전공 : 컴퓨터공학과 학년 : 4 학번 : 20212020 이름 : 박민준

1. 문제 해결에서 언급한 미로 생성 알고리즘들에서 Eller's algorithm을 제외한 나머지 알고리즘 중 하나를 선택하여 이를 조사하고 이해한 후 그 방법을 기술하시오.

- Kruskal 알고리즘은 완전 미로를 생성하기 위해 많이 사용되는 방법 중 하나이다. 이 알고리즘은 미로를 그래프로 간주하고 최소 신장 트리(MST) 접근 방식을 사용하여 임의의 두 점 사이에 오직 하나의 경로만 존재하도록 보장한다. Kruskal 알고리즘의 절차는 다음과 같다.

텍스트, 도표, 직사각형, 사각형이(가) 표시된 사진

자동 생성된 설명

1) 미로의 각 방을 그래프의 노드로 간주하고, 방 사이의 모든 벽을 그래프의 엣지로 간주하여 리스트에 추가한다.

2) 벽 리스트를 무작위로 섞어 임의의 순서로 정렬한다.

3) 각 방을 독립된 집합으로 시작한다. 이는 분리 집합 자료 구조를 사용하여 구현할 수 있다.

4) 벽 리스트에서 하나씩 벽을 꺼내어 다음을 수행한다

(1) 벽이 나누는 두 방이 다른 집합에 속해 있다면, 벽을 제거하고 두 집합을 합친다.

(2) 같은 집합에 속해 있다면, 벽을 그대로 둔다.

5) 벽 리스트가 빌 때까지 위의 과정을 반복한다.

- 이 과정을 통해 생성된 미로는 임의의 두 점 사이에 하나의 경로만 존재하는 완전 미로가 된다.

2. 본 실험에서 완전 미로(Perfect maze)를 만들기 위하여 선택한 알고리즘 구현에 필요한 자료구조를 설계하고 기술하시오. 설계한 자료구조를 사용하였을 경우 선택한 알고리즘의 시간 및 공간 복잡도를 보이시오.

1) Eller's 알고리즘을 효과적으로 구현하기 위해 필요한 자료구조는 다음과 같다

- 집합 배열은 각 행의 방들을 관리하기 위해 사용된다. 각 방은 특정 집합에 속하며, 이 집합은 배열로 관리된다. 예를 들어, sets[i]는 i번째 방이 속한 집합을 나타낸다. 이 자료구조는 집합 간의 결합 및 비교 작업을 용이하게 한다.

- 벽 리스트는 현재 행에서 제거되거나 유지될 벽들의 리스트이다. 이는 인접한 방들 사이의 벽을 추적하고, 무작위로 제거할 벽을 선택하는 데 사용된다.

- 연결 리스트는 각 행의 방들 간의 연결 상태를 추적한다. 이는 수직 연결을 만들고 다음 행으로 집합을 전달하는 데 사용된다.

2) Eller’s 알고리즘의 시간 및 공간 복잡도는 다음과 같다.

(1) 시간 복잡도

- 첫 번째 행에서 각 방을 서로 다른 집합에 초기화하는 작업은 O(N)이다.

- 각 행에서 벽을 제거하는 작업은 O(N)이다.

- 각 행에서 수직 경로를 생성하는 작업 역시 O(N)이다.

- 총 M개의 행에 대해 위의 작업을 반복하므로 **전체 시간 복잡도는 O(M \* N)이다.**

(2) 공간 복잡도

- N개의 방에 대해 각 방의 집합을 저장하는 데 O(N) 공간이 필요하다.

- N개의 방 사이의 벽을 저장하는 데 O(N) 공간이 필요하다.

- 각 행의 연결 상태를 저장하는 데 O(N) 공간이 필요하다.

- **전체적으로 공간 복잡도는 O(N)이다.** 이는 매 행마다 새로운 집합 배열과 벽 리스트를 생성하기 때문이다.

- 여기서 N과 M은 미로의 너비와 높이를 의미한다.

- N은 미로의 열(column) 수를 나타낸다. 즉, 미로의 가로 길이를 의미한다. 또한, M은 미로의 행(row) 수를 나타낸다. 즉, 미로의 세로 길이를 의미한다. 예를 들어, N이 5이고 M이 3이라면, 미로는 5개의 열과 3개의 행으로 구성된 5x3 크기의 미로가 된다.

- 정리하자면, Eller's 알고리즘을 위한 자료구조로는 집합 배열, 벽 리스트, 연결 리스트가 필요하다. 이를 통해 효율적인 미로 생성을 구현할 수 있다. 알고리즘의 시간 복잡도는 O(M \* N)이고, 공간 복잡도는 O(N)이다. 이는 미로의 크기에 비례하여 성능이 결정되므로 대규모 미로에서도 실용적이다.