

자료구조론 실습

DFS / BFS

2017. 06. 02

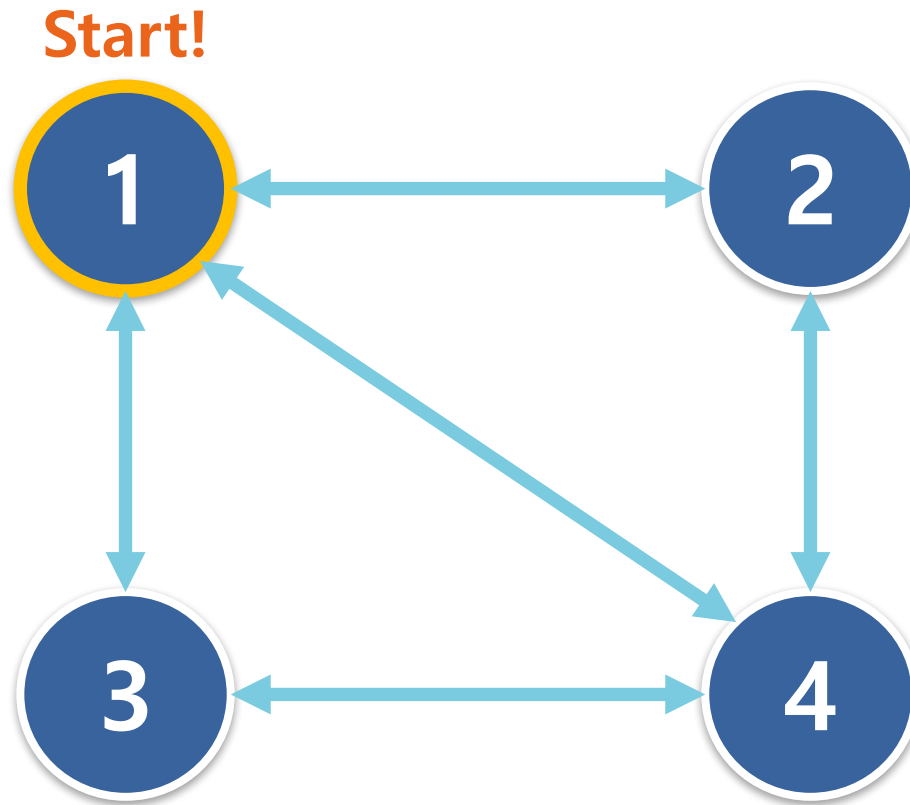
한양대학교

이주홍

Assignment 8 - Graph Search

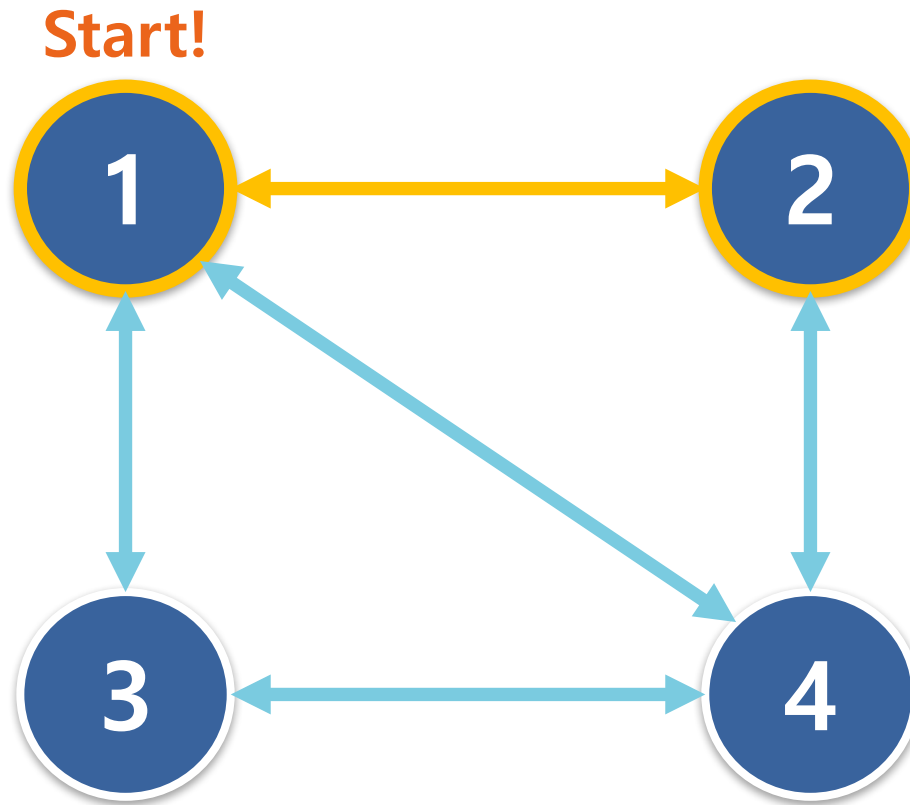
- 주어진 그래프를 DFS와 BFS로 탐색한 결과를 출력한다.
- 방문할 수 있는 정점이 여러 개일 경우, 정점 번호가 작은 것을 먼저 방문한다.

