<DFS= 깊이 우선 탐색>

루트 노드(혹은 임의의 노드)에서 시작하여 다음 분기로 넘어가기 전에 해당 분기를 완벽하게 탐색하는 방법

1. 미로를 탐색할 때 한 방향으로 갈 수 있을 때까지 계속 가다가 더 이상 갈 수 없게 되면 다시 가장 가까운 갈림길로 돌아와서 이곳으로부터 다른 방향으로 다시 탐색을 진행하는 방법

2. 넓게 가기 전에 깊게

3. 모든 노드를 방문하고자 할 때 이 방법 선택

4. 검색 속도는 BFS에 비해 느림

- 자기 자신을 호출하느 순환 알고리즘의 형태를 지님

- 그래프 탐색의 경우 어떤 노드를 **방문 했었는지 여부를 반드시 검사해야 한다**는 것(이를 검사하지 않을 경우 무한루프에 빠질 위험이 있음)

.- DFS는 그래프의 모든 간선을 좋아함.(정점의 수:N, 간선의 수 : E)

- 인접 리스트로 표현된 그래프 : O(N+E)

- 인접 행렬로 표현된 그래프 O(N^2)

<BFS = 너비 우선 탐색>

루트 노드에서 시작해서(혹은 다른 임의의 노드) 인접한 노드를 먼저 탐색하는 방법

1. 시작 정점으로부터 가까운 정점을 먼저 방문하고 멀리 떨어져 있는 정점을 나중에 방문하는 순회 방법

2. 즉 깊게 탐색하기 전에 넓게 탐색하는 것

3. 두 노드 사이의 최단 경로 혹은 임의의 경로를 찾고 싶을 때 이 방법을 선택함.

- BFS는 재귀적으로 동작하지 않는다.

- 어떤 노드를 방문했었는지 여부를 반드시 검사해야한다는 것. 이를 검사하지 않을 경우 무한 루프에 빠질 위험이 있다.

- BFS는 방문한 노드들을 차례로 저장한 후 꺼낼 수 있는 자료구조인 큐(Queue)를 사용함.

- 선입선출 원칙으로 탐색