**NGÂN HÀNG CÂU HỎI THI THỰC HÀNH PHÒNG MÁY**

**Môn:** Cấu trúc dữ liệu và giải thuật. **Phần:** Đồ thị

Phần này sẽ có các nhóm bài chủ đạo sau:

1. DFS, BFS
2. DSU
3. Các thuật toán tìm đường đi ngắn nhất trên đồ thị có trọng số

**MỤC LỤC**

[**BÀI 1. DSA\_P011. ĐẾM SỐ VẬT CẢN TRÊN MÊ CUNG** 2](#_Toc205451498)

[**BÀI 2. DSA\_P071. LIÊN THÔNG MẠNH** 3](#_Toc205451499)

[**BÀI 3. DSA\_P072. ĐƯỜNG ĐI BFS** 3](#_Toc205451500)

[**BÀI 4. DSA\_P073. CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG** 4](#_Toc205451501)

[**BÀI 5. DSA\_P074. THÀNH PHẦN LIÊN THÔNG – 1** 5](#_Toc205451502)

[**BÀI 6. DSA\_P075. ĐI HỌC [VDC]** 5](#_Toc205451503)

[**BÀI 7. DSA\_P076. ĐỈNH TRỤ VÀ CẠNH CẦU** 7](#_Toc205451504)

[**BÀI 8. DSA\_P078. THÀNH PHẦN LIÊN THÔNG – 2** 8](#_Toc205451505)

[**BÀI 9. DSA\_P080. KHÔNG LIÊN THÔNG VỚI ĐỈNH 1** 8](#_Toc205451506)

[**BÀI 10. DSA\_P081. CHIA CẮT ĐỒ THỊ** 9](#_Toc205451507)

[**BÀI 11. DSA\_P082. ĐƯỜNG ĐI BFS** 10](#_Toc205451508)

[**BÀI 12. DSA\_P089. ĐỈNH THẮT** 10](#_Toc205451509)

[**BÀI 13. DSA\_P122. ĐƯỜNG ĐI TRUNG BÌNH NGẮN NHẤT** 11](#_Toc205451510)

[**BÀI 14. DSA\_P123. DI CHUYỂN TRÊN ĐỒ THỊ CÓ TRỌNG SỐ** 12](#_Toc205451511)

[**BÀI 15. DSA\_P125. BỔ SUNG CẠNH** 13](#_Toc205451512)

[**BÀI 16. DSA\_P127. ĐƯỜNG ĐI DFS – BFS** 13](#_Toc205451513)

[**BÀI 17. DSA\_P139. SỐ LẦN DUYỆT ÍT NHẤT** 14](#_Toc205451514)

[**BÀI 18. DSA\_P208. BÌNH THÔNG NHAU** 14](#_Toc205451515)

[**BÀI 19. DSA\_P209. BÀI TOÁN SO SÁNH** 15](#_Toc205451516)

[**BÀI 20. DSA\_P216. MÊ CUNG** 15](#_Toc205451517)

[**BÀI 21. KIỂM TRA ĐỒ THỊ Ơ-LE** 16](#_Toc205451518)

[**BÀI 22. ĐƯỜNG KÍNH CỦA CÂY** 17](#_Toc205451519)

[**BÀI 23. DSATest28. HỆ THỐNG GIAO THÔNG** 18](#_Toc205451520)

[**BÀI 24. DSA\_P305. XOÁ CẠNH** 19](#_Toc205451521)

[**BÀI 25. HÌNH CHỮ NHẬT NHỎ NHẤT** 21](#_Toc205451522)

[**BÀI 26. CÂY KHUNG NHỎ NHẤT** 22](#_Toc205451523)

# **BÀI 1. DSA\_P011. ĐẾM SỐ VẬT CẢN TRÊN MÊ CUNG**

Một mê cung được mô tả dưới dạng ma trận ký tự trong đó dấu ‘.’ là mô tả ô trống, không có vật cản, dấu ‘#’ mô tả một vật cản. Các vật cản sẽ ghép lại với nhau thành vật cản lớn hơn nếu nó liền kề theo hàng hoặc cột.

Hãy đếm xem trong mê cung có bao nhiêu vật cản.

**Input**

Dong đầu ghi số hai số N, M (N, M ≤ 1000) là số hàng và số cột của mê cung.

N dòng tiếp theo mô tả mê cung trong đó chỉ có các ký tự ‘.’ và ‘#’.

