

## Lab11 Graph Operations - (BFS)

- 그래프 데이터는 다음 Adjacency Matrix 를 사용할 것
- 출력 결과:

\*\*\*\*\* **Breadth First Search (BFS)**

**v0 v1 v2 v3 v4 v5 v6 v7**

\*\*\*\*\* **Adjacent Matrix**

	v0	v1	v2	v3	v4	v5	v6	v7
v0	0	1	1	0	0	0	0	0
v1	1	0	0	1	1	0	0	0
v2	1	0	0	0	0	1	1	0
v3	0	1	0	0	0	0	0	1
v4	0	1	0	0	0	0	0	1
v5	0	0	1	0	0	0	0	1
v6	0	0	1	0	0	0	0	1
v7	0	0	0	1	1	1	1	0

- 참고

```
int graph[max][max]={ {0,1,1,0,0,0,0,0} {1,0,0,1,1,0,0,0}.....}
int front=0,rear=0,queue[5];      char visited[max];

main()
{   initializeQ( );                // Queue 초기화
    print Adjacent Matrix;      // 그래프 데이터 출력
    for (i=0; i<max;i++) visited[i]='f';    // 방문검사 위한 배열을 초기화
    bfs(v); // 첫번 노드부터 시작.
}
```

- BFS ADT

- 1) bfs ( ) { 강의 노트 알고리즘 참조 }
- 2) initializeQ { 이전 lab Queue 알고리즘 참조 }
- 3) addQ ( ) { 이전 lab Queue 알고리즘 참조 }
- 4) deleteQ ( ) { 이전 lab Queue 알고리즘 참조 }
- 5) print\_matrix() { 데이터 출력 }