

allmst

Khi học về cây khung nhỏ nhất, Alice đã nghĩ tới bài toán tìm tất cả các cây khung nhỏ nhất. Cụ thể, với một đơn đồ thị gồm n đỉnh (các đỉnh được đánh số từ 1 đến n) và m cạnh (các cạnh được đánh số từ 1 đến m), cạnh thứ k ($1 \leq k \leq m$) nối giữa hai đỉnh i_k, j_k và có trọng số là c_k , Alice cần tìm m cây khung nhỏ nhất, cây khung thứ k bắt buộc phải chứa cạnh thứ k , liên thông cả n đỉnh và có trọng số nhỏ nhất.

Input

- Dòng đầu chứa hai số nguyên n, m ($n - 1 \leq m \leq 10^6$);
- Tiếp theo là m dòng, mỗi dòng chứa ba số nguyên dương i_k, j_k, c_k ($1 \leq i_k \neq j_k \leq n; c_k \leq 10^9$).

Output

- Gồm m dòng, dòng thứ k chứa một số nguyên là tổng trọng số của cây thứ k tìm được.

Input	Output
3 3	2
1 2 1	2
1 3 1	3
2 3 2	

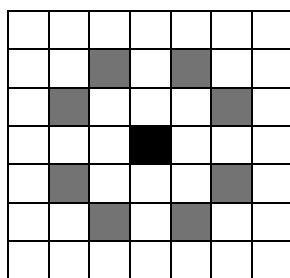
Subtask 1: $n \leq 300$;

Subtask 2: $n \leq 3 \times 10^5$;

Tô màu lưới

Cho lưới ô vuông kích thước $m \times n$. Các hàng được đánh số từ 1 đến m , từ trên xuống dưới; các cột được đánh số từ 1 đến n , từ trái qua phải. Ô vuông thuộc hàng thứ i và cột thứ j có tọa độ (i, j) . Người ta tô các ô vuông bởi $2 \times n$ màu có mã màu được đánh số từ 1 đến $2 \times n$ sao cho mỗi màu đều được tô cho ít nhất một ô. Ký hiệu L_j là số lượng màu khác nhau được sử dụng để tô các ô trong cột thứ j ($j = 1, 2, \dots, n$). Ta gọi **độ đa sắc** của lưới là giá trị $\max_{1 \leq j \leq n} L_j$.

Cho phép thực hiện việc hoán đổi màu của hai ô ở hai đỉnh đối diện trên đường chéo của hình chữ nhật kích thước 2×3 bất kỳ (xem hình bên dưới). Mỗi phép hoán đổi được mô tả bởi bốn số nguyên (u, v, s, t) cho biết hai ô vuông (u, v) và (s, t) được hoán đổi màu.



Ô đánh dấu bởi màu đen có thể hoán

đổi màu với một trong các ô đánh dấu bởi màu xám

Yêu cầu: Hãy xác định một dãy các phép hoán đổi màu để đưa lưới về trạng thái có độ đa sắc nhỏ nhất.

Input

Dòng thứ nhất chứa số nguyên dương T ($T \leq 30$) là số lượng bộ dữ liệu. Mỗi nhóm dòng trong T nhóm dòng tiếp theo mô tả một bộ dữ liệu theo khuôn dạng sau:

- Dòng đầu tiên chứa 2 số nguyên m, n ($4 \leq m, n \leq 50$) được ghi cách nhau bởi dấu cách;
- Dòng thứ i trong số m dòng tiếp theo chứa n số nguyên dương $c_{i1}, c_{i2}, \dots, c_{in}$ được ghi cách nhau bởi dấu cách, trong đó c_{ij} là mã màu của ô (i, j) trong lưới ban đầu ($j = 1, 2, \dots, n$).

Output

Gồm T nhóm dòng, mỗi nhóm là kết quả tìm được cho bộ dữ liệu tương ứng trong dữ liệu vào, theo khuôn dạng sau:

- Dòng đầu tiên ghi ra số nguyên không âm p là số lượng phép hoán đổi cần thực hiện;

- Tiếp đến là p dòng mô tả dãy các phép hoán đổi cần thực hiện để đưa lưới về trạng thái có độ đa sắc nhỏ nhất. Mỗi dòng ghi 4 số nguyên dương u, v, s, t cách nhau bởi dấu cách cho biết cần thực hiện việc hoán đổi màu của hai ô vuông (u, v) và (s, t) .