**Output**. Ghi ra số vật cản đếm được.

**Ví dụ**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5 6  .#....  ..#...  ..#..#  ...##.  .#.... | 5 |

**Phân dạng:** Đếm số TPLT

# **BÀI 2. DSA\_P071. LIÊN THÔNG MẠNH**

**Phân dạng:** Kosaraju

# **BÀI 3. DSA\_P072. ĐƯỜNG ĐI BFS**

Cho đồ thị có hướng G = <V, E> được biểu diễn dưới dạng danh sách cạnh.

Hãy tìm đường đi từ đỉnh u đến đỉnh v trên đồ thị bằng thuật toán BFS.

**Input:**

* Dòng đầu tiên đưa vào T là số lượng bộ test.
* Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm |E|+1 dòng:

+) Dòng đầu tiên đưa vào bốn số |V|, |E|, u, v tương ứng với số đỉnh, số cạnh, đỉnh xuất phát u, đỉnh kết thúc v;

+) |E| dòng tiếp theo mỗi dòng đưa vào bộ đôi x, y tương ứng với một cạnh của đồ thị.

* T, |V|, |E| thỏa mãn ràng buộc: 1 ≤ T ≤ 100; 1 ≤ |V| ≤ 103 ; 1 ≤ |E| ≤ |V| (|V| - 1) / 2;

**Output:**

Đưa ra đường đi từ đỉnh s đến đỉnh t của mỗi test theo thuật toán BFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  6 9 1 6  1 2  2 5  3 1  3 2  3 5  4 3  5 4  5 6  6 4 | 1 2 5 6 |

# **BÀI 4. DSA\_P073. CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG**

Cho đồ thị có hướng G = <V, E> được biểu diễn dưới dạng danh sách cạnh.

Kiểm tra xem đồ thị có tồn tại chu trình hay không?

**Input:**

* Dòng đầu tiên đưa vào T là số lượng bộ test.
* Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm |E| + 1 dòng:

Dòng đầu tiên đưa vào hai số |V|, |E| tương ứng với số đỉnh, số cạnh của đồ thị;

E dòng tiếp theo, mỗi dòng gồm 2 số x và y là điểm đầu và điểm cuối của cạnh đồ thị

* T, |V|, |E| thỏa mãn ràng buộc: 1≤T≤100; 1≤|V|≤103; 1≤|E|≤|V|(|V|-1)/2;

**Output:**

* Đưa ra YES hoặc “NO” kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  6 9  1 2  2 4  3 1  3 2  3 5  4 3  5 4  5 6  6 4 | YES |

# **BÀI 5. DSA\_P074. THÀNH PHẦN LIÊN THÔNG – 1**

Cho đồ thị vô hướng G có N đỉnh, M cạnh. Hãy liệt kê các đỉnh không cùng thành phần liên thông với một đỉnh cho trước.

**Input.**

Dòng đầu ghi 3 số N, M và X (0 < N < 300; 1 ≤ M ≤ N \* (N - 1) / 2), 0 < X < N).

Tiếp theo là M dòng, mỗi dòng ghi một cạnh của đồ thị.

Các cạnh được liệt kê với thứ tự bất kỳ.

**Output.**

Ghi ra các đỉnh không liên thông với đỉnh X theo thứ tự tăng dần, mỗi dòng ghi một đỉnh. Nếu không có đỉnh nào thì ghi ra số 0.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6 4 2  1 3  2 3  1 2  4 5 | 4  5  6 |

# **BÀI 6. DSA\_P075. ĐI HỌC [VDC]**

Tí và Tèo là bạn học cùng nhau hồi cấp 1. Lên cấp 2, Tí chuyển nhà nên phải học ở trường của xã bên cạnh. Tuy nhiên, 2 bạn vẫn rất thân với nhau và hay thường đợi nhau cùng đi học, cho dù chỉ đi chung 1 quãng đường hoặc gặp mặt nhau một cái.

Cho biết địa điểm nhà Tí là H1, trường của Tí là S1, nhà của Tèo là H2, trường của Tèo là S2 (4 địa điểm này khác nhau). Có M tuyến đường hai chiều biểu diễn mô hình giao thông tại địa phương của hai bạn. Biết rằng Tí và Tèo luôn đi theo con đường ngắn nhất để đến trường, và tốc độ của 2 bạn bằng nhau (1 đơn vị độ dài / 1 đơn vị thời gian). Hai bạn xuất phát cùng một lúc và phải di chuyển liên tục, trừ trường hợp đã tới trường học của mình.

Các bạn hãy xác định xem Tí và Tèo có thể gặp nhau tại bao nhiêu địa điểm? Trường hợp Tèo đã đến trường, mà trường học của Tèo lại nằm trên đường đi học của Tí, thì bạn ấy sẽ đứng đợi ở trường cho tới khi gặp được Tí đi học qua (và ngược lại).

