

# Cấu trúc dữ liệu và giải thuật



Lecture 7: Đồ thị có hướng

Ta Viet Cuong, Ph.D  
HMI laboratory, FIT-UET

# Today

I. Định nghĩa

II. Bài toán

III. Sắp xếp topo

IV. Thành phần liên thông mạnh

V. Bài tập

# Định nghĩa

Đồ thị có hướng là một cặp 2 tập hợp  $\langle V, E \rangle$

- $|V|=N$  là số đỉnh
- $|E|=M$  là số cạnh

Các khái niệm cơ bản:

- Đường đi
- Chu trình
- Liên thông mạnh
- Bậc: bậc vào, bậc ra

# Ví dụ

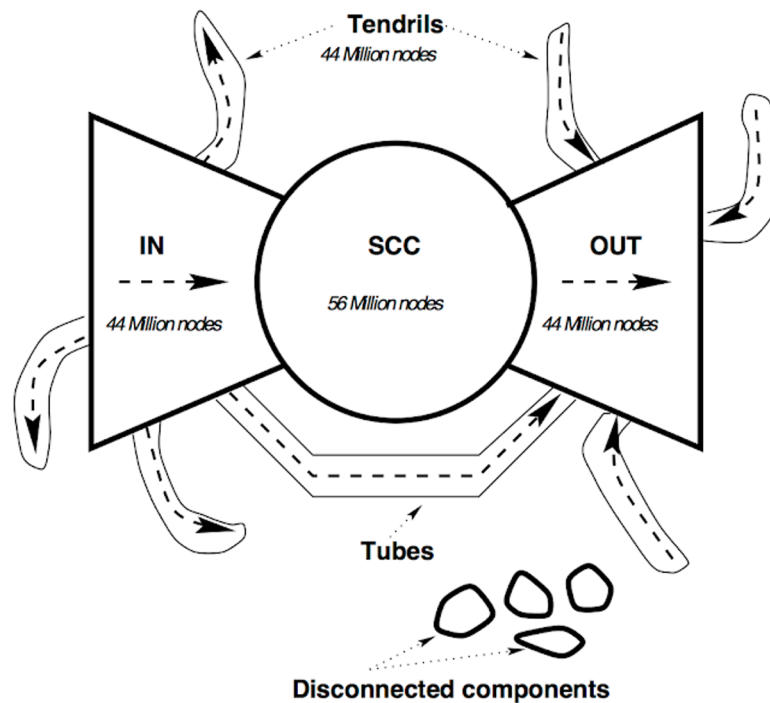
Các đồ thị dưới đây là đồ thị có hướng hay vô hướng:

- Đồ thị đường giao thông
- Đồ thị về bạn bè trên mạng xã hội
- Đồ thị về tương tác trên mạng xã hội



# Ví dụ

Mạng WWW trong  
thuật toán PageRank của google:



*Original "bowtie" schematic from "Graph Structure in the Web"*

\*Broder, Andrei, et al. "Graph structure in the web." *Computer networks* 33.1-6 (2000): 309-320.