Nếu có nhiều cách thực hiện để đưa lưới về trạng thái có độ đa sắc nhỏ nhất thì chỉ cần đưa ra một cách.

Input	Output
2	0
4 4	2
1 2 3 4	2 2 4 3
5 6 7 8	2 2 4 1
1 2 3 4	
5 6 7 8	
4 4	
1 2 3 4	
5 7 7 8	
1 2 3 4	
6 6 5 8	

r2color

Cho một đơn đồ thị vô hướng, mỗi đỉnh được tô bằng màu 1 hoặc 2. Một cách tô màu gọi là đẹp nếu hai đỉnh có cạnh nối thì không được cùng màu. Bạn cần tô lại màu của ít đỉnh nhất để nhận được cách tô màu đẹp, thật tiếc bạn chỉ được phép tô lại màu của một số đỉnh.

Yêu cầu: Cho đồ thị, cho cách tô màu ban đầu, cho biết các đỉnh được phép tô màu lại, hãy tính số đỉnh ít nhất cần tô lại để nhận được cách tô màu đẹp.

Input

- Dòng đầu chứa hai số nguyên n, m ($n, m \leq 3e5$);
- Dòng thứ hai gồm n số 1 hoặc 2 cho biết cách tô màu ban đầu;
- Dòng thứ ba gồm n số 0/1 cho biết đỉnh tương ứng không/có được phép tô lại màu;
- Tiếp theo là m dòng, mỗi dòng hai số mô tả các cạnh của đồ thị.

Output

- Gồm một số là số đỉnh ít nhất cần tô lại để nhận được cách tô màu đẹp hoặc ghi ra -1 nếu không tồn tại cách thực hiện.

Input	Output
4 4 1 1 1 1 1 1 1 0 1 2 2 3 3 4 4 1	2

Subtask 1: Có không quá một đỉnh được phép tô lại;

Subtask 2: Không có ràng buộc nào thêm.

TOLL

Vùng đất hạnh phúc được mô tả bởi một tập gồm N khu phố (đánh số từ 1 đến N), ban đầu các khu phố này được nối với nhau bởi M tuyến phố hai chiều (đánh số từ 1 đến M). Khu phố số 1 là khu phố trung tâm. Các tuyến phố đảm bảo rằng một người có thể di chuyển từ khu phố số 1 tới bất kỳ một khu phố khác sử dụng các tuyến phố này. Các tuyến phố này là các tuyến phố có thu phí. Một người đi trên tuyến phố i phải trả một mức phí là c_i xu cho chủ của tuyến phố này. Biết rằng, tất cả các mức phí c_i là khác nhau. Hiện tại, có thêm K tuyến phố mới được hoàn thành và chúng được sở hữu bởi tỉ phú Greedy. Ông Greedy có thể quyết định mức phí thu của các tuyến phố mới (mức phí thu không nhất thiết phải khác nhau) và ông sẽ thông báo mức phí thu trên các tuyến phố mới vào ngày mai.

Hai tuần sau, sẽ có một lễ hội hóa trang lớn ở Vùng đất hạnh phúc này! Một lượng lớn người tham dự sẽ di chuyển đến khu phố trung tâm và diễu hành dọc theo các tuyến phố. Có tất cả p_j người bắt đầu đi từ khu phố j đến khu phố trung tâm. Họ chỉ đi trên một tập các tuyến phố đã được chọn, các tuyến phố này sẽ được thông báo một ngày trước lễ hội. Theo truyền thống, các tuyến phố sẽ được lựa chọn bởi người giàu nhất Vùng đất hạnh phúc, người đó chính là ông Greedy. Cũng theo quy định truyền thống, ông Greedy phải lựa chọn một tập các tuyến phố sao cho tổng mức phí trên tập các tuyến phố được chọn là nhỏ nhất, và đồng thời đảm bảo rằng mọi người đều có thể di chuyển từ khu phố j đến khu phố 1 (tức là, các tuyến phố được chọn sẽ tạo thành một “cây khung nhỏ nhất” trong đó mức phí thu trên mỗi tuyến phố chính là trọng số của cạnh tương ứng trên cây khung). Nếu có nhiều tập các tuyến phố như vậy, ông Greedy có thể lựa chọn một tập tùy ý, miễn là tổng mức phí là nhỏ nhất.