**Input.**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 20).
* Mỗi test bắt đầu bởi hai số nguyên N, M (N, M ≤ 500 000).
* Tiếp theo là 4 số nguyên H1, S1, H2, S2.
* M dòng tiếp theo, mỗi dòng gồm 3 số nguyên u, v, c cho biết có cạnh nối giữa u và v, độ dài bằng c (u != v, c ≤ 109). Input đảm bảo đồ thị đã liên thông.

**Output.** Với mỗi test, in ra số lượng địa điểm có thể là nơi hai bạn sẽ gặp nhau trên đường đi học. Nếu đáp số là vô hạn, in ra “infinity”

**Ví dụ.**

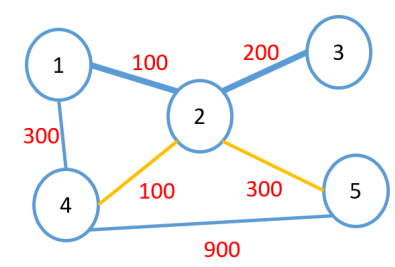
|  |  |
| --- | --- |
| **Input** | **Output** |
| 4  5 6  1 3 4 5  1 2 100  2 3 200  1 4 300  4 2 100  2 5 300  4 5 900  3 2  1 3 2 3  1 3 100  2 3 200  5 6  1 3 4 5  1 2 100  2 3 200  1 3 200  4 2 100  2 5 300  4 5 400  6 6  1 5 2 6  1 3 100  2 3 100  3 4 200  4 5 100  4 6 300 | 1  1  0  infinity |

**Giải thích test.**

Giải thích test 1: Điểm chung duy nhất là đỉnh 2.

Giải thích test 2: Đây là trường hợp đứng đợi tại trường học của bạn thứ nhất.

Giải thích test 4: 2 bạn đi chung trên con đường 3 trùng 4, và có vô số điểm chung nhau.

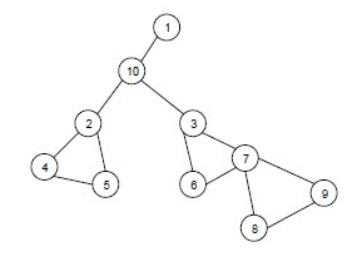
****

# **BÀI 7. DSA\_P076. ĐỈNH TRỤ VÀ CẠNH CẦU**

Cho đồ thị vô hướng có N đỉnh (1 ≤ N ≤ 104) và M cạnh (1 ≤ M ≤ 105). Ta nhớ lại các định nghĩa sau:

* Một đỉnh được gọi là đỉnh trụ nếu như xóa đỉnh đó đi sẽ làm tăng số lượng thành phần liên thông của đồ thị.
* Một cạnh được gọi là cạnh cầu nếu như xóa cạnh đó đi sẽ làm tăng số lượng thành phần liên thông của đồ thị.

Nhiệm vụ của bạn là hãy đếm số lượng đỉnh trụ và cạnh cầu của đồ thị đã cho.



**Input:**

* Dòng đầu tiên hai số nguyên N và M.
* M dòng tiếp theo, mỗi dòng gồm hai số nguyên u, v cho biết có cạnh nối giữa đỉnh u và v.

**Output.** In ra 2 số nguyên là số lượng đỉnh trụ và cạnh cầu của đồ thị

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10 12  1 10  10 2  10 3  2 4  4 5  5 2  3 6  6 7  7 3  7 8  8 9  9 7 | 4 3 |

# **BÀI 8. DSA\_P078. THÀNH PHẦN LIÊN THÔNG – 2**

Cho đồ thị vô hướng gồm N đỉnh và M cạnh. Xét lần lượt các đỉnh từ 1 đến N, hãy tính xem nếu xóa đỉnh đó đi thì đồ thị có bao nhiêu thành phần liên thông.

**Input.**

* Dòng đầu ghi 2 số N và M (1 < N < 20000; 1 < M < 50000).
* Tiếp theo là M dòng ghi các cạnh của đồ thị.

**Output.**

In ra N dòng, dòng thứ k (tính từ 1) là số thành phần liên thông của đồ thị nếu xóa đỉnh K

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 3  1 2  2 3  2 4 | 1  3  1  1 |

# **BÀI 9. DSA\_P080. KHÔNG LIÊN THÔNG VỚI ĐỈNH 1**

Cho đồ thị vô hướng G có N đỉnh, M cạnh. Hãy liệt kê các đỉnh không cùng thành phần liên thông với đỉnh 1.