# Bài toán

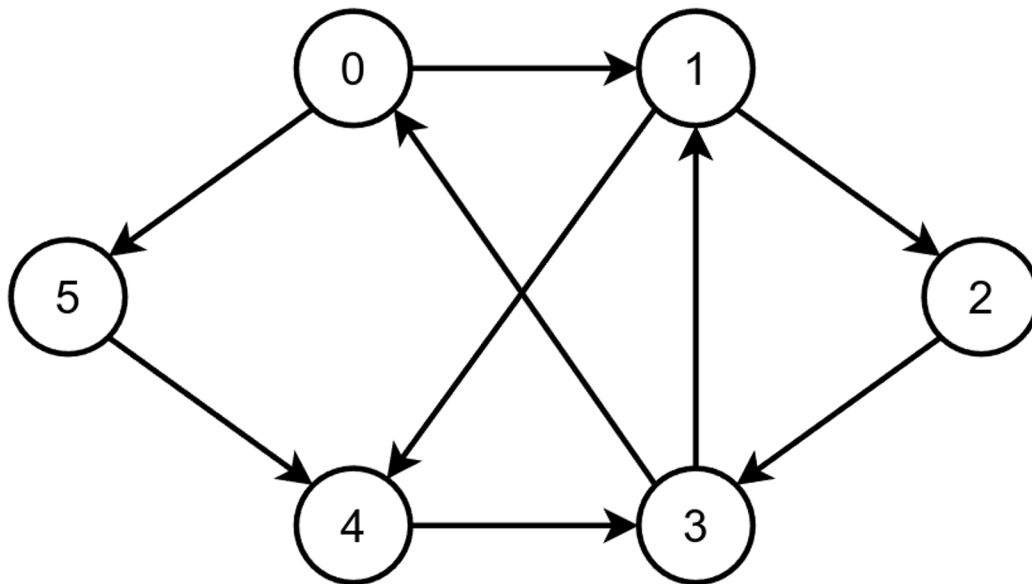
Bài toán	Mô tả
Đường đi s-t	Có đường đi giữa hai đỉnh $s$ và $t$ hay không
Đường đi ngắn nhất s-t	Tìm đường đi ngắn nhất từ $s$ tới $t$
Chu trình có hướng	Tìm một chu trình trong đồ thị
Sắp xếp topo	Sắp xếp các đỉnh theo một thứ tự để tất cả các mũi tên hướng theo cùng một hướng
Tính liên thông mạnh	Mọi đỉnh của đồ thị liên thông mạnh với nhau không ?
PageRank	Trong các đỉnh

# Duyệt trên đồ thị

In ra các đỉnh có thể đến được từ v theo 1 thứ tự nào đó

- Preorder: Thứ tự các nút theo lời gọi DFS(s) bắt đầu
- Postorder: Thứ tự các nút theo lời gọi DFS(s) trả về