Assignment 8 - BFS



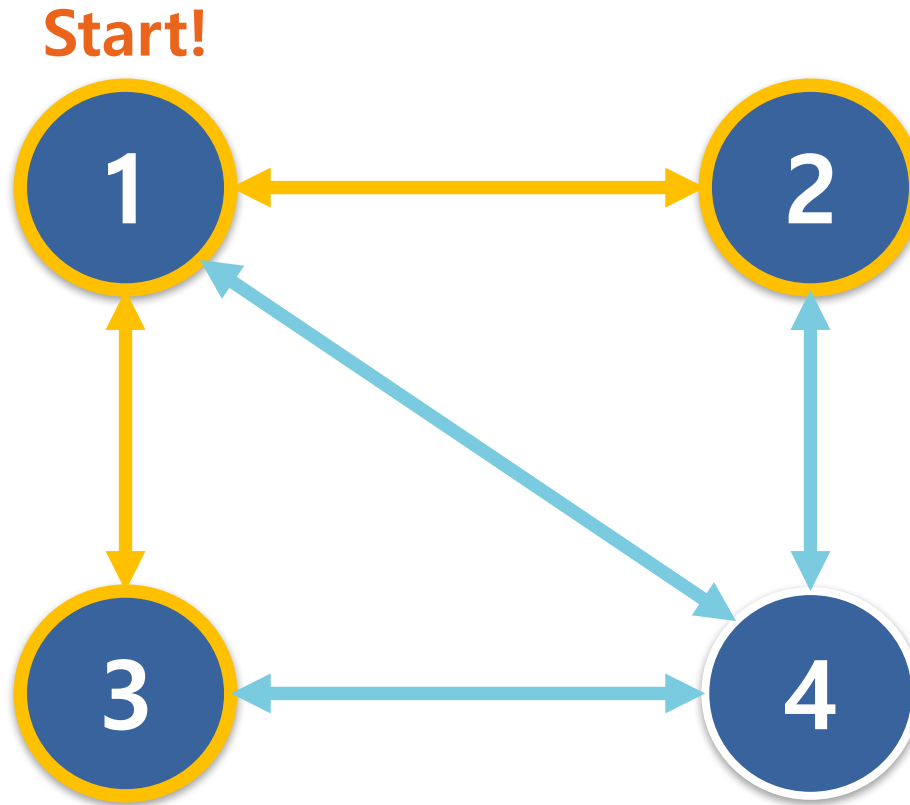
출력 >> 1

Assignment 8 - BFS



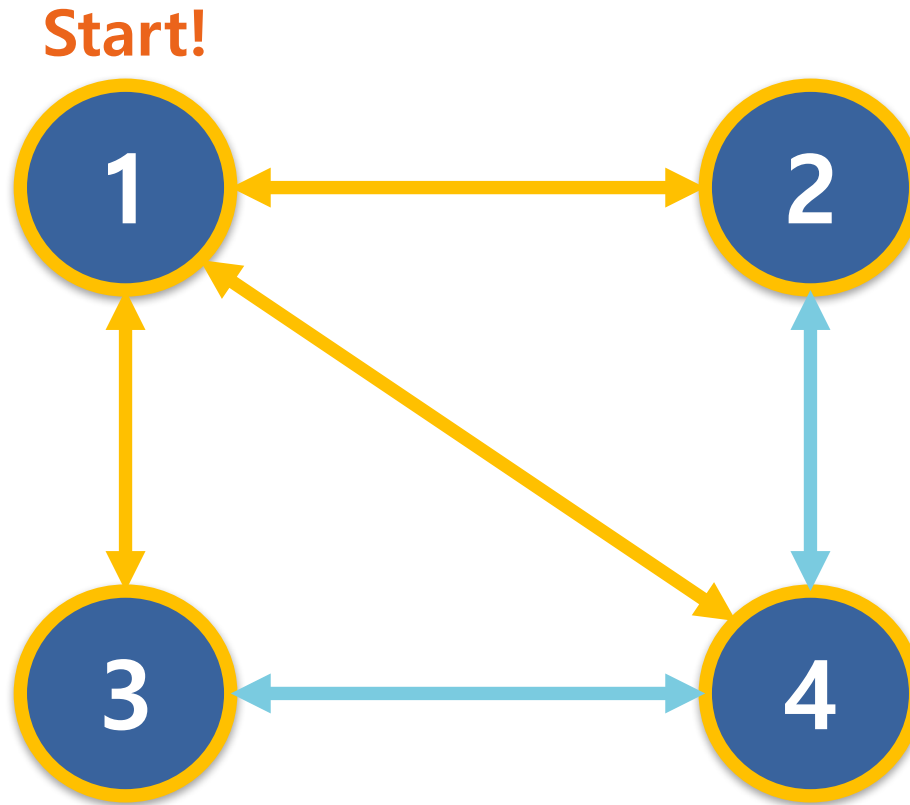
출력 >> 1 2

Assignment 8 - BFS



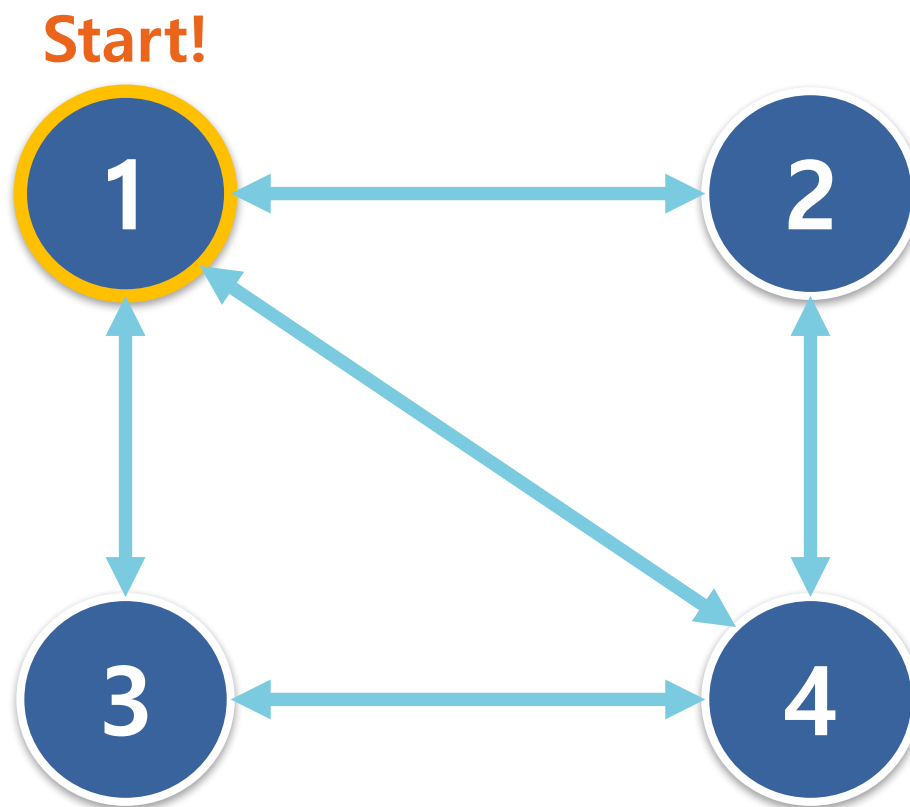