**Input.**

* Dòng đầu ghi 2 số N và M (0 < N < 300; 1 ≤ M ≤ N \* (N - 1) / 2)).
* Tiếp theo là M dòng, mỗi dòng ghi một cạnh của đồ thị. Các cạnh được liệt kê với thứ tự bất kỳ.

**Output.**

Ghi ra các đỉnh không liên thông với đỉnh 1 theo thứ tự tăng dần, mỗi dòng ghi một đỉnh. Nếu không có đỉnh nào thì ghi ra số 0.  
**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6 4  1 3  2 3  1 2  4 5 | 4  5  6 |

# **BÀI 10. DSA\_P081. CHIA CẮT ĐỒ THỊ**

Cho đồ thị vô hướng G có N đỉnh và M cạnh. Hãy tìm đỉnh u sao cho nếu loại bỏ đỉnh u ra khỏi đồ thị thì đồ thị bị chia cắt thành nhiều thành phần liên thông nhất.

**Input.**

Dòng đầu ghi số bộ test, mỗi bộ test gồm:

* Dòng đầu ghi số N là số đỉnh (1 < N < 100) và số M là số cạnh (M < N\*(N-1)/2).
* M dòng tiếp theo ghi các cạnh của đồ thị.

**Output.**

* Ghi ra thứ tự đỉnh (tính từ 1) thỏa mãn nếu loại bỏ đỉnh đó ra khỏi đồ thị thì sẽ chia cắt ra nhiều thành phần liên thông nhất.
* Nếu có nhiều hơn 1 đỉnh thỏa mãn thì in ra thứ tự đỉnh nhỏ nhất.
* Nếu không thể chia cắt được đồ thị thì ghi ra 0.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  5 5  1 2  1 3  2 3  3 4  3 5  5 7  1 2  1 3  2 3  2 5  3 4  3 5  4 5 | 3  0 |

# **BÀI 11. DSA\_P082. ĐƯỜNG ĐI BFS**

Cho đồ thị vô hướng G = <V, E> được biểu diễn dưới dạng danh sách cạnh.

Hãy tìm đường đi từ đỉnh s đến tất cả các đỉnh còn lại bằng thuật toán BFS.

**Input:**

* Dòng đầu tiên đưa vào T là số lượng bộ test.
* Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng:

+) Dòng đầu tiên đưa vào ba số N, M, s tương ứng với số đỉnh, số cạnh, đỉnh xuất phát. +) Dòng tiếp theo đưa vào các bộ đôi u, v tương ứng với một cạnh của đồ thị.

* T, N, M thỏa mãn ràng buộc: 1 ≤ T ≤ 100; 1 ≤ N ≤ 103 ; 1≤ M ≤ N \* (N - 1) / 2;

**Output**

* Đưa ra đường đi từ đỉnh s đến lần lượt từng đỉnh từ 1 đến N (trừ chính đỉnh s).
* Nếu đỉnh nào không có đường đi từ s thì ghi ra “No path”

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  6 9 1  1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6 | 1 2  1 3  1 3 4  1 2 5  1 2 5 6 |

# **BÀI 12. DSA\_P089. ĐỈNH THẮT**

Cho đồ thị có hướng liên thông G có N đỉnh và M cạnh. Với một cặp đỉnh (u,v), đỉnh thắt của cặp đỉnh này được định nghĩa là một đỉnh mà tất cả đường đi từ u tới v đều đi qua nó. Hãy đếm số đỉnh thắt với cặp đỉnh (u,v).

**Input.**

* Dòng đầu ghi số bộ test, không quá 100.
* Mỗi bộ test bắt đầu với một dòng ghi 4 số N, M, u, v

(0 < N ≤ 100; 1 < M ≤1000; 1 ≤ u,v ≤ N).

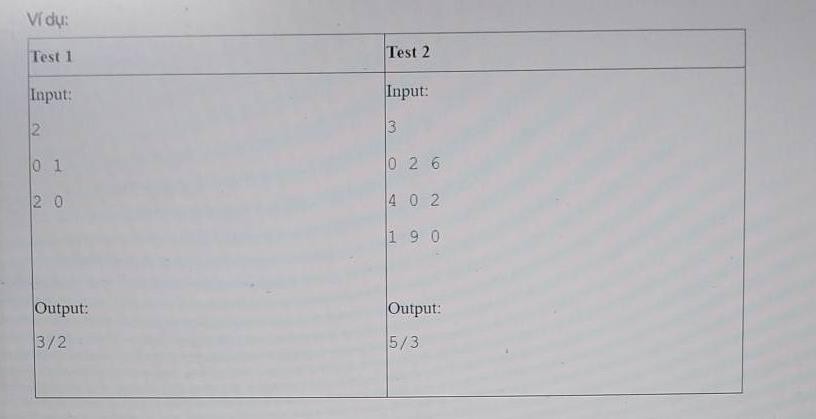
* Tiếp theo là M dòng ghi các cạnh của đồ thị