## Duyệt trên đồ thị

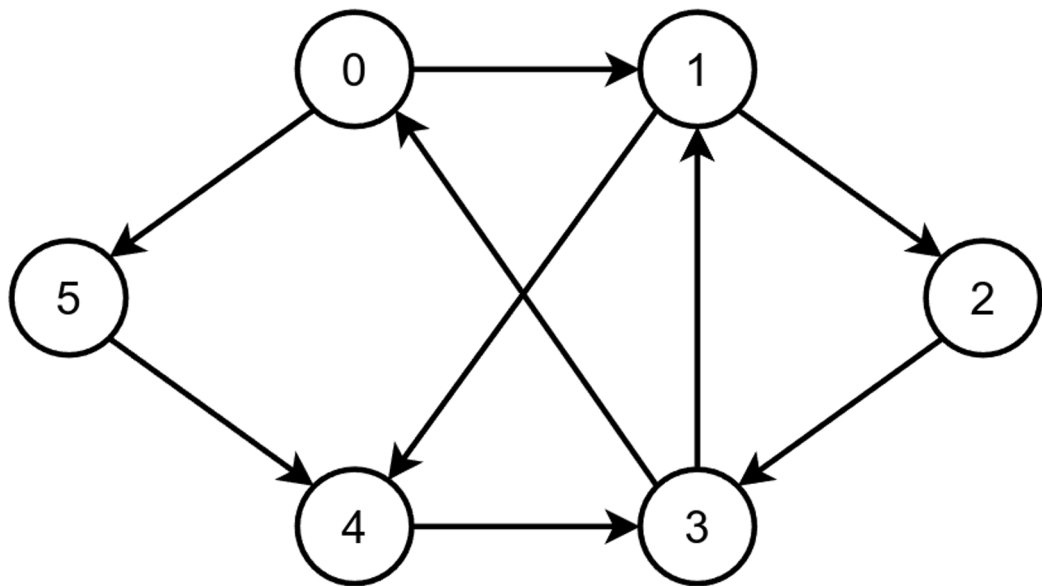




# Duyệt trên đồ thị

—> Tiền thứ tự là 0 1 2 3 4 5

—> Hậu thứ tự là 3 2 4 1 5 0



# Sắp xếp topo trên đồ thị có hướng không chu trình

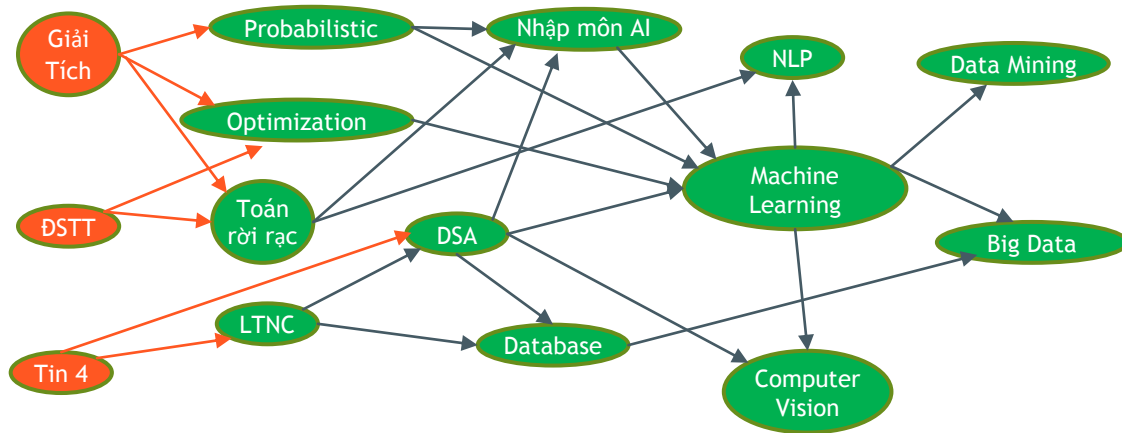
Đồ thị có hướng không có chu trình (Directed Acyclic Graph - DAG):

- Là đồ thị có hướng không tồn tại chu trình.

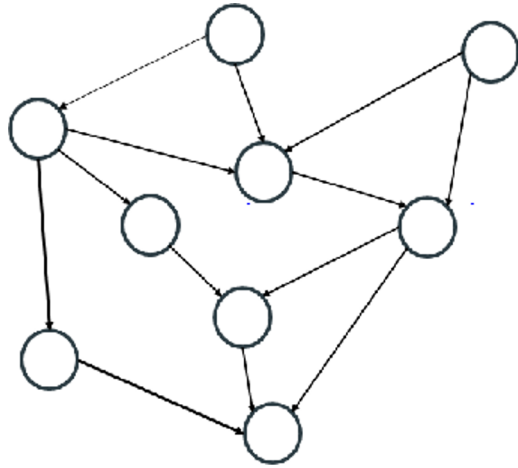
Sắp xếp topo: Tìm một thứ tự các đỉnh sao cho với mọi cặp đỉnh mà có hướng  $(u,v)$  có nghĩa  $u$  phải được thực hiện trước  $v$ .

