

101. ĐƯỜNG ĐI THỨ TỰ TỪ ĐIỂM NHỎ NHẤT

Hệ thống giao thông của một thành phố gồm n địa điểm đánh số từ 1 tới n và m con đường một chiều giữa các địa điểm đó, con đường thứ i cho phép đi từ địa điểm u_i tới địa điểm v_i .

Một dãy các địa điểm $P = \langle p_1, \dots, p_k \rangle$ sao cho có đường một chiều nối từ p_i tới p_{i+1} , ($\forall i: 1 \leq i \leq k$) được gọi là một *đường đi* từ p_1 tới p_k . Một đường đi gọi là *đơn giản* (hay *đường đi đơn*) nếu tất cả các địa điểm trên đường đi là hoàn toàn phân biệt.

Biết rằng tồn tại ít nhất một đường đi từ 1 tới n , hãy chỉ ra đường đi đơn có thứ tự từ điển nhỏ nhất.

Dữ liệu: Vào từ file văn bản DFS.INP

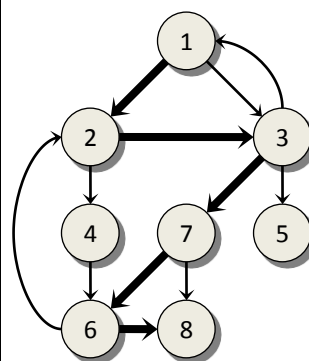
- Dòng 1 chứa số đỉnh $n \leq 10^3$, số cung $m \leq 10^5$
- m dòng tiếp theo, dòng thứ i chứa hai số nguyên dương u_i, v_i

Kết quả: Ghi ra trên một dòng của file văn bản DFS.OUT các địa điểm theo đúng thứ tự trên đường đi tìm được, bắt đầu từ địa điểm 1, kết thúc ở địa điểm n

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ

DFS.INP	DFS.OUT
8 12	1 2 3 7 6 8
1 2	
1 3	
2 3	
2 4	
3 1	
3 5	
3 7	
4 6	
6 2	
6 8	
7 8	
7 6	



102. ĐƯỜNG ĐI QUA ÍT ĐỊA ĐIỂM NHẤT

Hệ thống giao thông của một thành phố gồm n địa điểm đánh số từ 1 tới n và m con đường một chiều giữa các địa điểm đó, con đường thứ i cho phép đi từ địa điểm u_i tới địa điểm v_i .

Một dãy các địa điểm $P = \langle p_1, \dots, p_k \rangle$ sao cho có đường một chiều nối từ p_i tới p_{i+1} , ($\forall i: 1 \leq i \leq k$) được gọi là một *đường đi* từ p_1 tới p_k . Một đường đi gọi là *đơn giản* (hay *đường đi đơn*) nếu tất cả các địa điểm trên đường đi là hoàn toàn phân biệt.

Biết rằng tồn tại ít nhất một đường đi từ 1 tới n , hãy chỉ ra đường đi đơn qua ít địa điểm nhất

Dữ liệu: Vào từ file văn bản BFS.INP

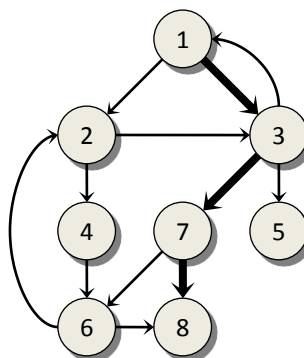
- Dòng 1 chứa số đỉnh $n \leq 10^3$, số cung $m \leq 10^5$
- m dòng tiếp theo, dòng thứ i chứa hai số nguyên dương u_i, v_i

Kết quả: Ghi ra trên một dòng của file văn bản BFS.OUT các địa điểm theo đúng thứ tự trên đường đi tìm được, bắt đầu từ địa điểm 1, kết thúc ở địa điểm n

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ

BFS.INP	BFS.OUT
8 12	1 3 7 8
1 2	
1 3	
2 3	
2 4	
3 1	
3 5	
3 7	
4 6	
6 2	
6 8	
7 8	
7 6	



103. CÁC THÀNH PHẦN LIÊN THÔNG

Cho đồ thị vô hướng $G = (V, E)$ gồm n đỉnh và m cạnh. Hãy liệt kê các thành phần liên thông của đồ thị

Dữ liệu: Vào từ file văn bản CONNECT.INP

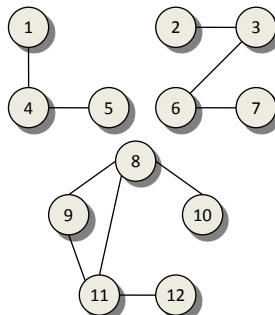
- Dòng 1 chứa hai số nguyên dương $n \leq 10^5$ và $m \leq 10^6$
- m dòng tiếp theo mỗi dòng chứa hai số nguyên dương u, v thể hiện có cạnh nối từ đỉnh u tới đỉnh v trong đồ thị.

