**SẮP XẾP TOPO**

**TRÊN ĐỒ THỊ CÓ HƯỚNG KHÔNG CHU TRÌNH (DAG)**

# ĐƯỜNG ĐI DÀI NHẤT 1

Cho đồ thị có hướng, không chu trình G gồm n đỉnh và m cạnh. Các đỉnh được đánh thứ tự từ 1 đến n.

***Yêu cầu:*** Tìm đường đi qua nhiều cạnh nhất trong G.

***Dữ liệu:*** Vào từ file LPATH1.INP bao gồm:

- Dòng đầu tiên gồm 2 số nguyên n, m (1≤n, m ≤ 105).

- m dòng tiếp theo, mỗi dòng ghi 2 số nguyên dương u, v thể hiện có một cung một chiều nối từ đỉnh u đến đỉnh v (1≤u,v ≤ n).

***Kết quả:*** Ghi ra file văn bản LPATH1.OUT gồm 1 số nguyên duy nhất là kết quả bài toán.

***Ví dụ:***

|  |  |
| --- | --- |
| **LPATH1.INP** | **LPATH1.OUT** |
| 5 8  5 3  2 3  2 4  5 2  5 1  1 4  4 3  1 3 | 3 |

# ĐƯỜNG ĐI DÀI NHẤT 2

Cho đồ thị có hướng, không có chu trình, có trọng số và 2 đỉnh s,t. Tìm đường đi ngắn nhất từ s tới t.

***Dữ liệu:*** Vào file LPATH2.INP gồm có:

- Dòng đầu tiên ghi hai số nguyên dương n, m (1≤n ≤ 3.105, 1≤ m ≤5.105) lần lượt là số đỉnh và số cạnh của đồ thị. Các đỉnh đánh số từ 1 đến n

- Dòng thứ hai ghi hai số nguyên dương s, t (1≤s, t≤n, s≠t)

- m dòng tiếp theo, mỗi dòng ghi 3 số nguyên dương u, v, c (1≤u, v≤n, u ≠v, |c| ≤103) thể hiện có một cung một chiều nối từ đỉnh u đến đỉnh v có trọng số là c.

- Dữ liệu đảm bảo rằng đồ thị không có chu trình.

***Kết quả:*** Ghi file LPATH2.OUT gồm có gồm 1 số duy nhất là độ dài của đường đi ngắn nhất từ s đến t. Nếu không tồn tại đường đi như vậy thì ghi -1

***Ví dụ:***

|  |  |
| --- | --- |
| **LPATH2.INP** | **LPATH2.OUT** |
| 5 6  1 5  1 4 5  4 5 -4  1 2 1  2 3 2  3 5 3  1 3 -7 | 6 |

# TÌM TỔNG SỐ ĐƯỜNG ĐI

Cho đồ thị có hướng, không có chu trình, có trọng số và 2 đỉnh s,t. Tìm tổng số đường đi từ s tới t.

***Dữ liệu:*** Vào file SPATH.INP gồm có:

- Dòng đầu tiên ghi hai số nguyên dương n, m (1≤n ≤ 3.105, 1≤ m ≤5.105) lần lượt là số đỉnh và số cạnh của đồ thị. Các đỉnh đánh số từ 1 đến n

- Dòng thứ hai ghi hai số nguyên dương s, t (1≤s, t≤n, s≠t)

- m dòng tiếp theo, mỗi dòng ghi 2 số nguyên dương u, v (1≤u, v≤n, u ≠v, |c| ≤103) thể hiện có một cung một chiều nối từ đỉnh u đến đỉnh v.

