

Bài 1: TAM GIÁC

Cho ma trận a gồm m hàng và n cột, các hàng đánh số từ trên xuống dưới, các cột đánh số từ trái sang phải. Ô ở hàng i cột j có giá trị nguyên $a_{i,j}$ ($|a_{i,j}| \leq 10^9$). Với mỗi ma trận vuông $k \times k$ định nghĩa *nửa dưới* của nó là tất cả các ô nằm trên đường chéo chính và dưới đường chéo chính trái trên – phải dưới.

Yêu cầu: Tìm hình vuông cỡ $k \times k$ sao cho tổng các số ở nửa dưới của nó đạt giá trị lớn nhất.

Dữ liệu: Vào từ file văn bản **TRIANGLE.INP**

- Dòng đầu ghi 3 số m, n, k ($m, n \leq 2000, k \leq \min(m, n)$)
- m dòng tiếp theo, mỗi dòng gồm n số là ma trận a .

Kết quả: Đưa ra file văn bản **TRIANGLE.OUT**

Ghi 1 số nguyên duy nhất là kết quả tìm được.

Ví dụ:

TRIANGLE.INP	TRIANGLE.OUT
3 4 2 1 2 <u>1 1</u> 2 1 <u>3 4</u> 1 2 1 2	8

Các giới hạn:

- Sub 1 (30%): $m, n \leq 50$
- Sub 2 (40%): $50 < m, n \leq 500$
- Sub 3 (30%): $500 < m, n \leq 2000$.

Bài 2: ĐÈN LỒNG

Trên trục đường phố chính của thành phố Thái Nguyên có n tòa nhà, được đánh số theo thứ tự từ 1 đến n , tòa nhà thứ i có chiều cao là một số nguyên dương h_i .

Để chuẩn bị cho Mùa du lịch Thái Nguyên năm nay, thành phố lập kế hoạch treo đèn lồng trang trí cho các tòa nhà. Hai tòa nhà liền kề i và $i + 1$ mất chi phí $c \times |h_i - h_{i+1}|$ (c là hằng số). Chi phí của trục đường là tổng chi phí của các tòa nhà kề nhau, tức là $S = c \times \sum_{i=1}^{n-1} |h_i - h_{i+1}|$. Thành phố đẹp nhất khi các con đường đều đẹp nhất. Tuy nhiên, do nguồn kinh phí có hạn, lãnh đạo thành phố quyết định chọn giải pháp cho tu sửa nâng chiều cao một số ngôi nhà để tiết kiệm chi phí, cụ thể nếu tòa nhà i nâng chiều cao thêm x (đơn vị, $x > 0$) thì thành phố phải mất một khoản chi phí là x^2 .

Yêu cầu: Cho biết n, c và các chiều cao h_i ($i = 1..n$), bạn hãy giúp thành phố tính chi phí S thấp nhất khi thực hiện theo kế hoạch nhé.

Dữ liệu: Vào từ file văn bản **MICOST.INP**

- Dòng đầu tiên chứa hai số nguyên n, c ($1 \leq n \leq 10^4, 1 \leq c \leq 10^6$);

- Mỗi dòng trong n dòng sau chứa một số nguyên h_i ($1 \leq h_i \leq 1000$).

Hai số liên tiếp trên cùng dòng được ghi cách nhau bởi dấu cách.

Kết quả: Đưa ra file văn bản **MICOST.OUT** một số nguyên là chi phí thấp nhất mà thành phố phải trả.

Ví dụ:

MICOST.INP	MICOST.OUT
5 2	15
2	
3	
5	
1	
4	

Giải thích: Nâng tòa nhà 1 thêm 1, nâng tòa nhà 4 thêm 2. Khi đó chiều cao các tòa nhà lần lượt là: 3, 3, 5, 3, 4.

Tổng chi phí là: $2 \times (0 + 2 + 2 + 1) + 1^2 + 2^2 = 15$.

Ràng buộc:

- Sub 1 (30%): $n \leq 10$; $h_i \leq 3$
- Sub 2 (40%): $n \leq 1000$; $h_i \leq 100$
- Sub 3 (30%): $n \leq 10000$; $h_i \leq 1000$.

Bài 3: TRUY VẤN TRÊN ĐỒ THỊ

Cho n đôi chè đánh số từ 1 đến n và $n - 1$ đường đi trực tiếp sao cho từ một đôi chè luôn có đường đi tới một đôi chè khác. Chi phí đi từ đôi chè i đến đôi chè j là một số nguyên dương c_{ij} .

Một đường đi đơn từ đôi chè u đến đôi chè v là dãy $u = x_1 x_2 \dots x_k = v$ trong đó (x_i, x_{i+1}) , $i = 1..(k - 1)$ là đường đi trực tiếp và với mọi i, j : $x_i \neq x_j$. Chi phí của đường đi trên là giá trị nhỏ nhất của các đường nối trực tiếp giữa hai đôi chè kề nhau nằm trên đường đi đó.

Yêu cầu: Cho Q truy vấn, mỗi truy vấn được mô tả bởi hai số nguyên k, v với ý nghĩa: Đếm xem có bao nhiêu đôi chè u mà đường đi đơn từ đôi chè u đến đôi chè v có chi phí không nhỏ hơn k .

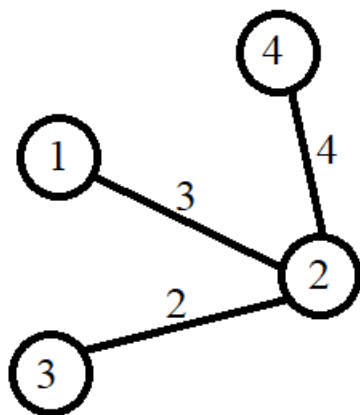
Dữ liệu: Vào từ file văn bản **QKGRAPH.INP**

- Dòng thứ nhất chứa hai số nguyên dương n, Q ($1 \leq n, Q \leq 10^5$)
- $n - 1$ dòng sau mô tả các đường nối trực tiếp giữa các đôi chè. Dòng thứ i chứa ba số nguyên dương p_i, q_i, c_i thể hiện có đường đi từ đôi chè p_i đến đôi chè q_i với chi phí c_i ($1 \leq p_i, q_i \leq n, 1 \leq c_i \leq 10^9$).
- Q dòng tiếp theo, mỗi dòng ghi một truy vấn gồm hai số nguyên k, v ($1 \leq k \leq 10^9, 1 \leq v \leq n$) thể hiện yêu cầu đếm xem có bao nhiêu đôi chè mà chi phí đường đi đơn từ nó đến đôi chè v không nhỏ hơn k .

Kết quả: Ghi ra file văn bản **QKGRAPH.OUT**

Gồm Q dòng, mỗi dòng ghi một số nguyên là kết quả của truy vấn tương ứng (theo thứ tự xuất hiện trong file dữ liệu vào).

Ví dụ:



QKGRAPH.INP	QKGRAPH.OUT
4 3	3
1 2 3	0
2 3 2	2
2 4 4	
1 2	
4 1	
3 1	

Các giới hạn:

- Sub 1 (30%): $n \leq 10^5, Q = 1$
- Sub 2 (30%): $n \leq 1000, Q \leq 10^3$
- Sub 3 (40%): $n, Q \leq 10^5$.

-----Hết-----