**Output.** Với mỗi bộ test, ghi ra số đỉnh thắt của cặp đỉnh (u,v)

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  5 7 1 3  1 2  2 4  2 5  3 1  3 2  4 3  5 4  4 5 1 4  1 2  1 3  2 3  2 4  3 4 | 2  0 |

# **BÀI 13. DSA\_P122. ĐƯỜNG ĐI TRUNG BÌNH NGẮN NHẤT**



# **BÀI 14. DSA\_P123. DI CHUYỂN TRÊN ĐỒ THỊ CÓ TRỌNG SỐ**

# **BÀI 15. DSA\_P125. BỔ SUNG CẠNH**

Cho đồ thị vô hướng G có thể chưa liên thông. Từ đỉnh 1, bằng cách duyệt đồ thị ta sẽ đánh dấu được các đỉnh cùng thành phần liên thông với đỉnh 1.

Người ta muốn đánh dấu được nhiều đỉnh hơn bằng cách bổ sung một cạnh duy nhất vào đồ thị. Hãy đếm số đỉnh nhiều nhất có thể đánh dấu theo mô tả trên. **[DSU]**

**Input.**

* Dòng đầu ghi 2 số N và M là số đỉnh và số cạnh của đồ thị. (1 ≤ N,M ≤ 105).
* M dòng tiếp theo, mỗi dòng ghi một cạnh của đồ thị.

**Output.**

Đưa ra số lượng đỉnh lớn nhất có thể được đánh dấu bằng cách đi từ đỉnh 1 nếu bổ sung một cạnh vào đồ thị.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3 2  1 2  3 2 | 3 |
| 5 3  1 4  4 2  2 1 | 4 |

# **BÀI 16. DSA\_P127. ĐƯỜNG ĐI DFS – BFS**

Cho đồ thị vô hướng G có N đỉnh và M cạnh. Hãy tìm và in ra:

* Đường đi từ đỉnh 1 đến các đỉnh khác bằng thuật toán DFS
* Đường đi ngược lại từ các đỉnh khác về 1 bằng thuật toán BFS.

**Input.**

* Dòng đầu ghi 2 số N và M (1 ≤ N ≤ 1000; 1 ≤ M ≤ 2000).
* M dòng tiếp theo, mỗi dòng ghi một cạnh vô hướng của đồ thị.

**Output.** Xét từng đỉnh từ 2 đến N, với mỗi đỉnh ghi ra 2 dòng:

* Dòng đầu ghi đường đi từ 1 đến đỉnh đó theo DFS
* Dòng tiếp theo ghi đường đi từ đỉnh đó về 1 theo thuật toán BFS

Nếu không có đường đi thì ghi ra -1

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 4  1 2  1 3  1 4  3 4 | 1 2  2 1  1 3  3 1  1 3 4  4 1 |

# **BÀI 17. DSA\_P139. SỐ LẦN DUYỆT ÍT NHẤT**

Cho đồ thị có hướng với N đỉnh và M cạnh. Mỗi lần duyệt đồ thị, bạn có thể lựa chọn bắt đầu từ bất cứ đỉnh nảo và cố gắng đi qua nhiều đỉnh nhất có thể (đỉnh nào đã đi qua sẽ được đánh dấu). Hãy tính xem cần ít nhất bao nhiêu lần duyệt để đi qua tất cả các đỉnh của đồ thị.

**Input.**

* Dòng đầu ghi 2 số N và M (1 ≤ N ≤ 1000; 0 ≤ M ≤ 10000)
* M dòng sau ghi các cạnh có hướng của đồ thị.

**Output.** Ghi ra một số nguyên duy nhất là số lần duyệt ít nhất tính được.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5 4  1 2  1 3  4 1  5 1 | 2 |

# **BÀI 18. DSA\_P208. BÌNH THÔNG NHAU**

Có N thùng nước được đánh số từ 1 đến N, giữa 2 thùng bất kỳ đều có một ống nối có một van có thể khóa hoặc mở. Ở trạng thái ban đầu tất cả các van đều đóng.

Bạn được cho một số yêu cầu, trong đó mỗi yêu cầu có 2 dạng:

* Dạng X Y 1 có ý nghĩa là bạn cần mở van nối giữa 2 thùng X và Y.
* Dạng X Y 2 có ý nghĩa là bạn cần cho biết với trạng thái các van đang mở / khóa như hiện tại thì 2 thùng X và Y có thuộc cùng một nhóm bình thông nhau hay không?