Ông Greedy hiểu rất rõ rằng tổng số tiền ông thu được trên K tuyến phố mới của ông không đơn thuần phụ thuộc vào mức phí của chúng. Số tiền thu được của một tuyến phố thực tế là tổng số phí thu được từ những người di chuyển trên tuyến phố đó. Chính xác hơn là, nếu có p người di chuyển trên tuyến phố i , tổng số tiền thu được từ tuyến phố i là tích $c_i \times p$. Lưu ý rằng, ông Greedy chỉ có thể thu phí trên các tuyến phố mới, bởi vì ông không sở hữu bất cứ một tuyến phố cũ nào. Ông Greedy lén lút xây dựng một kế hoạch nhằm cực đại hóa tổng số tiền ông có thể thu được trong thời gian diễn ra lễ hội bằng cách ấn định mức phí trên các tuyến phố mới cũng như lựa chọn tập các tuyến phố. Ông ta muốn ấn định mức phí trên các tuyến phố mới (các mức phí này sẽ được thông báo vào ngày mai), và lựa chọn các tuyến phố cho lễ hội (các tuyến phố được lựa sẽ được thông báo một ngày trước lễ hội) sao cho tổng số tiền ông ta thu được từ K tuyến phố mới là lớn nhất. Lưu ý rằng, ông Greedy vẫn phải tuân thủ các quy định truyền thống về việc lựa chọn tập các tuyến phố sao cho tổng mức phí là nhỏ nhất.

Bạn là một phóng viên và muốn vạch trần kế hoạch mờ ám của ông Greedy. Để làm được việc đó, trước tiên bạn phải viết một chương trình để xác định tổng số tiền mà ông Greedy có thể thu được từ kế hoạch mờ ám của ông ta.

Input

- Dòng đầu tiên chứa 3 số nguyên N , M và K , các số được phân cách nhau bởi dấu cách.
- Tiếp theo là M dòng mô tả về M tuyến phố ban đầu. Dòng thứ i trong số các dòng này chứa các số nguyên a_i , b_i và c_i , các số được phân cách nhau bởi dấu cách, cho biết có một tuyến phố hai chiều giữa hai khu phố a_i và b_i với mức phí là c_i .

- Tiếp theo là K dòng mô tả về K tuyến phố mới được xây dựng thêm. Dòng thứ i trong số các dòng này chứa các số nguyên x_i và y_i cách nhau bởi dấu cách cho biết có một tuyến phố mới nối hai khu phố x_i và y_i .

- Dòng cuối cùng chứa N số nguyên được phân cách nhau bởi dấu cách, số thứ j có giá trị p_j là số người từ khu phố j di chuyển tới khu phố 1.

Dữ liệu đầu vào cũng thỏa mãn các điều kiện ràng buộc sau:

- $1 \leq N \leq 100,000$
- $1 \leq K \leq 20$
- $1 \leq M \leq 300,000$
- $1 \leq c_i, p_j \leq 10^6$ cho mỗi i và j .
- $c_i \neq c_{i'}$, nếu $i \neq i'$.
- Tồn tại nhiều nhất một tuyến phố giữa hai khu phố bất kì (bao gồm cả các tuyến phố mới).

Output

- Ghi một số nguyên duy nhất, đó chính là tổng số tiền lớn nhất có thể thu được.

Input	Output
5 5 1 3 5 2 1 2 3 2 3 5 2 4 4 4 3 6 1 3 10 20 30 40 50	400

Subtask 1: $N \leq 10$, $M \leq 20$ và $K = 1$

Subtask 2: $N \leq 1,000$, $M \leq 5,000$ và $K \leq 10$.

Subtask 3: $N \leq 100,000$, $M \leq 300,000$ và $K \leq 15$.

Subtask 4: $N \leq 100,000$, $M \leq 300,000$ và $K \leq 20$.