출력 >> 1 2 3

Assignment 8 - BFS



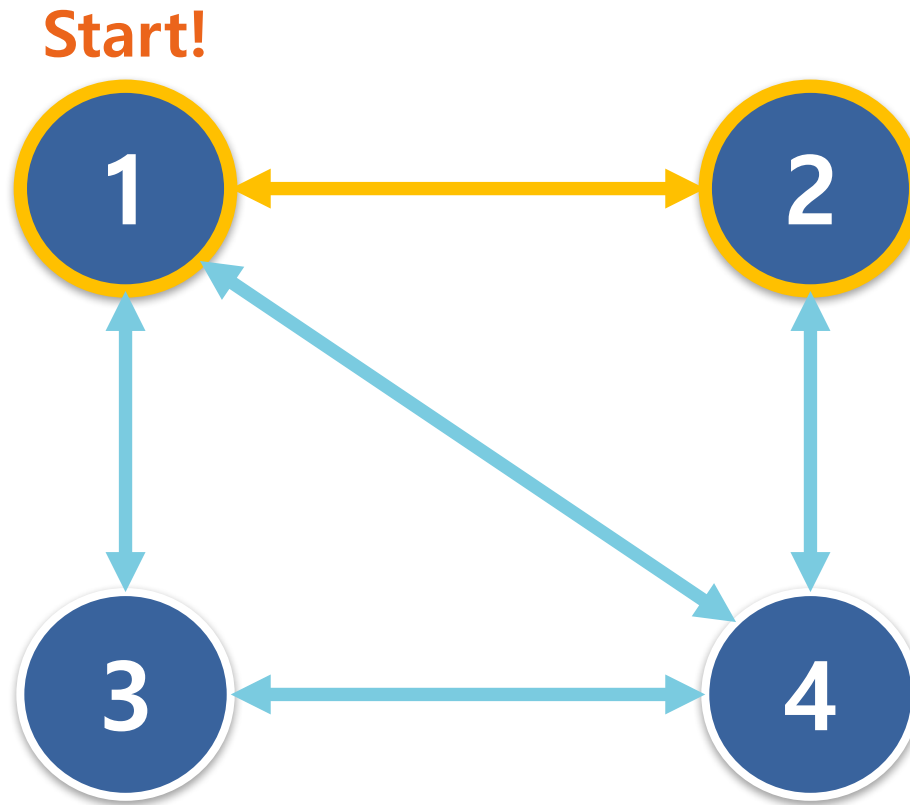
출력 >> 1 2 3 4

Assignment 8 - DFS



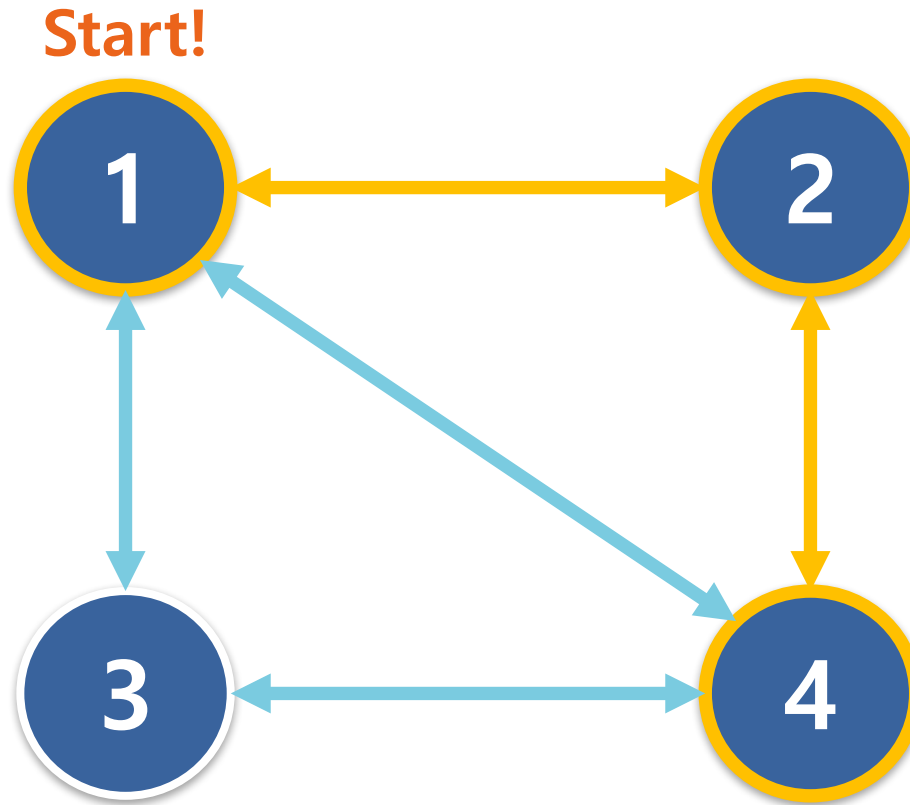
출력 >> 1

Assignment 8 - DFS