Hai thùng được coi là thuộc cùng một nhóm bình thông nhau nếu nước từ bình nàycó thể chảy đến được bình kia qua một số ống có van đang mở. **[DSU[**

**Input.**

Dòng đầu tiên là số lượng truy vấn Q (Q ≤ 105).

Mỗi truy vấn gồm 3 số nguyên X, Y, Z (X, Y ≤ 105).

**Output.** Với mỗi truy vấn, in ra đáp án tìm được trên một dòng.

**Ví du.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 9  1 2 2  1 2 1  3 7 2  2 3 1  1 3 2  2 4 2  1 4 1  3 4 2  1 7 2 | 0  0  1  0  1  0 |

# **BÀI 19. DSA\_P209. BÀI TOÁN SO SÁNH**

Cho một dãy các phép so sánh chiều cao giữa các sinh viên, mỗi sinh viên được đại diện bởi tên sinh viên (Dãy ký tự không có khoảng trống và không có 2 sinh viên nào trùng tên). Hãy chỉ ra liệu có thể tất cả các phép so sánh đó đều đúng hay không? **[Kahn, chu trình trên đồ thị có hướng]**

**Input**.

* Dòng đầu ghi số N là số phép so sánh. (1 ≤ N ≤ 105).
* Mỗi phép so sánh gồm 2 xâu ký tự đại diện cho 2 cái tên sinh viên, và một dấu lớn hơn hoặc nhỏ hơn, hai xâu ký tự đều không quá 20 chữ cái, không có khoảng trống.

**Output**

Ghi ra “possible” nếu tất cả phép so sánh đều có thể đúng hoặc “impossible” nếu ngược lại.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  An > Binh  Binh > Cong  An < Cong | impossible |
| 3  An > Binh  Binh > Cong  An > Cong | possible |

# **BÀI 20. DSA\_P216. MÊ CUNG**

Mê cung có N cửa vào đánh số từ 1 đến N. Với thiết kế mới, người chơi xuất phát từ cửa này thì đi lòng vòng 1 hồi sẽ đến một cửa khác, sau đó có thể từ đó đi tiếp. Một người chơi thắng cuộc nếu chọn được một cửa xuất phát nào đó mà sau một số lần di chuyển (không đi lại con đường cũ) sẽ quay lại được đúng cửa ban đầu.

Bạn là người thiết kế mê cung nên bạn cần kiểm tra tính khả thi của trò chơi. Tức là bạn sẽ phải thiết kế trước các tuyến đường giữa các cửa của mê cung sao cho phải có ít nhất một cách di chuyển nào đó giúp người chơi thắng cuộc. Cho dù thực tế việc di chuyển trong mê cung không hề dễ dàng.

Cho trước danh sách các tuyến đường. Hãy kiểm tra xem liệu có thể có cách nào giúp người chơi thắng cuộc được hay không.

**Input**

- Dòng đầu ghi số bộ test.

- Với mỗi bộ test:

• Dòng đầu ghi 2 số N và M (1 ≤ N, M ≤ 105).

• M dòng tiếp theo, mỗi dòng ghi một tuyến đường trong thiết kế của mê cung.

**Output.**

Với mỗi bộ test, ghi ra YES nếu có tuyến đường giúp người chơi thắng cuộc. Ghi ra NO nếu ngược lại

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  4 3  1 2  1 4  3 4  4 4  1 2  1 3  1 4  2 4 | NO  YES |

# **BÀI 21. KIỂM TRA ĐỒ THỊ Ơ-LE**

# **BÀI 22. ĐƯỜNG KÍNH CỦA CÂY**

Cho đồ thị vô hướng liên thông có N đỉnh và N – 1 cạnh. Tính đường đi dài nhất giữa 2 đỉnh bất kỳ trong đồ thị.

**Input.**

* Dòng đầu tiên là số N (1 ≤ N ≤ 105) là số đỉnh của đồ thị
* N dòng tiếp theo, mỗi dòng là bộ đôi 2 số x, y mô tả cạnh của đồ thị

