

## 〈알고리즘 실습〉 - 그래프 순회

### ※ 입출력에 대한 안내

- 특별한 언급이 없으면 문제의 조건에 맞지 않는 입력은 입력되지 않는다고 가정하라.
- 특별한 언급이 없으면, 각 줄의 맨 앞과 맨 뒤에는 공백을 출력하지 않는다.
- 출력 예시에서 □는 각 줄의 맨 앞과 맨 뒤에 출력되는 공백을 의미한다.
- 입출력 예시에서  $\mapsto$  이 후는 각 입력과 출력에 대한 설명이다.

[ 문제 1 ] (DFS) 입력으로 주어지는 그래프의 DFS 순회 결과를 출력하는 프로그램을 작성하시오.

### 입력 그래프의 성질:

- $N$  ( $1 \leq N \leq 100$ ) 개의 정점과  $M$  ( $1 \leq M \leq 1,000$ ) 개의 간선으로 구성
- 정점은  $1 \sim N$  사이의 정수로 번호가 매겨져 있고, 정점의 번호는 모두 다름
- 모든 간선은 무방향 간선이고, 한 정점에서 임의의 다른 정점으로 가는 경로는 반드시 존재

### 구현 조건:

- 그래프는 **인접리스트 구조**를 사용하여 표현해야 한다.
- 인접 정점의 조사 순서
  - 기준 정점  $u$ 의 인접 정점(or 부착 간선)들을 번호가 작은 정점부터 조사한다.  
(즉, 아래 DFS 의사 코드의 for문(☛)에서 인접 정점들을 번호가 작은 정점부터 큰 순서대로 조사한다. 조사 순서에 따라 방문 결과는 다를 수 있다.)

```
DFS(u)
{
    u방문;
    for u의 인접 정점들 x에 대해서    ☛
        if(x를 아직 방문하지 않았으면)
            DFS(x);
}
```

### - 힌트

- 1) 그래프 구축 시 간선이 추가될 때마다 “인접리스트”의 정렬된 위치에 삽입 (삽입 정렬과 유사)
- 2) for문(☛)에서  $u$ 의 인접 정점들을 인접 리스트에 저장된 순서대로 조사

### 입출력:

- 입력
  - 첫 줄에 정점의 개수  $N$ , 간선의 개수  $M$ , 순회 시작 정점 번호  $S$ 가 주어진다.
  - 이후  $M$ 개의 줄에 한 줄에 하나씩 간선의 정보(간선의 양 끝 정점 번호)가 주어진다.  
간선은 임의의 순서로 입력되고, 중복 입력되는 간선은 없다.  
(무방향 간선이므로 간선  $(u,v)$ 와  $(v,u)$ 는 동일한 간선으로 취급)

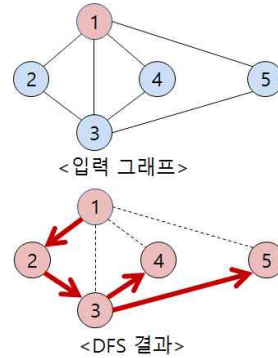
○ 출력

- 시작 정점 S에서 출발하는 DFS의 방문 순서대로 정점 번호를 출력한다.

입력 예시 1

5 7 1	↦ N=5, M=7, S=1	1
1 2		2
1 4		3
5 1		4
3 5		5
4 3		
3 1		
2 3		

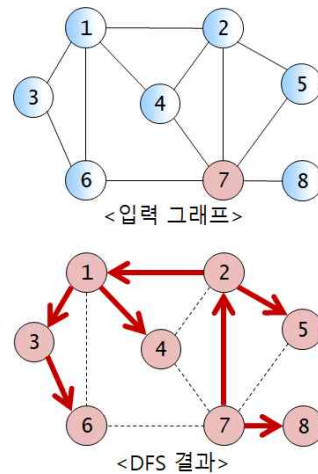
출력 예시 1



입력 예시 2

8 12 7	↦ N=8, M=12, S=7	7
1 2		2
2 4		1
4 7		3
3 6		6
6 1		4
7 6		5
7 8		8
1 3		
2 7		
1 4		
2 5		
7 5		

출력 예시 2



[ 문제 2 ] (BFS) 입력으로 주어지는 그래프의 BFS 순회 결과를 출력하는 프로그램을 작성하시오.

입력 그래프의 성질:

- 문제 1과 동일

구현 조건:

- 그래프는 **인접행렬 구조**를 사용하여 표현해야 한다.
- 인접 정점의 조사 순서
  - 문제 1과 동일하게 기준 정점의 인접 정점(or 부착 간선)들을 번호가 작은 정점부터 조사한다.

입출력:

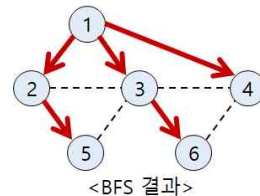
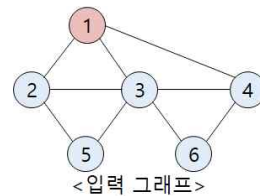
- 입력 : 문제 1과 동일
- 출력 : 시작 정점 S에서 출발하는 BFS의 방문 순서대로 정점 번호를 출력한다.

입력 예시 1

```
6 9 1  → N=6, M=9, S=1
3 5
1 3
4 1
2 3
3 4
6 4
3 6
1 2
2 5
```

출력 예시 1

```
1
2
3
4
5
6
```



입력 예시 2

```
8 12 4  → N=8, M=12, S=4
1 2
2 4
4 7
3 6
6 1
7 6
7 8
1 3
2 7
1 4
2 5
7 5
```

출력 예시 2

```
4
1
2
7
3
6
5
8
```