Kết quả: Ghi ra file văn bản CONNECT.OUT gồm nhiều dòng, mỗi dòng liệt kê các đỉnh thuộc một thành phần liên thông

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ

CONNECT . INP	CONNECT . OUT
12 10	1 4 5
1 4	2 3 6 7
2 3	8 9 10 11 12
3 6	
4 5	
6 7	
8 9	
8 10	
9 11	
11 8	
11 12	



104. CÁC THÀNH PHẦN LIÊN THÔNG MẠNH

Cho đồ thị có hướng $G = (V, E)$ gồm n đỉnh và m cung. Hãy liệt kê các thành phần liên thông mạnh của đồ thị

Dữ liệu: Vào từ file văn bản SCONNECT.INP

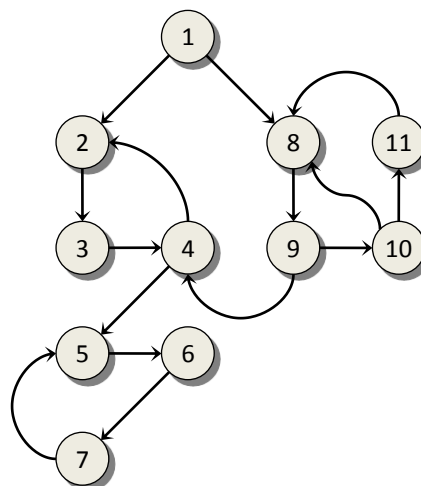
- Dòng 1 chứa hai số nguyên dương $n \leq 10^5$ và $m \leq 2 \cdot 10^5$
- m dòng tiếp theo mỗi dòng chứa hai số nguyên dương u, v thể hiện có cung nối từ đỉnh u tới đỉnh v trong đồ thị.

Kết quả: Ghi ra file văn bản SCONNECT.OUT gồm nhiều dòng, mỗi dòng liệt kê các đỉnh thuộc một thành phần liên thông mạnh

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

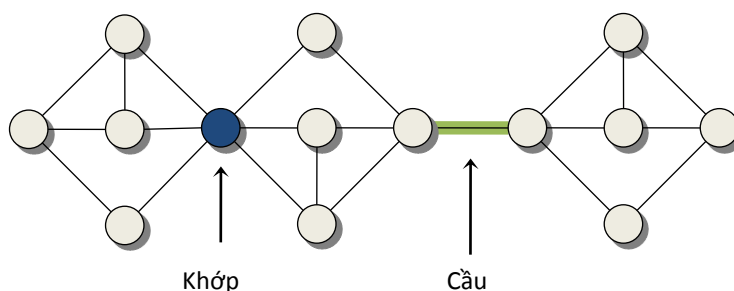
Ví dụ

SCONNECT . INP	SCONNECT . OUT
11 15	7 6 5
1 2	4 3 2
1 8	11 10 9 8
2 3	1
3 4	
4 2	
4 5	
5 6	
6 7	
7 5	
8 9	
9 4	
9 10	
10 8	
10 11	
11 8	



105. KHỚP VÀ CẦU

Cho đồ thị vô hướng $G = (V, E)$, các đỉnh được đánh số từ 1 tới n . Một đỉnh được gọi là khớp (articulation node) nếu ta xóa đỉnh đó và các cạnh liên thuộc với nó khỏi đồ thị thì sẽ được một đồ thị mới có nhiều thành phần liên thông hơn đồ thị ban đầu. Một cạnh được gọi là cầu (bridge) nếu ta xóa cạnh đó khỏi đồ thị thì được một đồ thị mới có nhiều thành phần liên thông hơn đồ thị ban đầu.



