



Dịch đề: HDP và đồng đảng

Bờ bên này là Thành Công, Bờ bên kia là
Thất Bại. Ở giữa là con sông Bỏ Cuộc.
Bắc ngang sông là cây cầu Cổ Găng. Tất
cả là lựa chọn của BẠN !

HSGS Trường
Mễ Trì thôn
Lương Thế Vinh phố
Thanh Xuân quận
Hà Nội thành

Mục lục bài

UVA908. TÁI KẾT NỐI.....	3
UVA1208. BẢO VỆ THÀNH PHỐ.....	3
UVA1235. KHÓA CHỐNG TRỘM.....	4
UVA10034. MỤN TRÚNG CÁ.....	5
UVA11228. HỆ THỐNG GIAO THÔNG.....	6
UVA11631. CHIẾU SÁNG BAN ĐÊM.....	7
UVA11710. CON ĐƯỜNG ĐẤT ĐỎ.....	7
UVA11733. ĐƯỜNG GIAO THÔNG.....	8
UVA11747. ĐỒ THỊ VÔ HƯỚNG.....	9
UV11857. CHẶNG XE.....	10
UVA10099. HƯỚNG DẪN VIÊN DU LỊCH.....	10
UVA10147. ĐƯỜNG BỘ.....	11
UVA10842. LUỒNG GIAO THÔNG.....	12
UVA534. ẾCH.....	13
UVA10369. MẠNG LƯỚI CĂN CỨ.....	14
UVA1216. LẮP MÁY THU PHÁT.....	14
UVA544. CHUYÊN TRỞ HÀNG HÓA.....	15

UVA908. TÁI KẾT NỐI

Hãy xét vấn đề từ tập hợp M đường truyền nối từng cặp hai máy tính, lựa chọn ra tập T các đường truyền để kết nối máy tính sao cho giữa mỗi cặp máy tính đều có một dãy các đường truyền nối với nhau. Mỗi đường truyền có một giá thuê bao hàng tháng. Mục tiêu của chúng ta là phải giảm thiểu tổng chi phí giá thuê bao của tất cả các đường truyền nối N máy tính (chỉ xét các đường truyền có trong tập T). Giả định là trước đó, chúng ta đã giải quyết được vấn đề này cho N máy tính và M đường truyền. Nhưng bây giờ, có thêm K đường truyền mới. Bạn phải tìm ra tập T' sao cho tổng giá của T' nhỏ hơn tổng giá của T (bởi vì lúc này, sẽ có thêm $M+K$ đường truyền).

INPUT

Input có nhiều test, các test cách biệt nhau bằng dòng trống, mỗi test có khuôn dạng sau

- Dòng đầu tiên chứa số lượng máy tính N với $1 \leq N \leq 1000000$, các máy tính có số hiệu i , $1 \leq i \leq N$.
- Tập T đã được chọn trước, gồm $N-1$ dòng, mỗi dòng mô tả về một đường truyền có khuôn dạng $x \ y \ z$, thể hiện đường truyền nối máy tính x với máy tính y , giá thuê bao của đường truyền này là z .
- Một dòng ghi số nguyên K $1 \leq K \leq 10$ là số lượng đường truyền mới.
- Sau đó là K dòng, mỗi dòng mô tả về một đường truyền có khuôn dạng $x \ y \ z$, thể hiện đường truyền nối máy tính x với máy tính y , giá thuê bao của đường truyền này là z .
- Một dòng ghi số M với $N-1 \leq M \leq N(N-1)/2$ là số lượng đường truyền ban đầu.
- Sau đó là M dòng, mỗi dòng mô tả về một đường truyền có khuôn dạng $x \ y \ z$, thể hiện đường truyền nối máy tính x với máy tính y , giá thuê bao của đường truyền này là z .

OUTPUT

Với mỗi test, output phải tuân theo mẫu dưới đây. Mỗi output sẽ cách nhau 1 dòng trống.

Output file phải có 1 dòng ghi số chi phí ban đầu của M đường kết nối tới N trang mạng và một dòng ghi chi phí của $M+K$ đường kết nối tới N trang mạng. Nếu 2 giá trị bằng nhau thì giá trị đó sẽ được viết 2 lần.

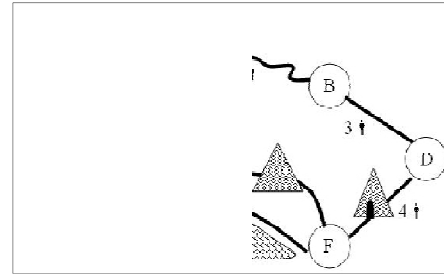
Sample Input	Sample Output
5 1 2 5 1 3 5 1 4 5 1 5 5 1 2 3 2 6 1 2 5 1 3 5 1 4 5 1 5 5 3 4 8 4 5 8	20 17

UVA1208. BẢO VỆ THÀNH PHỐ

Vào thế kỷ thứ 25, sau nhiều thảm họa thiên nhiên, loài người phải sống trong những thành phố pháo đài, kết nối với nhau qua các đường hầm, di chuyển bằng phương tiện Distro-Plane. Mỗi thành phố có một loại mỏ oreo dùng để tu sửa tường thành. Oreo từ các thành phố hai khác nhau kết hợp với nhau tạo thành vật liệu siêu bền Oreon. Bức tường bảo vệ của mỗi thành phố được tạo thành từ loại vật chất đặc biệt siêu bền Oreon này.

Bên ngoài những thành phố này là những tên khủng bố với súng Micro\$ Beam có thể bắn nát đường hầm, nhưng chỉ làm tróc sơn lớp bảo vệ thành phố. Vậy nên mỗi thành phố kết nối với nhiều thành phố khác nhau phòng trường hợp các đường hầm này bị hỏng.