Sử dụng thuật toán DFS

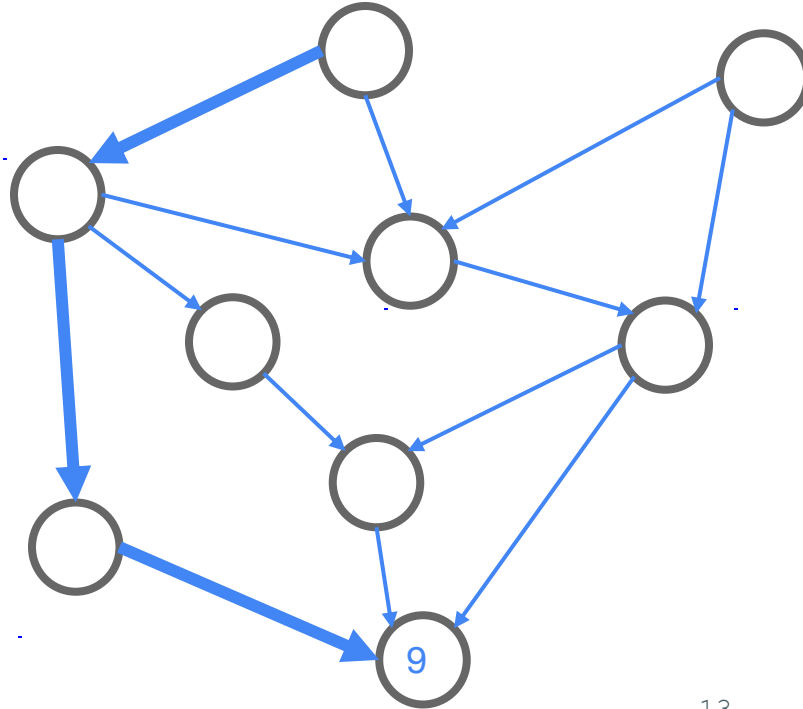
# Topological Sorting



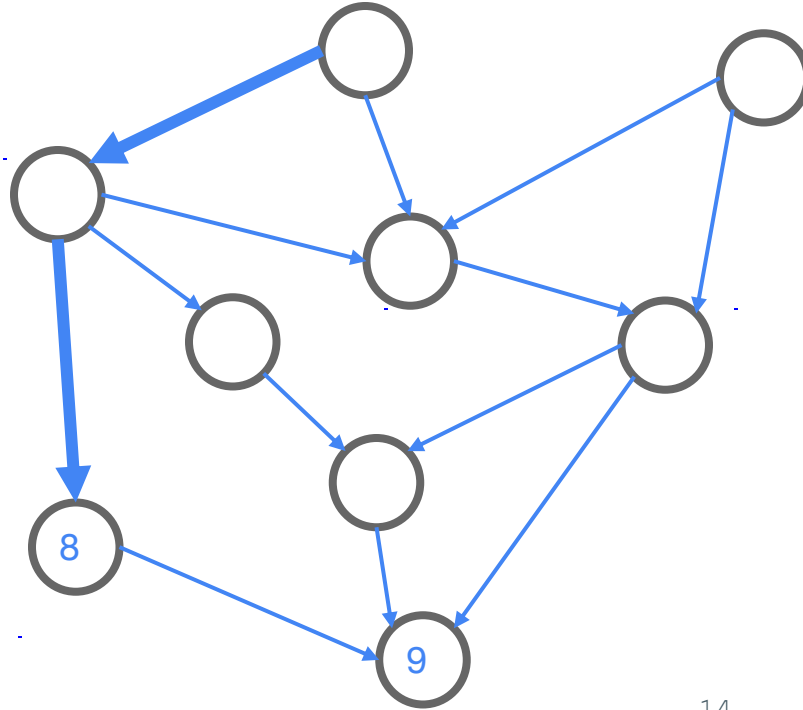
# Topological Sorting - DFS



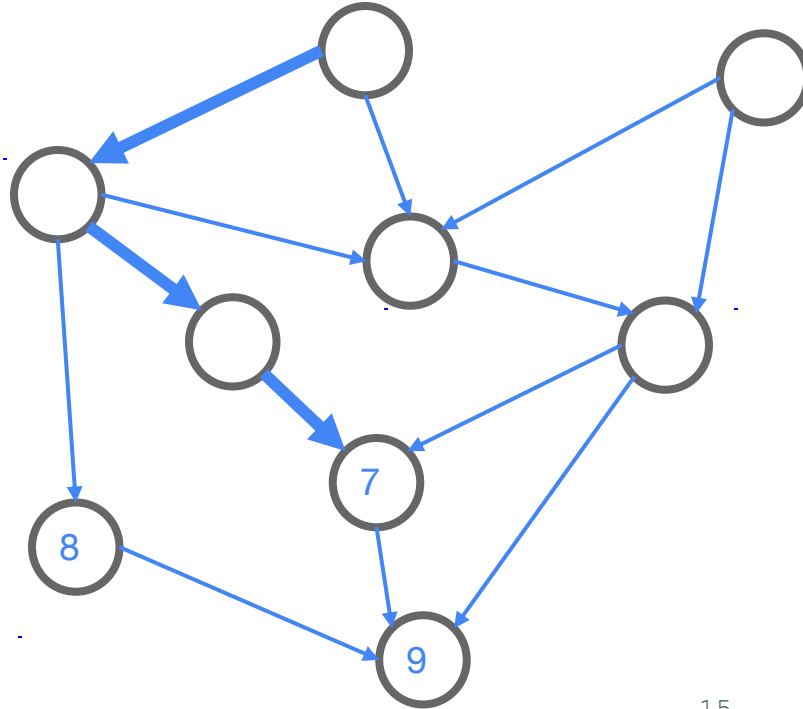
# Topological sort by DFS



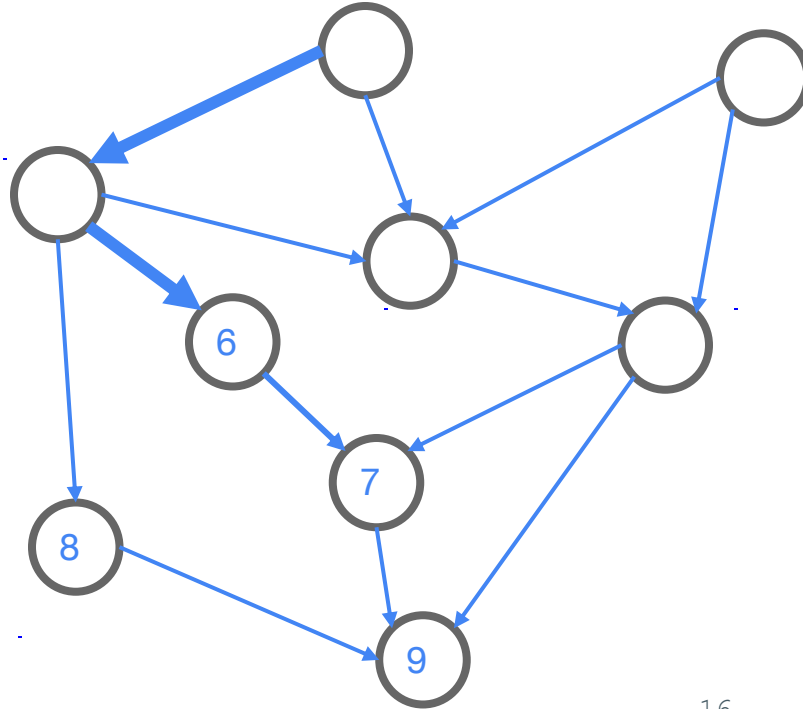
# Topological sort by DFS



# Topological Sorting - DFS

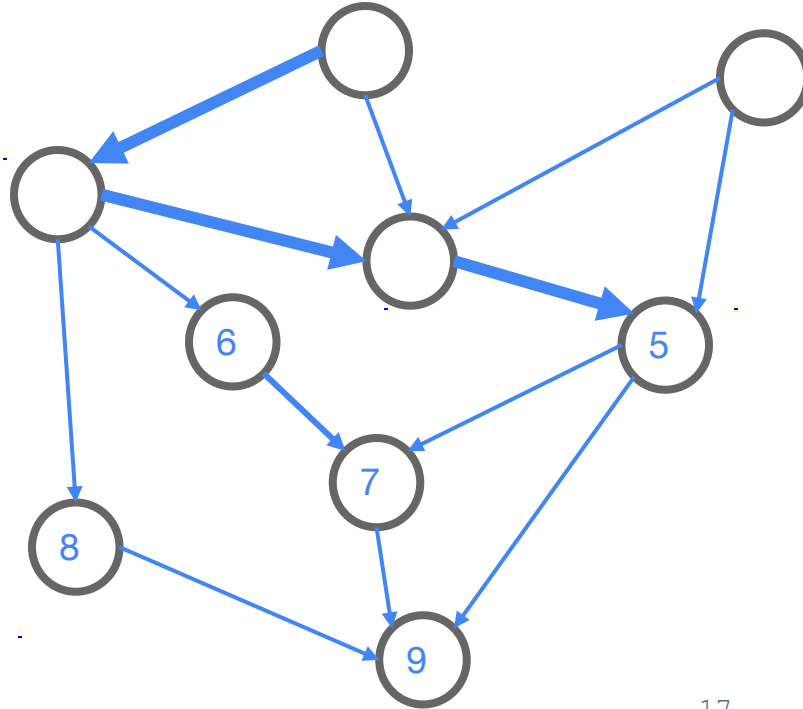


# Topological Sorting - DFS

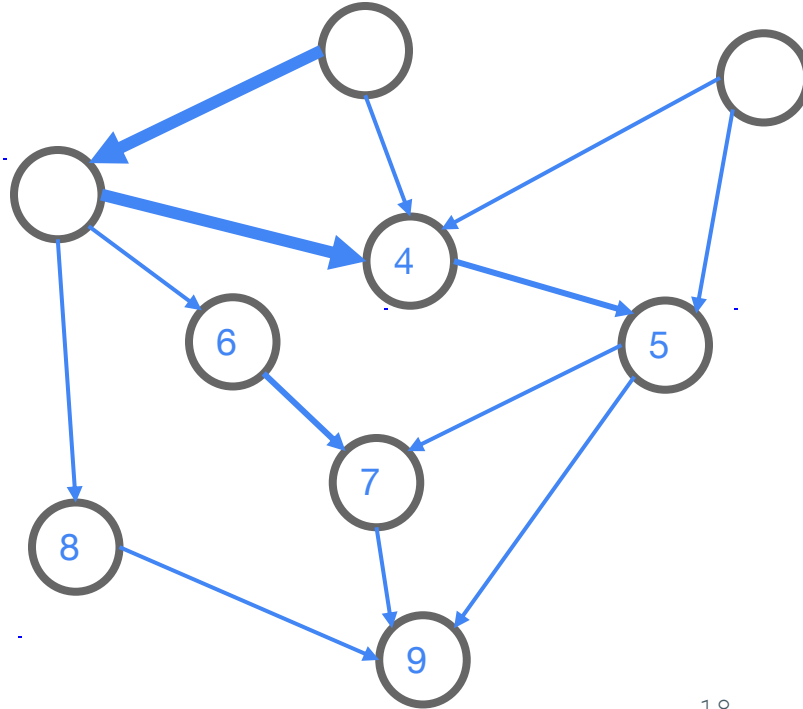




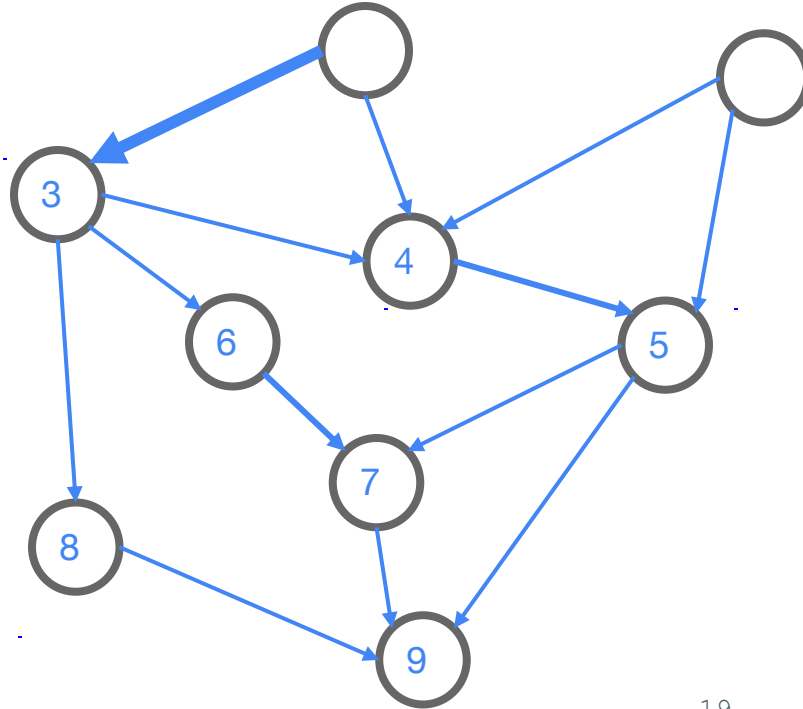
# Topological Sorting - DFS



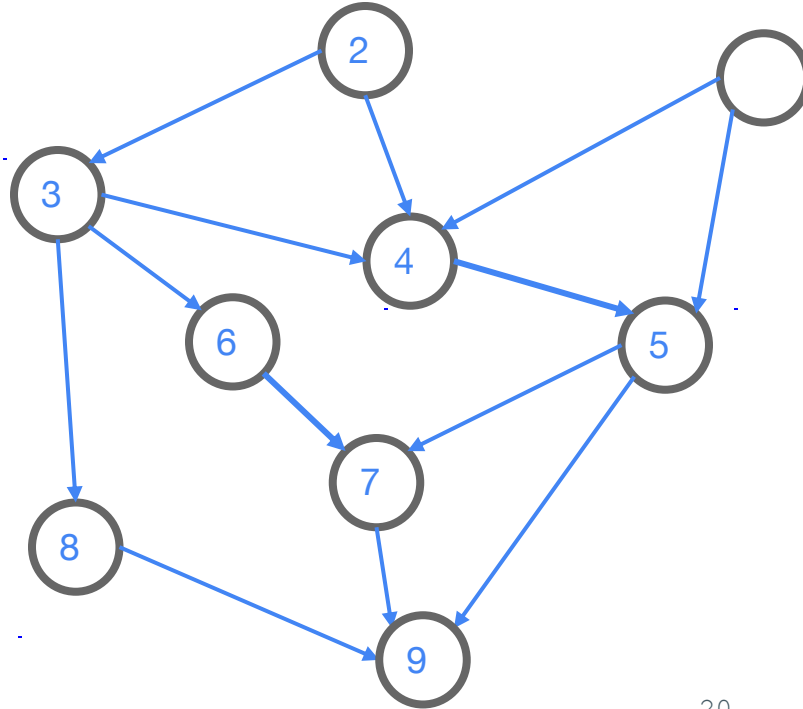
