|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 1 주차 | **기간** | 6.29 ~ 7.2 | **지도교수** | (서명) |
| 이번주 한일 요약 | 자료구조 복습 및 그래프(Dijikstra, DFS, BFS) 알고리즘 구현 | | | | |

<상세 수행내용>

이번 주차에서는 지난 학기에 했던 자료구조의 내용 중 졸업작품에서도 사용할 것으로 예상했던 길찾기 알고리즘을 구현하고자 했습니다. 밑의 그림을 이용해 대표적인 그래프 형식의 DFS알고리즘, BFS알고리즘, 다익스트라 알고리즘을 구현 했습니다.

원, 스크린샷이(가) 표시된 사진

자동 생성된 설명

이중 벡터를 사용해 인접한 정점들을 구분해 구현할 수 있었습니다. DFS 알고리즘은 인접한 길을 체크를 통해 재귀함수를 통한 구현을 해보았습니다. 위 사진대로 DFS 알고리즘은 0번에서 시작해, 인접한 1번과 3번을 체크 해 처음 1번의 호출에서 또 인접한 2번과 3번을 체크할 수 있어 찾을 수 있도록 이런 식으로 깊게 들어가게 끔 구현해 보았습니다. BFS 알고리즘은 큐를 통한 구현을 해보았습니다. BFS 알고리즘은 처음 들어간 위치에서 가까운 길부터 탐색할 수 있도록 구현했습니다. 큐를 통해 처음 발견한 위치를 넣고 큐가 비어있는 지만을 체크해 비어있지 않다면 인접한 위치를 찾으며 큐에서 하나씩 POP을 하는 방식을 사용했습니다. 다익스트라 알고리즘은 밑의 사진을 통해 가중치를 활용한 알고리즘을 구현했습니다. 따라서 BFS 알고리즘과 유사하지만 우선순위 큐를 사용했습니다.

|  |  |  |  |
| --- | --- | --- | --- |
| **문제 점 정리** | 기본적인 알고리즘을 통해 길 찾기를 수행했으나 정확한 목적지를 찾는 알고리즘인 A\* 알고리즘을 구현하지 못함 | | |
| **해결방안** | X | | |
| **다음주차** | 2 | **다음기간** | 7.03 ~ 7.10 |
| **다음주 할일** | Windows API, 게임수학 공부를 통한, 3.4주차 서버 연동에 필요한 간단한 클라이언트 제작 | | |
| **지도 교수**  **Comment** |  | | |