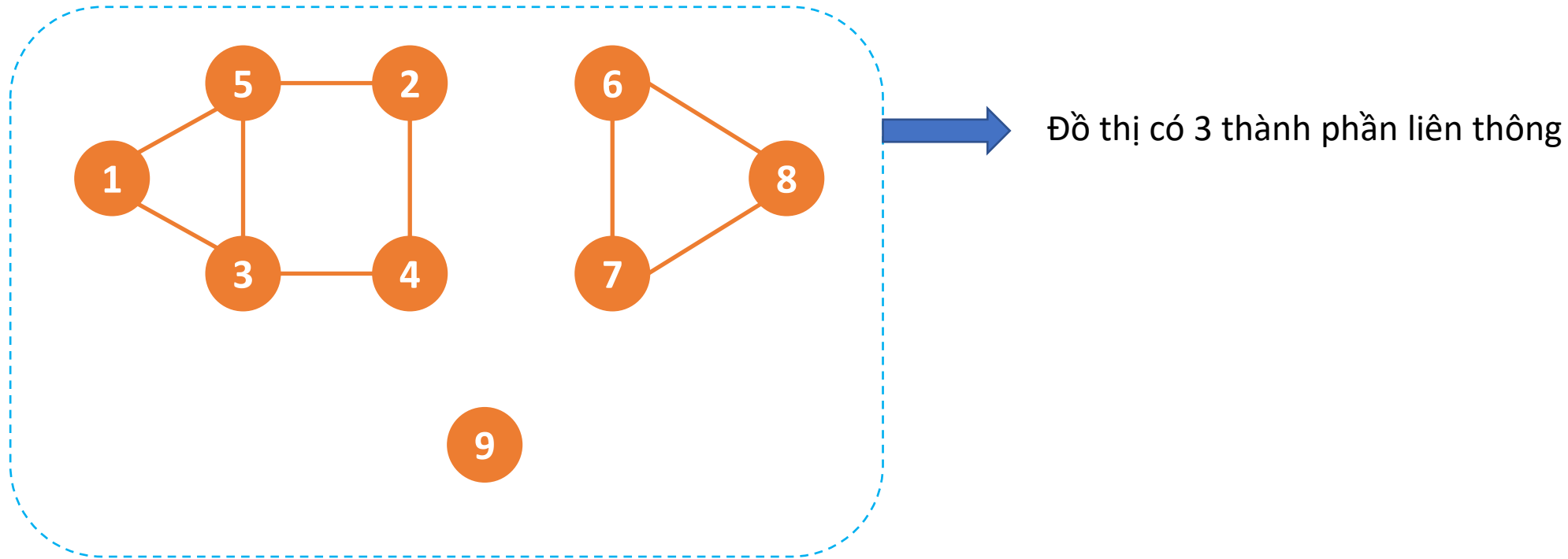


Ứng dụng DFS: tìm thành phần liên thông

- Cho đồ thị vô hướng $G = (V, E)$ trong đó $V = \{1, 2, \dots, n\}$ là tập đỉnh. Một đồ thị con lớn nhất liên thông của G được gọi là một thành phần liên thông



Ứng dụng DFS: tìm thành phần liên thông

- DFS(u): thủ tục duyệt theo chiều sâu trên G bắt đầu từ đỉnh u cho phép thăm tất cả các đỉnh cùng thành phần liên thông với u
- Cấu trúc dữ liệu
 - $A[v]$: danh sách các đỉnh kề với v
 - $C[v]$: chỉ số của thành phần liên thông chứa đỉnh v
 - nbCC: số thành phần liên thông

```
DFS( $u, A$ ) {  
     $C[u] = \text{nbCC}$ ;  
     $\text{visited}[u] = \text{true}$ ;  
    for  $v \in A(u)$  do {  
        if( $\text{visited}[v] = \text{false}$ ) {  
            DFS( $v, A$ );  
        }  
    }  
}
```

```
DFS( $G=(V,E), A$ ) {  
    for  $v \in V$  do {  
         $\text{visited}[v] = \text{false}$ ;  
    }  
    nbCC = 0;  
    for  $u \in V$  do {  
        if( $\text{visited}[u] = \text{false}$ ) {  
            nbCC = nbCC + 1;  
            DFS( $u, A$ );  
        }  
    }  
}
```

Ứng dụng DFS: tìm thành phần liên thông mạnh

- Cho $G = (V, E)$ là đồ thị có hướng. Một đồ thị con lớn nhất liên thông mạnh (giữa 2 cặp đỉnh u và v bất kỳ, luôn có đường đi từ u đến v và từ v đến u) của G được gọi là một thành phần liên thông mạnh của G .
- Thuật toán tìm thành phần liên thông mạnh
- Bước 1: Thực hiện DFS trên G và ghi nhận $f[v]$ là thời điểm đỉnh v kết thúc thăm (thời điểm mà DFS(v) thực hiện xong).
- Bước 2: Xây dựng đồ thị bù $G' = (V', E')$ của G
 - $V' = V$
 - (u, v) thuộc E' khi và chỉ khi (v, u) thuộc E
- Bước 3: Thực hiện DFS trên G' với thứ tự các đỉnh được duyệt là thứ tự giảm dần của f .
 - Tập các đỉnh được thăm bởi lời gọi DFS(u) sẽ là tập các đỉnh của 1 thành phần liên thông mạnh của G