# Topological Sorting - DFS



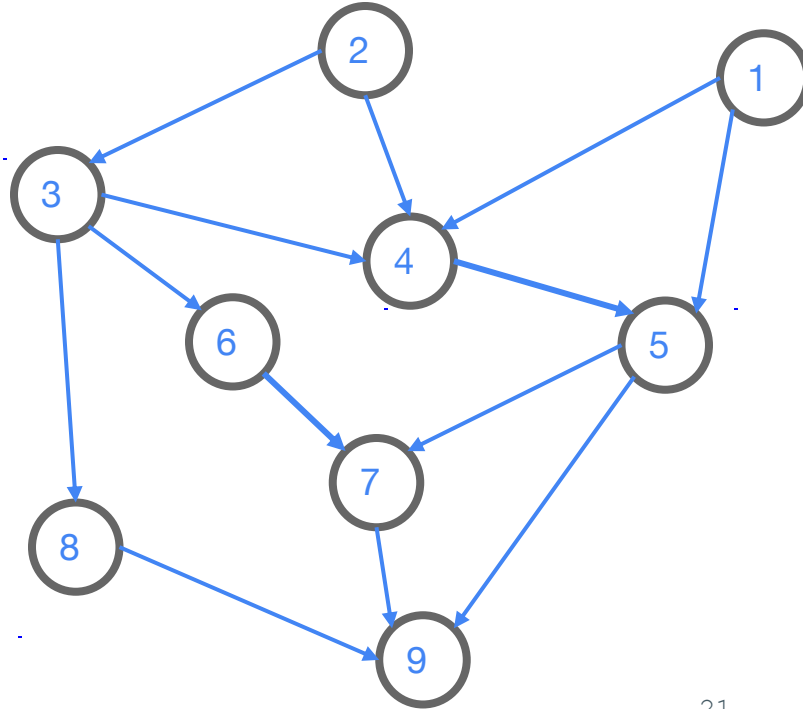
# Topological Sorting - DFS



# Topological Sorting - DFS



# Topological Sorting - DFS



# Thành phần liên thông mạnh

Hai đỉnh  $u, v$  **liên thông mạnh** nếu tồn tại đường đi qua lại giữa chúng Quan hệ liên thông mạnh là quan hệ tương đương (phản xạ, đối xứng, bắc cầu) nên sẽ chia đồ thị có hướng thành các thành phần liên thông mạnh:

- Các phần tử cùng thành phần liên thông mạnh thì liên thông mạnh với nhau
- Các phần tử khác thành phần liên thông mạnh thì không liên thông mạnh với nhau (không có đường đi theo cả 2 chiều hoặc 1 trong 2 chiều)