Hãy liệt kê tất cả các khớp và cầu của G

Dữ liệu: Vào từ file văn bản CUT.INP

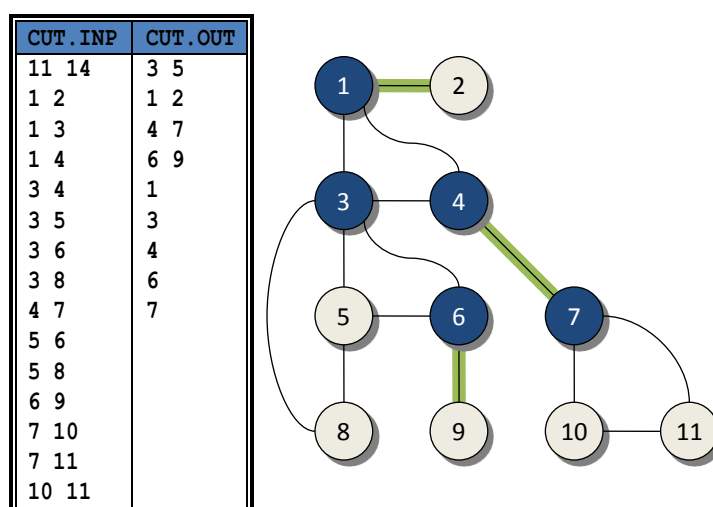
- Dòng 1: Chứa số đỉnh $n \leq 10^5$, số cạnh $m \leq 2 \cdot 10^5$ của đồ thị G
- m dòng tiếp theo, mỗi dòng ghi hai số nguyên dương u, v , thể hiện (u, v) là một cạnh của G .

Kết quả: Ghi ra file văn bản CUT.OUT

- Dòng 1: Ghi số cầu P và số khớp Q của đồ thị G
- P dòng tiếp theo, mỗi dòng ghi chỉ số hai đỉnh đầu mút của một cầu
- Q dòng tiếp theo, mỗi dòng ghi chỉ số một khớp

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ



106. CHU TRÌNH EULER

Cho $G = (V, E)$ là một đa đồ thị vô hướng liên thông, mọi đỉnh đều có bậc chẵn. Biết rằng G có n đỉnh và m cạnh.

Hãy tìm chu trình Euler của đồ thị G .

Dữ liệu: Vào từ file văn bản EULER.INP

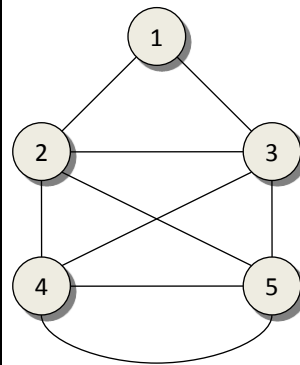
- Dòng 1: Chứa hai số nguyên dương $n \leq 10^5$ và $m \leq 4 \cdot 10^5$
- m dòng tiếp theo, dòng thứ i ghi chỉ số hai đầu mút của cạnh thứ i

Kết quả: Ghi ra file văn bản EULER.OUT các đỉnh trên chu trình Euler tìm được theo đúng thứ tự đi qua trên chu trình (bắt đầu và kết thúc ở cùng một đỉnh).

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách.

Ví dụ

EULER.INP	EULER.OUT
5 9	1 2 4 5 2 3 4 5 3 1
1 2	
1 3	
2 3	
2 4	
2 5	
3 4	
3 5	
4 5	
4 5	



107. CÂY

Cho đồ thị vô hướng dạng cây $G = (V, E)$ gồm n đỉnh đánh số từ 1 tới n . Định nghĩa khoảng cách giữa hai đỉnh s, t là số cạnh trên đường đi đơn duy nhất từ s tới t . Độ rộng của cây được định nghĩa bằng tổng khoảng cách giữa mọi cặp đỉnh s, t ($s < t$)

Yêu cầu: Xác định độ rộng của cây.

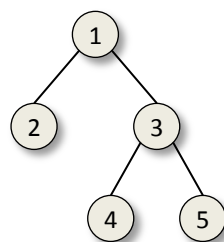
Dữ liệu: Vào từ file văn bản TREE.INP

- Dòng 1 chứa số nguyên dương $n \leq 10^5$
- $n - 1$ dòng tiếp theo, mỗi dòng chứa chỉ số hai đầu mút của một cạnh cách nhau ít nhất một dấu cách

Kết quả: Ghi ra file văn bản TREE.OUT một số nguyên duy nhất là độ rộng của cây đã cho

Ví dụ

TREE.INP	TREE.OUT
5 1 2 1 3 3 4 3 5	18



108. ĐƯỜNG ĐI NGẮN NHẤT

Cho đồ thị có hướng có trọng số $G = (V, E, w)$ gồm n đỉnh đánh số từ 1 tới n và m cung đánh số từ 1 tới m . Cung thứ i nối từ u_i tới v_i và có trọng số w_i . Hãy xác định độ dài đường đi ngắn nhất từ 1 tới n .