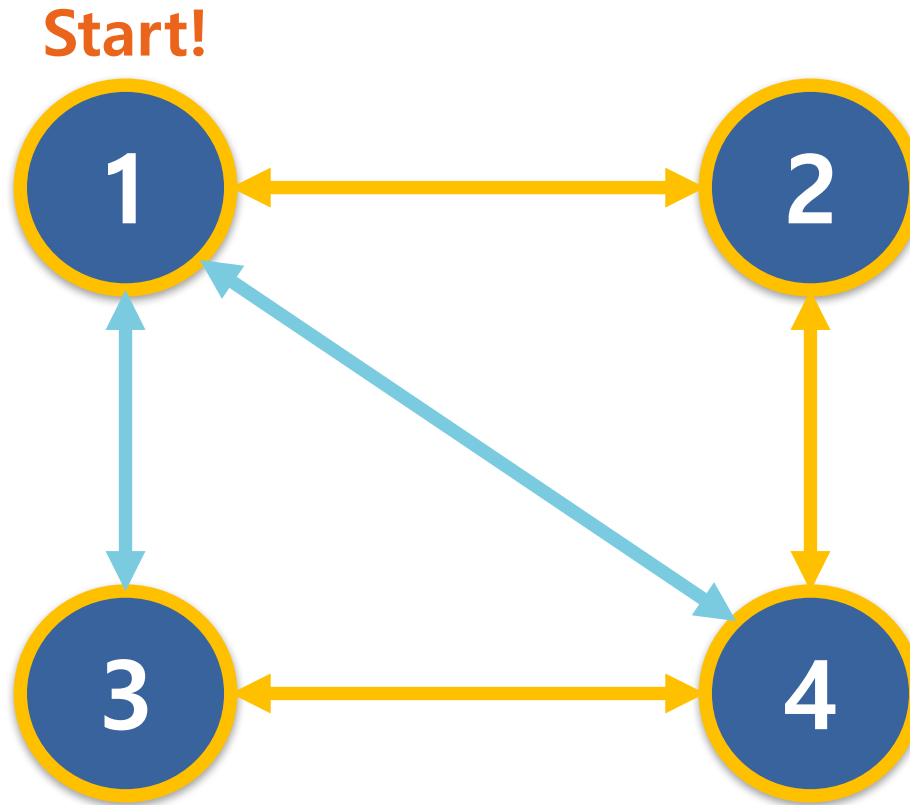
출력 >> 1 2

Assignment 8 - DFS



출력 >> 1 2 4

Assignment 8 - DFS



출력 >> 1 2 4 3

Assignment 8 - Graph Search

- 주어진 그래프를 DFS와 BFS로 탐색한 결과를 출력한다.
- 방문할 수 있는 정점이 여러 개일 경우, 정점 번호가 작은 것을 먼저 방문한다.
