \leq n$)
- `int get_a(int i)`: trả về số đồng xu ở nút i vào thời điểm ban đầu ($2 \leq i \leq n$)
- `void play(int i, int k)`: Thực hiện lượt chơi của bạn, lấy đi k đồng xu ở nút i ($2 \leq i \leq n, 1 \leq k \leq a_i$)
- `void get_play(int& i, int& k)`: Máy thực hiện lượt chơi, nút được chơi sẽ lưu vào biến i , số đồng xu được lấy sẽ được lưu vào biến k

Để sử dụng được thư viện, bạn cần có dòng khai báo `#include "tgamelib.h"` ở đầu chương trình. Bạn sẽ bị xử thua nếu:

- Gọi đến hàm `play(i,x)` hai lần liên tiếp mà không gọi đến hàm `get_play(i,x)` xen giữa
- Gọi đến hàm `get_play(i,x)` hai lần liên tiếp mà không gọi đến hàm `play(i,x)` xen giữa
- Gọi đến các hàm với tham số truyền vào không thuộc phạm vi
- Sau khi thực hiện lượt chơi, cây không còn đảm bảo tính thống trị

Trong quá trình chơi, nếu một trong hai đấu thủ thua cuộc, hệ thống sẽ tự động ngắt chương trình và chấm điểm

Hạn chế

- $2 \leq n \leq 10^5, 0 \leq a_i \leq 10^6, \sum_{i=2}^n a_i \leq 10^6$
- Subtask 1: Độ sâu của cây (số cạnh nhiều nhất trên đường đi từ gốc đến nút lá) bằng 1 và $\sum_{i=2}^n a_i \leq 10^3$
- Subtask 2: Độ sâu của cây (số cạnh nhiều nhất trên đường đi từ gốc đến nút lá) bằng 1
- Subtask 3: Độ sâu của cây (số cạnh nhiều nhất trên đường đi từ gốc đến nút lá) bằng $n - 1$
- Subtask 4: Ràng buộc gốc

Bài D. FPAIR

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

Một câu lạc bộ lập trình có nhiều thành viên rải rác khắp thành phố. Sắp tới họ muốn ghép đôi với nhau để làm việc hiệu quả hơn. Bản đồ thành phố là một đơn đồ thị vô hướng liên thông, gồm n địa điểm và m con đường hai chiều nối các địa điểm này với nhau. Có k thành viên trong câu lạc bộ đánh số từ 1 đến k , người thứ i sống tại địa điểm p_i (không nhất thiết phân biệt)

Vì ngại di chuyển, mỗi người trong số họ đều muốn ghép đôi với người gần họ nhất, nếu có nhiều người gần nhất thì người đó sẽ chọn người có chỉ số nhỏ nhất. Cho biết bản đồ thành phố. Với mỗi thành viên, hãy xác định xem anh ta muốn ghép đôi với người nào

Dữ liệu vào

- Dòng đầu tiên chứa n m k
- Dòng tiếp theo chứa dãy p
- m dòng tiếp theo, mỗi dòng chứa một cạnh: u v w

Kết quả

- Gồm k số nguyên là kết quả cho k người

Ví dụ

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

Hạn chế

- $1 \leq n, m, k \leq 10^5$
- $0 \leq w \leq 10^4$
- Có 30% test với $n, m, k \leq 1000$