***Kết quả:*** Ghi file SPATH.OUT gồm có gồm 1 số duy nhất là số lượng đường đi từ s đến t. Vì con số này có thể rất lớn nên chỉ cần lấy phần dư của chúng khi chia cho 109+7

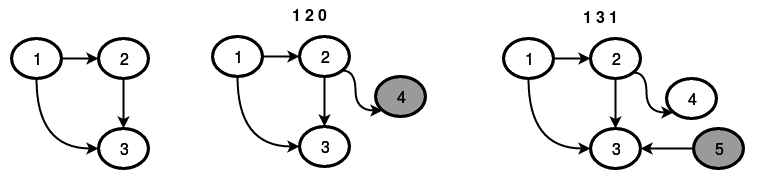
***Ví dụ:***

|  |  |
| --- | --- |
| **SPATH.INP** | **SPATH.OUT** |
| 5 6  1 5  1 4  4 5  1 2  2 3  3 5  1 3 | 3 |

# VÙNG ĐẤT MỚI

Một đất nước có n các thành phố được nối với nhau bằng m con đường một chiều. Các thành phố được đánh số từ 1 đến n. Gần đây chính phủ quyết định xây dựng một số thành phố mới. Để thực hiện cần có q công việc. Mỗi công việc được thể hiện một trong 2 hình thức sau:

- *1 x d*: thành phố n+1 được xây dựng và kết nối với thành phố x. Nếu d =0, chiều con đường mới đi từ thành phố x đến thành phố n+1; nếu d=1 thì có chiều ngược lại



- *2 x y*: Ghi ‘Yes’ nếu tồn tại đường đi từ x đến y. Ghi ‘No’ cho trường hợp ngược lại

***Yêu cầu:*** Xác định có tồn tại đường đi từ x đến y hay không trong công việc ở dạng 2

***Dữ liệu:*** file EPATH.INP

- Dòng đầu tiên gồm 2 số nguyên n, m (1≤n,m ≤ 5.104).

- m dòng tiếp theo, mỗi dòng ghi 2 số nguyên dương u, v thể hiện có một cung một chiều nối từ đỉnh u đến đỉnh v (1≤u,v ≤ n)

- Dòng tiếp theo ghi số nguyên dương q (1≤q ≤ 105)

- q dòng còn lại, mỗi dòng cho biết hình thức của mỗi công việc (d∈{0,1})

***Kết quả:*** file EPATH.OUT cho biết kết quả của các công việc ở dạng 2.

**Ví dụ:**

|  |  |
| --- | --- |
| **EPATH.INP** | **EPATH.OUT** |
| 4 4  1 2  1 3  2 4  3 4  5  1 2 0  2 3 5  2 1 5  1 1 1  2 6 4 | NO  YES  YES |

# CẮT CÂY

Cho một đồ thị dạng cây gồm đỉnh đánh số từ 1 đến , đỉnh có trọng số . Cho lần cắt cây (mỗi lần xóa một cạnh hiện đang có) ta sẽ có một rừng gồm cây. Trọng số của một cây được định nghĩa bằng tổng trọng số các đỉnh có trong cây đó.

Bài toán yêu cầu hãy tìm cách cắt cây sao cho trọng số lớn nhất của các cây sinh ra là nhỏ nhất

***Dữ liệu:***

- Dòng đầu tiên gồm hai số )

- Dòng thứ hai ghi số là trọng số các đỉnh (

- dòng tiếp theo, mỗi dòng gồm hai số mô tả một cạnh của cây

***Kết quả:*** Một số nguyên duy nhất là trọng số của cây có trọng số lớn nhất trong phương án tối ưu

***Ví dụ:***

|  |  |
| --- | --- |
| **TRCUT.INP** | **TRCUT.OUT** |
| 8 2  7 4 3 8 5 7 5 4  2 1  3 1  4 3  5 2  6 1  7 6  8 1 | 20 |

# BÀI TẬP TỰ LUYỆN

<https://www.spoj.com/problems/DAGCNT/>

<https://www.spoj.com/problems/TOPOSORT/>

<https://codeforces.cc/gym/101102/problem/K>

<https://codeforces.com/problemset/problem/770/C>

<https://codeforces.com/contest/919/problem/D>