- 입출력 설명
 - 정점의 수 $n(\leq 1,000)$, 간선의 수 $m(\leq 10,000)$, 시작 정점의 번호 v 가 첫째 줄에 입력된다.
 - 둘째 줄부터 $m+1$ 째 줄에 걸쳐 간선이 연결하는 두 정점의 번호가 주어진다. 모든 간선은 양방향 간선이다.
 - 첫째줄에 DFS로 탐색한 결과를, 둘째줄에 BFS로 탐색한 결과를 출력한다.

Assignment 8 - Graph Search

정점 4개 (1~4)
간선 5개
시작은 1에서

입력 예시

4 5 1

1 2

1 3

1 4

2 4

3 4

m(=5)개의 간선 정보

출력 예시

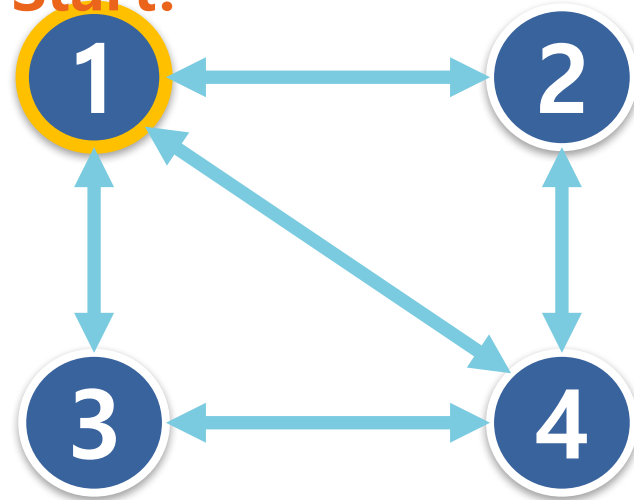
1 2 4 3

DFS 결과

1 2 3 4

BFS 결과

Start!