Nếu một đường hầm bị hỏng, mọi người sẽ không đi qua được và cần một lượng Oreon để sửa lại. Khi một thành phố bị cô lập (nghĩa là mọi đường hầm nối đến nó đều đã bị thiêu rụi), thì đồng nghĩa với việc thành phố đó bị khủng bố chiếm hữu hoàn toàn.



Bạn là lãnh đạo của phe phòng thủ. Nhiệm vụ của bạn là phải đảm bảo tất cả các thành phố phải giữ được ít nhất một đường hầm đi đến nó. Với đạo quân ít ỏi trong tay, bạn sẽ phải quyết định nên bảo vệ những đường hầm nào để sử dụng ít nhân lực nhất, nhưng vẫn đảm bảo không thành phố nào bị cô lập. Hình sau cho thấy bản đồ của nhiều thành phố

INPUT

Input bao gồm nhiều test. Dòng đầu là số số test. Mỗi test bắt đầu với một số biểu diễn số lượng thành phố pháo đài, tên các pháo đài là chữ cái thường tiếng Anh. Sau số đó là số nhân lực cần thiết để bảo vệ đường hầm. Giá trị 0 biểu thị không còn đường ống ở (do đó không cần người bảo vệ).

OUTPUT

Bạn cần xuất ra dòng đầu là thứ tự của test, tiếp đến là tên 2 thành phố có đường hầm nối với nhau. Một dấu cách và số người cần để bảo vệ đường hầm. Sắp xếp theo thứ tự từ điển.

Sample Input	Sample Output
1 6 0, 8, 12, 0, 0, 7 8, 0, 0, 3, 0, 0 12, 0, 0, 0, 6, 0 0, 3, 0, 0, 0, 4 0, 0, 6, 0, 0, 5 7, 0, 0, 4, 5, 0	Case 1: B-D 3 D-F 4 E-F 5 C-E 6 A-F 7

UVA1235. KHÓA CHỐNG TRỘM



Gần đây, có vấn đề nghiêm trọng với chiếc két sắt ở vùng Panda: một số thứ đồ an toàn đã bị lấy cắp! Chiếc két sử dụng khóa xoay gồm 4 chữ số (bạn chỉ phải xoay các chữ số, lên hoặc xuống, cho tới khi cả 4 chữ số trùng với khóa

bí mật). Mỗi khóa được thiết kế để xoay từ 0 đến 9. Xoay quá số 9 sẽ được số 0, và xoay quá số 0 sẽ lại được số 9. Do đó sẽ có thể xoay được tất cả 10000 khóa, từ 0000 đến 9999, mọi người đều có thể thử từng khóa cho đến khi trùng khóa bí mật.

Điều gì xảy ra cũng đã xảy ra rồi. Nhưng để giảm thiểu tốc độ tấn công của kẻ trộm về sau, Cục Bảo Mật Vùng Panda (PSA) đã thiết kế một loại khóa an toàn hơn với nhiều khóa bí mật. Thay vì chỉ sử dụng một khóa bí mật, khóa mới sử dụng tới N khóa bí mật, khóa chỉ có thể mở một khi tất cả các khóa bí mật đều được mở. Cơ chế hoạt động khóa mới như sau:

1. Ban đầu các kí tự là 0000.
2. Các khóa bí mật có thể được mở theo một thứ tự bất kì, bằng cách xoay các kí tự trong khóa để trùng với kí tự bí mật, và sau đó ấn nút UNLOCK.
3. Nút JUMP thần kì có thể chuyển tất cả các kí tự trở thành một khóa bí mật bất kỳ đã được mở trước đó mà không cần phải xoay.
4. Chiếc két sẽ được mở nếu và chỉ nếu tất cả các khóa bí mật đều được mở bằng số bước xoay nhỏ nhất, không tính nút JUMP (đây chính là tính năng tuyệt vời nhất).

Nếu số lần xoay vượt số lần nhỏ nhất, thì tất cả các kí tự sẽ trở thành 0000 và chiếc khóa sẽ bị khóa lại. Nói cách khác, trạng thái của chiếc khóa sẽ trở lại ban đầu, quá trình phá mã thất bại. PSA thấy khá tự tin rằng hệ thống mới sẽ giảm thiểu tốc độ phá khóa, cho chúng ta đủ thời gian để nhận ra và bắt lũ trộm. Để xác định số bước xoay nhỏ nhất cần thiết để phá khóa, PSA nhờ bạn để viết một chương trình. Cho tất cả các khóa bí mật, tính toán số lần xoay nhỏ nhất để mở tất cả các khóa.

INPUT

Dòng đầu tiên bao gồm chữ T, số lượng các trường hợp test sau đó. Mỗi trường bao gồm một số N, số lượng các khóa, N dòng tiếp theo, mỗi dòng gồm một số có 4 chữ số (có thể có các chữ số 0 đứng đầu) là các khóa bí mật.

OUTPUT

Với mỗi trường hợp test, in ra một dòng duy nhất là số các bước nhỏ nhất để mở tất cả các khóa bí mật.

Sample Input	Sample Output
4	16
2 1155 2211	20
3 1111 1155 5511	26
3 1234 5678 9090	17
4 2145 0213 9113 8113	

Giải thích trường hợp thứ 2: Xoay 0000 thành 1111, số lần xoay: 4. Xoay 1111 thành 1155, số lần xoay: 8. Nhảy 1155 thành 1111, có thể làm được điều này bởi 1111 đã được mở trước đó. Xoay 1111 thành 5511, số lần xoay: 8. Tổng số lần xoay = 4 + 8 + 8 = 20.

