

CSE3013 (컴퓨터공학 설계 및 실험 I)

PRJ-2 미로 프로젝트 3주차 예비 보고서

서강대학교 컴퓨터공학과 박수현 (20181634)

서강대학교 컴퓨터공학과

1 목적

DFS와 BFS 알고리즘의 시간 복잡도를 보이고 설계한 자료구조에서 각각의 알고리즘을 이용해 어떻게 경로를 찾을 수 있는지 보인다.

2 문제

2.1 그래프 탐색 알고리즘의 시간 복잡도

그래프 $G = (V, E)$ 가 있다고 하자. DFS와 BFS는 모두 모든 정점을 한 번씩만 방문하는 알고리즘이므로 정점을 방문하는 데에는 $\mathcal{O}(\|V\|)$ 가 소요된다. 또한 정점에 대해 인접한 정점을 쿼리해 진행해야 하는데, 인접한 정점을 쿼리하는 연산은 어느 자료 구조를 사용하느냐에 따라 시간 복잡도가 달라진다. 인접 리스트를 사용할 경우 인접한 정점을 쿼리하는 연산의 시간 복잡도는 $\mathcal{O}(\|E\|)$ 이고, 인접 행렬을 사용할 경우 각 노드마다 연결 여부를 체크해야 하므로 전체 시간 복잡도는 $\mathcal{O}(\|V\|^2)$ 이다.

따라서 전체 시간 복잡도는 인접 리스트의 경우 $\mathcal{O}(\|V\| + \|E\|)$ 이고 인접 행렬의 경우 $\mathcal{O}(\|V\|^2)$ 이다.

2.2 탐색 알고리즘의 구현 방법

- **DFS** - 스택 S 를 정의한다. S 에 첫 노드를 넣고, S 의 첫 원소 s 에 대해 $N(s)$ 중 아직 방문하지 않은 정점들을 전부 S 에 추가하고, S 의 첫 원소를 제거한다. 이를 S 가 빌 때까지 반복한다.
- **BFS** - 큐 Q 를 정의한다. Q 에 첫 노드를 넣고, Q 의 첫 원소 q 에 대해 $N(q)$ 중 아직 방문하지 않은 정점들을 전부 Q 에 추가하고, Q 의 첫 원소를 제거한다. 이를 Q 가 빌 때까지 반복한다.

구현한 자료구조의 크기가 $col \times row$ 라 하면, 가장자리는 모두 벽이므로 맨 왼쪽 위 칸을 $(0, 0)$ 이라 할 때 첫 노드는 $(1, 1)$ 이고 마지막 노드는 $(col - 2, row - 2)$ 이다. 인접한 노드는 인접한 4방향의 칸 중 통로(' ' 문자) 칸으로 정의한다.