Graph Representation

정점 4개 (1~4)
간선 5개
시작은 1에서

입력 예시

4 5 1

1 2

1 3

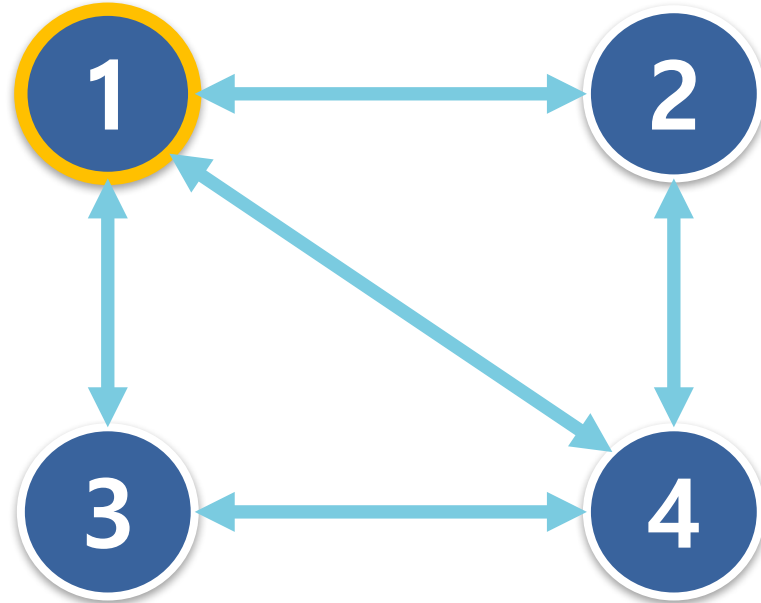
1 4

2 4

3 4

m(=5)개의 간선 정보

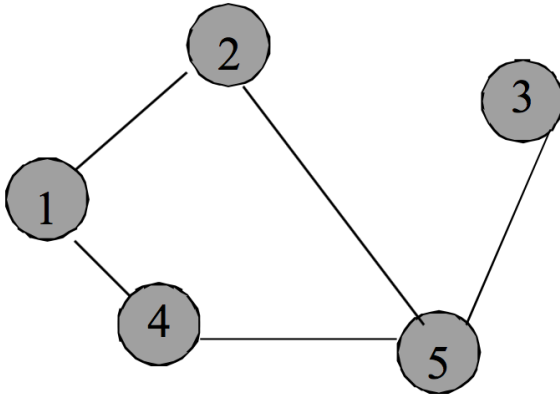
Start!



Graph Representation

Adjacency Matrix

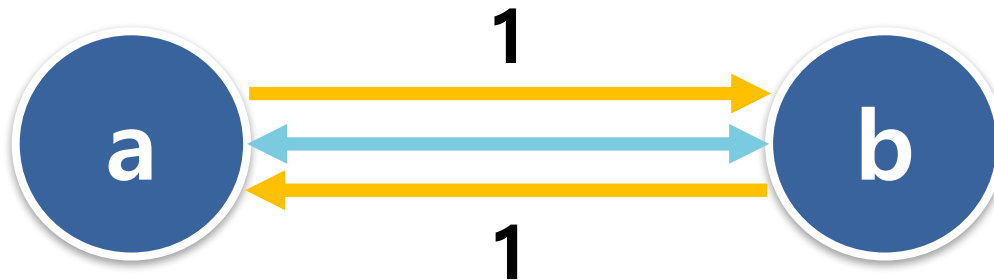
- 0/1 $n \times n$ matrix, where $n = \#$ of vertices
- $A(i,j) = 1$ iff (i,j) is an edge



	1	2	3	4	5
1	0	1	0	1	0
2	1	0	0	0	1
3	0	0	0	0	1
4	1	0	0	0	1
5	0	1	1	1	0

Graph Representation

```
int a, b;  
cin >> a >> b;  
node[a][b] = 1;  
node[b][a] = 1;
```



Depth First Search => Recursion

```
void dfs(int now)
{
    for (int j = 1; j <= n; j++)
    {
        int next = j;
        if (node[now][next] == 1)
        {
            if (visit[next] == false)
            {
                visit[next] = true;
                dfs(next);
            }
        }
    }
}
```


Breadth First Search => Queue

```
void bfs(int start)
{
    Queue Q

    Q.Enqueue(start)

    while (Q is not Empty)
    {
        int now = Q.Dequeue();
        for (int j = 1; j <= n; j++)
        {
            int next = j;
            if (node[now][next] == 1)
            {
                if (visit[next] == false)
                {
                    visit[next] = true;
                    Q.Enqueue(next);
                }
            }
        }
    }
}
```

Assignment 8 - Graph Search

- 제출방식: Assignment8 폴더 만들고
Assignment8_학번.c 파일 저장
Ex) Assignment8/Assignment8_2016000000.c
- GitLab(<https://hconnect.hanyang.ac.kr/>)으로 제출
- GitLab이 안되면 이메일(roomylee@naver.com) 제출
- 제출기한: 6월 8일 23시 59분

감사합니다