# Bài tập

Tính độ phức tạp tính toán cho hàm sum() dưới đây:

```
int sum(int m, int K ) {  
    int s = 0;  
    for (int i = 0; i < m; i += i) {  
        s += i;  
    }  
    return s;  
}
```

# Bài tập

Cho một dãy  $A$  gồm  $n$  nguyên dương được lưu giữ dưới dạng một danh sách liên kết đơn (single linked-list). Mỗi nút trong danh sách có cấu trúc như sau:

Hãy viết hàm `int count(Node* head, int key)` với C/C++, hay

```
// C/C++  
  
Node {  
  
    int data;  
  
    Node* next;  
  
}
```



# Bài tập

1. Hãy dùng hàng đợi ưu tiên để giải bài toán sau: Cho danh sách  $M$  cuộc điện thoại tới một mạng điện thoại (danh sách rất dài, không thể lưu lại trong bộ nhớ máy tính, dữ liệu về mỗi cuộc điện thoại bao gồm số điện thoại được gọi đến và độ dài cuộc gọi), hãy liệt kê  $N$  cuộc gọi dài nhất. Lưu ý: không cần cài đặt hàng đợi ưu tiên mà chỉ cần sử dụng đến hàng đợi ưu tiên trong thuật toán của bạn.
2. Để thuật toán trên chạy với hiệu quả tốt nhất, bạn sử dụng cách cài đặt hàng đợi ưu tiên bằng cấu trúc dữ liệu gì? Mảng, danh sách liên kết, cây nhị phân tìm kiếm, heap, bảng băm? Khi đó thuật toán bạn thiết kế có độ phức tạp như thế nào?

# Bài tập

Cho đồ thị vô hướng 8 đỉnh với các đỉnh đánh số từ 0 đến 7 và các cạnh sau:

0 1, 0 2, 0 3, 0 5, 1 4, 1 6, 2 3, 3 4, 5 6, 5 7, 6 7.

1. Hãy trình bày mô hình đồ thị theo phương pháp danh sách kề
2. Áp dụng thuật toán duyệt đồ thị theo chiều rộng bắt đầu từ đỉnh 0, hãy liệt kê các đỉnh của đồ thị theo thứ tự duyệt.

Lưu ý: thứ tự ưu tiên duyệt trong danh sách kề của mỗi đỉnh là thứ tự tăng dần của chỉ số đỉnh. Ví dụ, thứ tự duyệt từ đỉnh 0 là 1, 2, 3, 5. Từ đỉnh 4 là 1, 3.