UVA10034. MỤN TRỨNG CÁ

Ở một tập phim trong loạt phim Dick Van Dyke, cậu bé Richie nối các cái mụn trứng cá trên lưng của cha để phác họa bức tranh Liberty Bell. Giả định lưng của người cha là một mặt phẳng tọa độ và tọa độ các mụn trứng cá là (x,y) . Bạn hãy chỉ cho Richie cách kết nối các điểm lại sao cho tốn ít mực nhất. Rechie nối các điểm bằng cách vẽ một đường thẳng nối giữa 2 điểm, có thể nhắc bút giữa các đường nối. Khi Rechie hoàn thành thì giữa bất kỳ 2 mụn trứng cá nào cũng phải có đường đi.

INPUT

Dòng đầu tiên nhập số nguyên dương là số test, sau đó cách ra một dòng. Với mỗi test cũng sẽ cách nhau một dòng và được mô tả dưới đây: Dòng đầu tiên là số $0 < n \leq 100$ là số mụn trứng cá ở lưng của người cha. Với mỗi cái mụn thì có 2 số thực x,y là tọa độ của nó.

OUTPUT

Chú ý: với mỗi test khác nhau thì phải cách nhau một dòng trống. In ra một số thực có 2 số sau dấu phẩy : tổng độ dài nhỏ nhất của các đường nối mà 2 điểm bất kỳ đều có đường đến với nhau.

Sample input:	Sample output:
1	3.41
3	
1.0 1.0	
2.0 2.0	
4.0 4.0	

UVA11228. HỆ THỐNG GIAO THÔNG

Trong đất nước Graphland, có rất nhiều thành phố nhưng không có đường. Chính quyền liên bang muốn thay đổi và lên kế hoạch xây dựng đường bộ và đường sắt để tất cả các thành phố trong nước đều được nối với nhau qua hệ thống đường giao thông mới. Tuy nhiên, Graphland sẽ chỉ xây dựng các tuyến đường giữa các thành phố trong phạm vi cùng 1 bang và sử dụng các tuyến đường sắt để kết nối các thành phố khác bang. Để thực hiện điều này, chính quyền coi khoảng cách dài nhất giữa 2 thành phố trong cùng 1 bang không vượt quá r . Để tiết giảm chi phí xây dựng đường bộ và đường sắt, chính quyền chỉ muốn xây dựng những tuyến đường bộ và đường sắt cần thiết để bất kỳ hai thành phố nào của đất nước cũng có thể đi được đến nhau. Bạn phải xác định các mạng lưới giao thông với chi phí thấp nhất mà Graphland sẽ xây dựng.

INPUT

Dòng thứ nhất của là số bộ test. Dòng đầu tiên của bộ test gồm 2 số nguyên n ($1 \leq n \leq 1000$), là số thành phố trong Graphland, và r ($0 \leq r \leq 40000$) là giá trị để xác định các thành phố có trong cùng 1 bang hay không. n dòng tiếp theo là danh sách tọa độ nguyên (x,y) của các thành phố ($-10000 \leq x,y \leq 10000$).

OUTPUT

Mỗi bộ test xuất ra kết quả nằm trên một dòng. Dòng đó sẽ xác định các bộ dữ liệu với một số (bắt đầu từ 1 và tăng dần lên với mỗi bộ dữ liệu), sau đó là số các bang trong Graphland và độ dài nhỏ nhất (làm tròn thành phần nguyên) của cả đường bộ và đường sắt phải xây dựng để thỏa mãn điều kiện đề bài.

Sample input	Sample output
3	Case #1: 1 2 0
3 100	Case #2: 3 0 200
0 0	Case #3: 2 24 28
1 0	
2 0	
3 1	
0 0	
100 0	
200 0	
4 20	
0 0	
40 30	
30 30	
10 10	

UVA11631. CHIẾU SÁNG BAN ĐÊM

Trong thời kỳ kinh tế khó khăn, ngay cả với vùng Byteland. Để giảm thiểu chi phí vận hành, chính quyền Byteland quyết định tối ưu hóa việc thắp sáng đèn đường. Đến nay tất cả những con đường đều đã được chiếu sáng vào ban đêm, với chi phí 1 đôla Byteland trên 1 mét, 1 ngày. Để tiết kiệm tiền, chính quyền quyết định không thắp sáng tất cả con đường và sẽ tắt đèn ở một số con đường. Để chắc chắn rằng tất cả những người dân ở Byteland đều cảm thấy an toàn, mặc dù tắt đèn trên một số con đường khi trời tối nhưng sẽ có ít nhất một tuyến đường được thắp sáng nối giữa hai ngã tư bất kỳ.

Câu hỏi đặt ra là số tiền tối đa mà chính quyền Byteland có thể tiết kiệm mà vẫn làm người dân cảm thấy an toàn?

INPUT

File input gồm nhiều test. Mỗi test bắt đầu bằng 2 số m và n là số ngã tư của Byteland và số đường ở Byteland. Input sẽ kết thúc khi $m=n=0$. Ồ ừ không, $1 \leq m \leq 200000$ và $m-1 \leq n \leq 200000$. Sau n là một bộ 3 số x, y, z thể hiện có một con đường hai chiều giữa x và y với chiều dài là z mét ($0 \leq x, y < m$ và $x \neq y$). Đồ thị trong mỗi test là liên thông. Tất cả chiều dài của tất cả những con đường trong mỗi test nhỏ hơn 231.