**Ouput.** In ra độ dài đường đi dài nhất giữa 2 đỉnh bất kỳ trên đồ thị

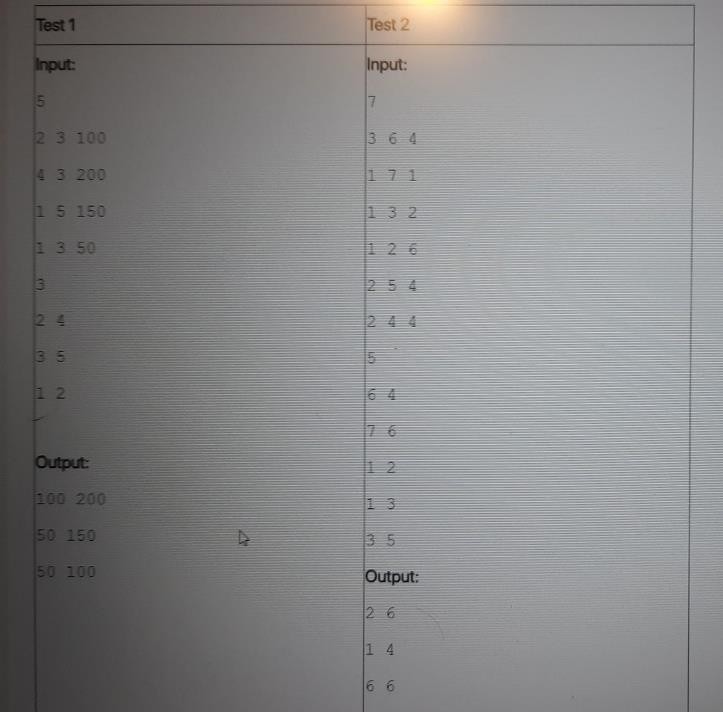
**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  1 2  2 3  2 4  1 5 | 3 |

**Giải thích test.**

Có 2 đường đi dài nhất: 5 🡪 1 🡪 2 🡪 3, 5 🡪 1 🡪 2 🡪 4. Chúng đều có độ dài là 3.

# **BÀI 23. DSATest28. HỆ THỐNG GIAO THÔNG**



# **BÀI 24. DSA\_P305. XOÁ CẠNH**

Cho đồ thị vô hướng có N đỉnh và M cạnh. Biết rằng các đỉnh và các cạnh được đánh số từ 1. **Yêu cầu:** Cho q truy vấn, mỗi truy vấn gồm 2 số x và y. Bạn cần kiểm tra xem sau khi xoá cạnh thứ x thì còn bao nhiêu đỉnh cùng thành phần liên thông với đỉnh y

**Lưu ý:**

* Tất cả các thao tác đều chỉ được thực hiện lần lượt trên cùng đồ thị ban đầu.
* Sau khi xoá cạnh xong thì đồ thị không được khôi phục như ban đầu nữa.
* Trường hợp thao tác xoá bị trùng lặp, tức là cạnh ấy đã xoá trước đó rồi nhưng bây giờ lại chọn nó để xoá, thì coi như không có thao tác đó, vẫn trả lời truy vấn bình thường

**Input**

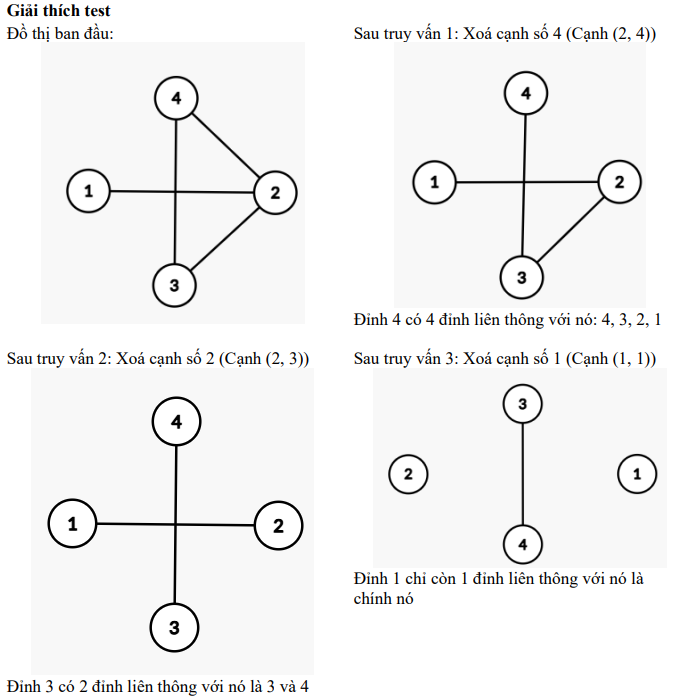
* Dòng đầu tiên ghi 3 số N, M, Q. Với N là số đỉnh, M là số cạnh, Q là số truy vấn
* Tiếp theo là M dòng ghi M cạnh của đồ thị
* Tiếp theo là Q dòng, mỗi dòng gồm 2 số x và y. Trong đó, x là số thứ tự cạnh bị xoá, y là đỉnh cần kiểm tra xem TPLT chứa nó còn lại bao nhiêu đỉnh.