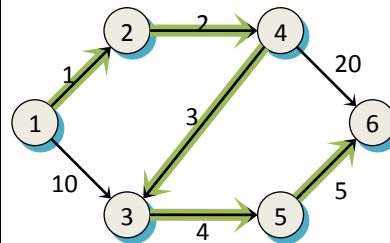
Dữ liệu: Vào từ file văn bản DIJKSTRA.INP

- Dòng 1 chứa hai số nguyên dương $n \leq 10^5$; $m \leq 2 \cdot 10^5$
- m dòng tiếp theo, dòng thứ i chứa ba số nguyên dương u_i, v_i, w_i ($w_i \leq 10^9$)

Kết quả: Ghi ra file văn bản DIJKSTRA.OUT một số nguyên là độ dài đường đi ngắn nhất từ 1 tới n . Nếu không tồn tại đường đi từ 1 tới n , ghi ra số -1.

Ví dụ

DIJKSTRA.INP	DIJKSTRA.OUT
6 7 1 2 1 1 3 10 2 4 2 3 5 4 4 3 3 4 6 20 5 6 5	15



109. ĐƯỜNG ĐI

Cho đồ thị có hướng $G = (V, E)$ gồm n đỉnh và m cung, s và t là hai đỉnh của G . Một dãy các đỉnh $P = \langle s = p_0, p_1, \dots, p_k = t \rangle$ sao cho $(p_{i-1}, p_i) \in E, \forall i: 1 \leq i \leq k$ được gọi là một đường đi từ s tới t .

Yêu cầu: Tìm một đường đi từ s tới t , biết rằng tồn tại đường đi từ s tới t

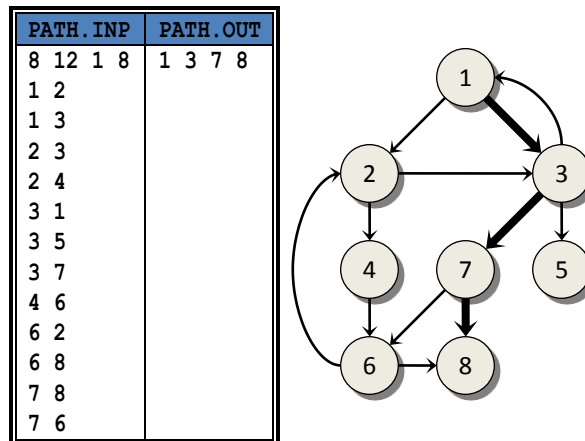
Dữ liệu: Vào từ file văn bản PATH.INP

- Dòng 1 chứa số đỉnh $n \leq 10^5$, số cung $m \leq 10^6$, đỉnh xuất phát s , đỉnh cần đến t .
- m dòng tiếp theo, mỗi dòng chứa hai số nguyên dương u, v thể hiện có cung nối từ đỉnh u tới đỉnh v trong đồ thị.

Kết quả: Ghi ra trên một dòng của file văn bản PATH.OUT các đỉnh theo đúng thứ tự trên đường đi tìm được, bắt đầu từ đỉnh s , kết thúc ở đỉnh t

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ



110. CÂY KHUNG NHỎ NHẤT

Cho một đồ thị vô hướng G gồm n đỉnh đánh số từ 1 tới n và m cạnh đánh số từ 1 tới m , cạnh thứ i nối hai đỉnh u_i, v_i và có trọng số là w_i . Giữa hai đỉnh có thể có nhiều cạnh nối.

Yêu cầu: Cho biết trọng số của cây khung nhỏ nhất của đồ thị G .

Dữ liệu: Vào từ file văn bản MST.INP

- Dòng 1: Chứa hai số nguyên dương $n \leq 10^5$; $m \leq 10^5$
- m dòng tiếp, dòng thứ i chứa ba số nguyên u_i, v_i, w_i ($|w_i| \leq 10^9, \forall i$)

Kết quả: Ghi ra file văn bản MST.OUT một số nguyên duy nhất là trọng số cây khung nhỏ nhất của đồ thị, ghi ra thông báo DISCONNECTED nếu đồ thị không tồn tại cây khung

Ví dụ

MST . INP	MST . OUT
4 4 1 2 2 1 3 3 2 4 4 2 3 1	7