OUTPUT

Mỗi test in ra một dòng ghi số tiền tối đa mà chính quyền có thể tiết kiệm được

Sample Input	Sample Output
7 11 0 1 7 0 3 5 1 2 8 1 3 9 1 4 7 2 4 5 3 4 15 3 5 6 4 5 8 4 6 9 5 6 11 0 0	51

UVA11710. CON ĐƯỜNG ĐẤT ĐỎ

Peter sống trong *Thành Phố Đất Đỏ*, một trong những thành phố đắt nhất trên thế giới. Peter không đủ tiền để mua ô tô và các tuyến xe buýt ở đây khá tệ, vì vậy anh ta quyết định chọn tàu điện ngầm là phương tiện đi làm hàng ngày. Cho đến bây giờ, vé của tàu điện ngầm khá rẻ bạn có thể đi đến bất cứ đâu chỉ với 1 vé có giá \$2. Ồ hưng tháng trước, giám đốc xí nghiệp tàu điện đã quyết định thành lập EFS (hệ thống vé siêu đắt).

Với hệ thống này, mọi người sẽ bắt buộc phải sử dụng vé tháng cho từng tuyến, bạn có thể đi bao nhiêu lần tùy thích trên tuyến mà vé đó đã quy định (theo cả 2 chiều). Số tiền của mỗi loại vé tháng được cho trước, vì vậy bạn phải lựa chọn thật cẩn thận các loại vé cần mua sao cho tối ưu nhất. Để đi từ Picadilly đến Queensway thì số tiền ít nhất bạn phải trả là \$12 (mua 2 vé Picadilly-Victoria và Victoria-Queensway). Peter là một nhân viên tiếp thị và anh cần phải đến mọi nơi trong thành phố. Và bây giờ là lúc bạn cần phải xuất hiện để giúp đỡ. Biết trước các trạm và giá tiền xe điện giữa hai trạm bất kỳ, cũng như trạm gần nhà Peter nhất, chương trình phải xuất ra tổng số tiền tối thiểu mà Peter phải trả để đi hết qua các trạm. Chương trình cũng phải xác định liệu từ bến tàu điện ngầm gần nhà Peter không thể đi đến được địa điểm nào đó không (vì khi đó anh ta sẽ quyết định chuyển sang đi xe buýt).



INPUT

Bao gồm nhiều test. Mỗi test bắt đầu bằng 1 dòng gồm 2 số. $1 < s \leq 400$ (số trạm) và $0 \leq c \leq 78900$ (số tuyến) cách nhau bởi 1 dấu cách. Tiếp theo là s dòng là tên từng bến tàu điện biểu diễn bằng xâu kí tự (tối đa 10 chữ cái). Tiếp sau đó là c dòng mỗi dòng bao gồm 2 trạm tàu điện có tuyến đi giữa chúng và giá vé tháng của tuyến đó mỗi cái đều cách nhau 1 dấu cách. Dòng cuối cùng của mỗi test là tên trạm gần nhà Peter nhất. Input sẽ kết thúc khi gặp dòng '0 0'.

OUTPUT

Mỗi test sẽ in ra số tiền ít nhất mà Peter phải trả để có thể đi đến mọi nơi trong thành phố. In ra impossible nếu có 1 vài điểm nào đó không thể đến được từ trạm tàu gần nhà.

Sample Input	Sample Output
3 3 Picadilly Victoria Queensway Picadilly Victoria 2 Queensway Victoria 10 Queensway Picadilly 20 Picadilly 4 2 Picadilly Victoria Queensway Temple Picadilly Victoria 2 Temple Queensway 100 Temple 0 0	12 Impossible

UVA11733. ĐƯỜNG GIAO THÔNG

Chính quyền của một quốc gia đang phát triển muốn xây dựng tuyến giao thông tới các vùng khó tiếp cận nhất, để thu hút vốn đầu tư. Vùng này gồm một số trọng điểm bắt buộc phải được kết nối với sân bay. Đương nhiên, một giải pháp là xây dựng sân bay ở mọi địa điểm – nhưng có cách rẻ hơn là xây sân bay ở một địa điểm và từ các địa điểm khác có thể đi đường bộ đến sân bay. ả hững tuyến đường đều là đường 2 chiều. Và có thể có nhiều hơn một tuyến đường đi giữa 2 địa điểm (ví dụ: một đường hầm qua núi và một đường khác đi vòng quanh núi) với giá tiền có thể khác nhau. Một địa điểm được coi là có kết nối tới sân bay khi có sân bay ở địa điểm đó hoặc là tồn tại tuyến đường đi từ địa điểm này tới một địa điểm khác có đặt sân bay.

Bạn được cho biết chi phí xây dựng sân bay và một danh sách các đường đi giữa các cặp địa điểm và giá tiền tương ứng. Chính quyền muốn bạn đưa ra quyết định để đảm bảo mọi địa điểm đều có kết nối tới sân bay. ả ếu tồn tại nhiều phương án với cùng giá tiền tối thiểu, hãy chọn ra phương án có nhiều sân bay nhất.

Lưu ý: Input có thể rất lớn, bạn hãy đảm bảo I/O code của bạn chạy đủ nhanh.

Input

Dòng đầu tiên chứa số nguyên T ($T < 25$), là số test. Phần còn lại của input là T tests. Mỗi test gồm 3 số nguyên ả , M và A ($0 < ả \leq 10,000$, $0 < M \leq 100,000$, $0 < A \leq 10,000$) cách nhau 1 dấu cách. ả là số địa điểm, M là số tuyến đường có

thể xây dựng, và A là giá để xây dựng sân bay. M dòng tiếp theo chứa 3 số nguyên X, Y and C ($1 \leq X, Y \leq A$, $0 < C \leq 10,000$), cách nhau bởi dấu cách. X và Y là 2 địa điểm, và C là giá tiền để xây dựng tuyến đường giữa X và Y.