**Ouput.** Với mỗi truy vấn, in ra số đỉnh thuộc thành phần liên thông của đỉnh y trên 1 dòng

**Giới hạn.** 1 ≤ N ≤ 100000, 1 ≤ M ≤ 500000, 1 ≤ Q ≤ 100000

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 4 3  1 2  2 3  3 4  2 4  4 4  2 3  1 1 | 4  2  1 |

****

# **BÀI 25. HÌNH CHỮ NHẬT NHỎ NHẤT**

Cho N điểm trên mặt phẳng Oxy và M đoạn thẳng nối các điểm này với nhau. Nhiệm vụ của bạn là cần tìm ra 1 hình chữ nhật có các cạnh song song với 2 trục toạ dộ Ox và Oy, có chu vi nhỏ nhất và che phủ trọn vẹn được 1 tập hợp các điểm liên thông với nhau và tính chu vi của hình chữ nhật đó.

**Lưu ý:** Trường hợp tìm được hình chữ nhật bao phủ là 1 đoạn thẳng (Với các điểm thẳng hàng, chiều còn lại bằng 0) thì kết quả ấy vẫn được công nhận. Đáp án là 2 lần độ dài đoạn thẳng đó.

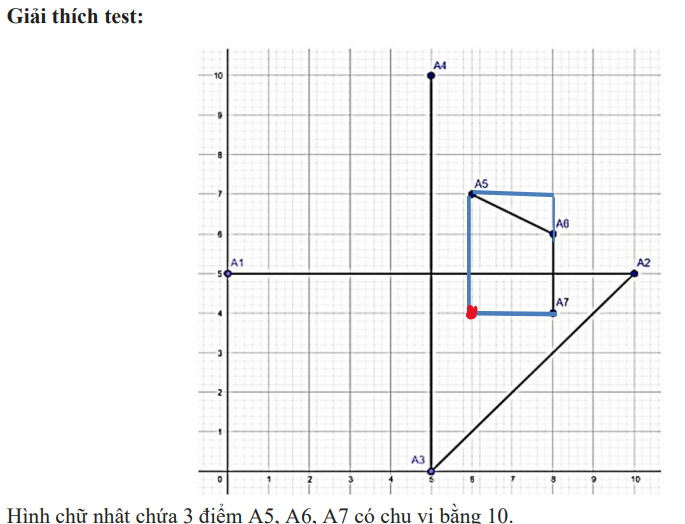
**Input:**

* Dòng đầu tiên gồm hai số nguyên N và M (2 ≤ N, M ≤ 100 000).
* N dòng tiếp theo, mỗi dòng gồm 2 số nguyên X[i] và Y[i] mô tả tọa độ của một điểm. Các tọa độ nằm trong phạm vi [0, 109].
* Theo sau đó là M dòng, mỗi dòng gồm 2 số nguyên u và v, mô tả một cạnh nối điểm u tới điểm v.

**Output:** In ra chu vi của hình chữ nhật nhỏ nhất tìm được

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 7 5  0 5  10 5  5 0  5 10  6 7  8 6  8 4  1 2  2 3  3 4  5 6  7 6 | 10 |



# **BÀI 26. CÂY KHUNG NHỎ NHẤT**

Trong đồ thị vô hướng, liên thông có N đỉnh, M cạnh với trọng số không âm, cây khung nhỏ nhất (minimum spanning tree) là đồ thị con chứa vừa đủ N-1 cạnh, vẫn đảm bảo tính liên thông và tổng trọng số nhỏ nhất có thể.

Bài toán đặt ra là hãy tìm và loại bỏ duy nhất 1 cạnh sao cho giá trị tổng trọng số của cây khung nhỏ nhất là lớn nhất có thể. Giả sử đồ thị luôn đảm bảo nếu loại bỏ 1 cạnh bất kỳ thì vẫn liên thông.

**Input.**

* Dòng đầu ghi 2 số N và M (3 ≤ N ≤ 105 ; 3 ≤ M ≤ 106 ). Các đỉnh đánh số từ 1 đến N.
* M dòng tiếp theo, mỗi dòng ghi 3 số u, v, w, tương ứng là đỉnh đầu, đỉnh cuối, và trọng số từng cạnh. (1 ≤ u < v ≤ N ; 1 ≤ w ≤ 106).

**Output.** Ghi ra tổng trọng số của cây khung nhỏ nhất sau khi đã xóa 1 cạnh như mô tả trên.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5 7  2 5 8  1 3 19  4 5 9  1 5 15  1 2 14  3 4 16  2 4 15 | 54 |
| 4 5  2 3 5  1 2 2  1 3 4  1 4 2  3 4 3 | 10 |