Output

Chương trình của bạn phải chứa đúng T dòng tương ứng với T test. Mỗi dòng có dạng: “Case #X: Y Z”, trong đó X là số thứ tự của test, Y là giá tiền tối thiểu để xây dựng sân bay và các tuyến đường, và Z là số sân bay.

Sample Input	Sample Output
2 4 4 100 1 2 10 4 3 12 4 1 41 2 3 23 5 3 1000 1 2 20 4 5 40 3 2 30	Case #1: 145 1 Case #2: 2090 2

UVA11747. ĐỒ THỊ VÔ HƯỚNG

Cho một đồ thị vô hướng, các cạnh có trọng số. Cây khung tối thiểu là tập các cạnh có tổng trọng số nhỏ nhất, mà bất kì 2 đỉnh nào trong tập đều có đường đi đến nhau. Một thuật toán nổi tiếng để tìm cây kéo khung nhỏ nhất trong đồ thị như sau:

- Cho T ban đầu rỗng.
- Sắp xếp các cạnh $e_1, e_2 \dots e_n$ thứ tự tăng dần theo độ dài.
- Cho e_i vào T nếu 2 đỉnh đầu mút của e_i không tạo ra thành phần liên thông trong T.

Một thuật toán khác:

- Cho T ban đầu là tập tất cả các cạnh.
- Luôn có chu trình C trong T.
 - Xóa các cạnh e khỏi T, trong đó e có trọng số lớn nhất trong C.

Ấm nhiệm vụ của bạn là viết một hàm trong thuật toán. Cho một đồ thị vô hướng G với trọng số của các cạnh. Ấm nhiệm vụ của bạn là xuất ra các cạnh có độ dài lớn nhất trong một vài chu trình của G.

Input

Input của mỗi test bắt đầu với n và m ($1 \leq n \leq 1000$; $0 \leq m \leq 25000$), trong đó n là số đỉnh và m là số cạnh của đồ thị. Tiếp theo đó là m dòng bao gồm bộ 3 số nguyên u,v và w với cạnh (u,v) có chiều dài w ($0 \leq w < 2^{31}$; $0 \leq u, v < n$). Input kết thúc với dòng có m=n=0. Giả định không có 2 cạnh nào có cùng độ dài và giữa hai đỉnh không có nhiều hơn một cạnh.

Output

Với mỗi test, in ra độ dài của tất cả các cạnh là cạnh dài nhất trong những chu trình của đồ thị đã cho. Độ dài của các cạnh được ghi theo thứ tự tăng dần và cách nhau bởi dấu cách. Luôn không thỏa mãn đề bài thì in ra forest.

Sample Input	Sample Output
3 3 0 1 1 1 2 2 2 0 3 4 5 0 1 1 1 2 2 2 3 3 3 1 4 0 2 0 3 1 0 1 1 0 0	3 2 4 forest

UV11857. CHẶNG XE

Bây giờ, nhiều nhà sản xuất xe ô tô phát minh những chiếc xe có khả năng chạy bằng điện chứ không dùng xăng. Ắc quy pin ô tô thường nặng và đắt, nên những người thiết kế phải hết sức cẩn trọng khi xác định dung tích của pin, và khoảng chạy của xe. Công việc của bạn là xác định khoảng chạy của xe ngắn nhất để xe có thể đi giữa hai thành phố bất kỳ liên kết bằng mạng lưới giao thông (đường là đường hai chiều). Mỗi thành phố chứa trạm nạp nhiên liệu để xe có thể nạp đầy điện tại đó. Xuyên suốt tuyến đường giữa hai thành phố, xe có thể đi qua các thành phố trung gian nhưng khoảng cách giữa 2 thành phố liên tiếp trên tuyến đường không được dài hơn khoảng chạy của xe..

INPUT

Input có dãy của mạng lưới giao thông. Dòng đầu tiên của mạng lưới chứa 2 số nguyên dương n và m , số thành phố và số con đường. ($n, m \leq 1000000$). Thành phố đánh số từ 0 đến $n - 1$. m dòng sau chứa 3 số nguyên khác 0. 2 số đầu tiên là 2 thành phố kết nối với nhau, số thứ ba là độ dài của con đường. Input kết thúc bởi dòng chứa 2 số 0.

OUTPUT

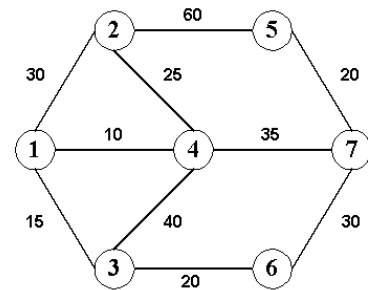
Xuất ra dòng ghi khoảng chạy nhỏ nhất của xe. Ắc quy không thể đi từ một thành phố nào đó sang một thành phố khác dù khoảng xe có lớn bao nhiêu đi chăng nữa thì in ra dòng chữ IMPOSSIBLE.

Sample Input	Sample Output
3 3 0 1 3 1 2 4 2 1 5 2 0 0 0	4 IMPOSSIBLE

UVA10099. HƯỚNG DẪN VIÊN DU LỊCH

Mr. Khờ làm hướng dẫn viên du lịch. Ắc hiệm vụ của anh là đưa đoàn du lịch từ thành phố này sang thành phố khác. Với mỗi cặp 2 thành phố cạnh nhau, có một tuyến xe bus chạy giữa 2 thành phố. Mỗi tuyến xe bus có giới hạn về số người tối đa trên một chuyến xe bus. Mr. Khờ có một bản đồ về quan hệ giữa các thành phố. Anh ấy hiểu rằng có lúc sẽ không thể nào đưa được toàn bộ đoàn khách du lịch đi từ thành phố đến thành phố khác trong một chuyến đi duy nhất. Với ví dụ dưới đây là đồ thị miêu tả các thành phố và trọng số trên các cạnh là số người đi tối đa đi trên tuyến

xe bus nối giữa hai thành phố, Muốn 99 du khách đi từ thành phố 1 đến thành phố 7 cần ít nhất 5 chuyến đi, và tuyến đường sẽ là 1- 2 - 4 -7. Mr. Kờ cần chọn tuyến đường sao cho tổng số chuyến đi nhỏ nhất.



INPUT

Input có nhiều test. Dòng đầu tiên chứa số nguyên n ($n \leq 100$) và số R thể hiện số thành phố và số đường nối các cạnh của đồ thị. R dòng sau mỗi dòng chứa 3 số nguyên $C1, C2, P$. $C1$ và $C2$ là tên các thành phố và P ($P > 1$) là số người cực đại đi được trên một chuyến xe bus nối hai thành phố. Tên thành phố là các số nguyên từ 1 đến n . Dòng thứ $(R + 1)$ chứa ba số nguyên: S, D, T là thành phố bắt đầu, thành phố cần đến và số khách du lịch. Input kết thúc bởi 2 số nguyên $n = R = 0$.

OUTPUT

Dòng đầu tiên ghi số thứ tự của test. Dòng sau in ra số nguyên là số chuyến đi nhỏ nhất.

Sample Input	Sample Output
7 10	Scenario #1
1 2 30	Minimum Number of Trips = 5
1 3 15	
1 4 10	
2 4 25	
2 5 60	
3 4 40	
3 6 20	
4 7 35	
5 7 20	
6 7 30	
1 7 99	
0 0	

UVA10147. ĐƯỜNG BỘ

Quốc đảo Flatopia rất bằng phẳng. Không may, hệ thống đường bộ của Flatopia chưa được tốt. ả hiện ra vấn đề này, chính quyền đã xây dựng một con số đường để kết nối các thị trấn quan trọng với nhau. Mặc dù vậy, vẫn có một số thị trấn chưa có đường bộ đi tới. Cần phải xây dựng thêm nhiều tuyến đường để có thể đi lại giữa hai thị trấn bất kỳ bằng đường bộ.

Các thị trấn của Flatopian được đánh số từ 1 đến n và thị trấn thứ i có tọa độ (x_i, y_i) . Mỗi tuyến đường sẽ nối 2 thị trấn với nhau. Tất cả các tuyến đường (bao gồm cả đường đã có và đường được xây dựng thêm) nằm theo đường thẳng nối hai thị trấn đầu mút, và độ dài tuyến đường là khoảng cách giữa hai thị trấn. Tất cả các tuyến đường có thể được sử dụng theo hai hướng. Các tuyến đường có thể cắt nhau, nhưng người đi đường chỉ có thể thay đổi tuyến đường ở thị trấn đầu mút của tuyến đường.

Chính quyền Flatopian muốn tối ưu hóa giá thành xây dựng. Mặc dù vậy, họ muốn đảm bảo mỗi thị trấn đều có thể tới được mọi thị trấn khác. Vì Flatopia bằng phẳng, số tiền bỏ ra làm đường luôn luôn tỷ lệ với độ dài. ả hư thế, chi phí tối thiểu ứng với tổng độ dài các tuyến đường nhỏ nhất.

INPUT

Dòng đầu tiên của input là số bộ test. Sau đó có một dòng trống và các test cách nhau bởi một dòng trống. Mỗi bộ test gồm 2 phần. Phần thứ nhất cho biết tất cả các thị trấn trên đất nước, và phần thứ 2 cho biết tất cả các tuyến

đường đã được xây dựng trước đó. Dòng đầu tiên của bộ test là một số nguyên a ($1 \leq a \leq 750$) là số thị trấn. Sau đó là a dòng, mỗi dòng ghi tọa độ của thị trấn cách nhau bởi dấu cách. Giá trị tuyệt đối của các tọa độ không lớn hơn 10000. Mỗi thị trấn có một vị trí xác định. Dòng tiếp theo bao gồm số nguyên M ($0 \leq M \leq 1000$), biểu thị số đường đã xây dựng. M dòng tiếp theo mỗi dòng có 1 cặp số nguyên cách nhau bởi dấu cách. 2 số nguyên này là thứ tự của 2 thị trấn mà nó kết nối. Mỗi cặp thị trấn được kết nối bởi đúng một con đường.

OUTPUT

Viết ra file output một dòng cho mỗi tuyến đường mới mà phải được xây dựng để kết nối tất cả các thị trấn với độ dài tối thiểu. Mỗi tuyến đường biểu diễn bằng cách in ra số thứ tự 2 thị trấn mà nó kết nối, cách nhau bởi dấu cách. Nếu không có tuyến đường mới cần phải xây dựng (Tất cả các thị trấn đã được kết nối), thì sau đó output file phải gồm một dòng ghi “ a o new highways need”. In ra dòng trống giữa các bộ test.

Sample input	Sample output
1	1 6
9	3 7
1 5	4 9
0 0	5 7
3 2	8 3
4 5	
5 1	
0 4	
5 2	
1 2	
5 3	
3	
1 3	
9 7	
1 2	

UVA10842. LUỒNG GIAO THÔNG

Một thành phố có n giao lộ và m con đường hai chiều nối mỗi cặp giao lộ. Mỗi con đường có một thông lượng nhất định - là số xe đi qua trong mỗi phút. Biết rằng có một tuyến đường nối giữa hai cặp giao lộ bất kỳ (tuyến đường là một chuỗi các con đường). Do hết kinh phí, Cục bảo hành đường bộ quyết định chấm dứt việc sử dụng một số con đường nhưng không làm phá vỡ liên kết giữa các giao lộ. Trong những con đường được giữ lại, Cục muốn thông lượng con đường bé nhất là lớn nhất có thể.

INPUT

Dòng đầu tiên là số test a . a test tiếp theo, mỗi test bắt đầu bằng n ($0 < n \leq 100$) và m ($0 < m \leq 10000$), m dòng tiếp theo sẽ diễn tả m con đường, mỗi dòng gồm 3 số nguyên, u , v và c ($0 \leq u, v < n$), ($0 < c \leq 1000$), u và v là 2 giao lộ đầu mút của con đường, c là thông lượng của nó.

OUTPUT

Với mỗi test in ra “Case # x :” là thông lượng nhỏ nhất của những con đường còn lại.

Sample Input	Sample Output
2	Case #1: 20
2 3	

0 1 10 0 1 20 0 0 30 4 5 0 1 1 3 1 2 1 2 3 2 3 4 0 2 5	Case #2: 3
--	------------

UVA534. ẾCH

Chú ếch Khờ đang ngồi trên một phiến đá ở giữa hồ. Bất ngờ nó thấy cô ếch Xờ cũng đang ngồi trên một phiến đá khác. Ếch Khờ quyết định tới thăm, nhưng nước hồ rất bẩn và đầy những con ma nước, nó muốn tránh việc bơi và thay thế bằng cách nhảy.

Tiếc thay, chỗ của Xờ nằm ngoài phạm vi nhảy của Khờ. Bởi vậy Khờ sẽ sử dụng các phiến đá khác làm trạm trung gian và đến chỗ Xờ bằng một dãy các bước nhảy.

Để thực hiện một chuỗi các bước nhảy, hiển nhiên độ dài bước nhảy của Khờ sẽ phải lớn hơn hoặc bằng khoảng cách dài nhất giữa hai phiến đá liên tiếp.

Khoảng cách ếch (loài người gọi là khoảng cách minimax) giữa hai phiến đá là độ dài bước nhảy tối thiểu trên mọi con đường nối hai phiến đá.

Bạn được cho tọa độ phiến đá của Khờ, Xờ và tất cả các phiến đá khác trong hồ. ả nhiệm vụ của bạn là tìm khoảng cách ếch từ chỗ của Khờ đến chỗ của Xờ.



INPUT

Dữ liệu gồm nhiều test. Dòng đầu mỗi test là số phiến đá n ($2 \leq n \leq 200$). ả dòng tiếp theo, mỗi dòng chứa 2 số nguyên x_i, y_i ($0 \leq x_i, y_i \leq 1000$) thể hiện tọa độ của phiến đá thứ i . Phiến đá thứ nhất là của Freddy, thứ hai là của Fiona, còn lại là $n-2$ phiến đá trống. Có một dòng trống sau mỗi test. Kết thúc bằng trường hợp $n=0$.

OUTPUT

Với mỗi test, in hai dòng “Scenario # x ” và “Frog Distance = y ”, trong đó x là thứ tự test (được đánh số từ 1) và y là số thực có 3 chữ số sau dấu phẩy. Thêm một dòng trống sau mỗi test, kể cả test cuối.

Sample Input	Sample Output
2 0 0 3 4 3 17 4 19 4 18 5 0	Scenario #1 Frog Distance = 5000 Scenario #2 Frog Distance = 1.414

UVA10369. MẠNG LƯỚI CĂN CỨ.

Bộ Quốc Phòng mong muốn kết nối các căn cứ quân sự ở phía Bắc với nhau bằng mạng không dây. Hai kỹ thuật kết nối khác nhau được sử dụng trong việc tổ chức mạng lưới: tất cả các căn cứ đều có máy truyền phát tín hiệu radio và một số căn cứ còn có kênh vệ tinh.

Bất kỳ 2 căn cứ nào – dù ở đâu nhưng nếu có kênh vệ tinh thì đều có thể kết nối với nhau qua vệ tinh. Mặt khác, 2 căn cứ chỉ có thể kết nối với nhau qua tín hiệu radio nếu khoảng cách giữa chúng không được vượt quá D . Để đơn giản hóa vấn đề, ta sẽ coi giá trị của D tại mọi căn cứ là giống nhau.

Ấm nhiệm vụ của bạn là hãy xác định giá trị D nhỏ nhất cho các máy thu phát tín hiệu radio để luôn luôn có thể kết nối 2 căn cứ bất kỳ (có hướng hoặc không có hướng).

Dòng đầu tiên là số α - số test. Dòng đầu tiên của mỗi test gồm : $1 \leq S \leq 100$: số kênh vệ tinh. $S \leq P \leq 500$: số căn cứ. P dòng sau, mỗi dòng là một cặp số (x, y) là tọa độ của căn cứ ($0 \leq x, y \leq 10,000$). Với mỗi test, chúng ta phải in ra giá trị nhỏ nhất của D (gồm 2 chữ số sau dấu phẩy).

Sample Input	Sample Output
1 2 4 0 100 0 300 0 600 150 750	212.13

UVA1216. LẮP MÁY THU PHÁT

Mr. Khờ làm chủ một nông trại có diện tích rất lớn bên Lâm Gia Thôn. Để giám sát số lượng sâu bọ trong nông trại, ông đã nhờ đến sự trợ giúp của Ms Xờ, một học sinh xuất sắc của HSGS.

Ms Xờ tìm ra một vài cách có hiệu quả ngay lập tức và đề nghị Mr. Khờ lắp một hệ thống cảm biến không dây. ả ó được lắp đặt như sau: Ở mỗi một địa điểm sẽ lắp đặt một máy cảm biến không dây. Bởi vì tất cả các máy cảm biến đều chạy bằng pin, nên năng lượng tiêu tốn sẽ được tính bằng khoảng cách truyền thông hữu dụng ECD (Effective Communication Distance) giữa 2 máy. Vì Mr. Khờ là một chàng trai tốt bụng nên không yêu cầu rắc rối với Ms Xờ, nên tất cả các máy cảm ứng được cài đặt có ECD như nhau.

ả ông trại lớn đến nỗi tất cả các máy cảm biến không thể phủ kín toàn bộ nông trại. Tuy nhiên, máy cảm biến A có thể phát dữ liệu thu được sang máy B nếu B nằm trong vùng ECD của A. ả hưng tổng số lượng máy cảm biến tương đối ít so với diện tích vùng đất rộng, mỗi ngày Mr. Khờ cần phải đi khắp nông trại để thu dữ liệu ở tất cả các máy cảm biến.

Mr. Khờ càng ngày càng lười. Thế nên ông mong rằng máy tính của mình có thể nhận được hết dữ liệu từ tất cả các máy cảm biến một cách tự động. Một lần nữa ông lại nhờ Ms Xờ giúp. Lần này Ms Xờ giúp Mr. Khờ lắp một trạm thu ở nhà. Điều đặc biệt là căn nhà nằm ở chính giữa nông trại.

Vì ngân sách eo hẹp, nên số lượng bộ phát dữ liệu mà Mr. Khờ có thể mua được tương đối ít so với số máy cảm biến. ả hụ vậy không thể lắp một máy phát cho một máy cảm biến. Vì vậy, máy cảm biến không có máy phát sẽ phải gửi dữ liệu sang máy cảm biến khác có máy phát. Sau đó bộ phận phát sẽ gửi dữ liệu về máy thu ở nhà của Mr. Khờ.

Ms Xờ hứa viết một chương trình làm cho tất cả máy cảm biến hoạt động theo cách trên, nhưng một vấn đề đã nảy sinh. ả ếu tất cả các máy đều được cài đặt ở mức năng lượng lớn nhất thì chắc chắn sẽ hết pin rất nhanh. Để tiết kiệm năng lượng Ms Xờ cần phải xác định mức năng lượng ít nhất.

Mặc dù Ms Xờ viết code rất giỏi nhưng xây dựng thuật toán thì không tốt lắm. Công việc của bạn là xác định số ECD có mức năng lượng thấp nhất.

Sau đây là ví dụ: Coi như vùng đất là 10×10 . Có 3 máy cảm ứng (1,1), (2,1), và (8,7). Đồng thời có 2 bộ phận thu nhận. Ở trong trường hợp này, $ECD=1$.

INPUT

Dòng đầu tiên là số test w $1 \leq w \leq 10$. Sau đó là w test. Trong mỗi test, dòng đầu tiên là 1 số nguyên biểu diễn số bộ máy phát. Sau đó, có 1 số dòng với 2 số nguyên có dạng $X Y$ cách nhau một dấu cách. Trong đó X chỉ tọa độ theo phương ngang và Y chỉ phương thẳng đứng của máy cảm ứng. Mỗi test kết thúc bởi dòng "-1". Giới hạn của vùng đất là 100000×100000

OUTPUT

Với mỗi test in ra số ECD

Sample Input	Sample Output
1 2 1 1 2 1 8 7 -1	1

UVA544. CHUYÊN TRỞ HÀNG HÓA

Big Johnsson Trucks Inc là công ty chuyên sản xuất ô tô tải. Model mới nhất của họ, Godzilla V12, lớn đến nỗi số lượng hàng hóa có thể chuyên chở được không bao giờ có thể lấp đầy thùng xe. Tuy nhiên mỗi con đường nối giữa hai thành phố có một trọng tải - trọng lượng hàng hóa tối đa trên một chiếc xe di chuyển trên con đường đó.

Cho thành phố xuất phát và thành phố đích, công việc của bạn là xác định số lượng hàng hóa tối đa mà Godzilla V12 có thể chở được trên tuyến đường nối giữa hai thành phố. Tuyến đường gồm nhiều con đường liên tiếp nhau.

INPUT

Input có nhiều test. Dòng đầu tiên của mỗi test sẽ là 2 số nguyên: n là số lượng thành phố ($2 \leq n \leq 200$) và r là trọng tải tối đa của các con đường.

Tiếp theo đó là r dòng, mỗi dòng miêu tả tên và trọng tải tối đa của những con đường nối giữa 2 thành phố. Tên không quá 30 kí tự và trong tên không có dấu cách. Trọng tải tối đa là số nguyên trong khoảng 0-10000. Các con đường có thể di chuyển 2 chiều.

Kết thúc Input khi $n=r=0$.

OUTPUT

Với mỗi test, in ra 3 dòng: 1 dòng "Scenario #x" với x là thứ tự test. 1 dòng "y tons" với y là số hàng hóa lớn nhất có thể vận chuyển. 1 dòng trống.

Sample Input	Sample Output
4 3 Karlsruhe Stuttgart 100 Stuttgart Ulm 80 Ulm Muenchen 120 Karlsruhe Muenchen	Scenario #1 80 tons Scenario #2 170 tons

5 5 Karlsruhe Stuttgart 100 Stuttgart Ulm 80 Ulm Muenchen 120 Karlsruhe Hamburg 220 Hamburg Muenchen 170 Muenchen Karlsruhe 0 0	
--	--