길 찾기 알고리즘

BFS 너비우선 탐색을 이용해 최단경로를 찾는 알고리즘을 사용해 길찾기 알고리즘을 만들었습니다

하지만 처음 갔던 길을 체크하는 배열 맵을 만들지 않고 전에 있었던 방향으로 되돌아가지만 않게 구현했는데 속도가 상당히 느렸습니다 그제야 제대로 글을 읽어보고 배열을 구현해 잘 작동하는 길찾기 알고리즘을 만들 수 있었습니다. 하지만 아직 배우지않은 다익스트라, A\* JFS 같은 알고리즘도 공부해봐야 합니다.

생각보다 너무 어려워서 힘들어 이런 부분이라도 보여드리기 위해 만들었지만 아쉬운 부분이 너무 많습니다.

일단 구현부터 되면 되지 않을까 라는 마음으로 코딩을 한 것이

수많은 버그와 오류, 수정에도 어려움을 겪었습니다

게다가 코드에 일관성이나 주석도 많지 않아 더욱 어려웠습니다

특히 디자인패턴이나 이벤트 함수, 클래스는 어떻게 분류하는지 어려워 했습니다. 조금은 그래도 감을 찾아가는 것 같습니다.

처음, 더블버퍼링과의 만남

렌더링

모든 Actor를 돌며 백버퍼에 그려질 문자 대입

StringBuilder와 백버퍼를 이용한 렌더링

frontBuffer는 backBuffer은 char배열

StringBuilder로 줄바꿈을 추가하고 한번에 Write

더블버퍼링을 보고 이런 방식으로 전부 그리는게 가능하다는 것을 보고 놀라움을 느꼈습니다.

하지만 처음 시도 당시 구현에 윈도우API를 사용하는 법, 전달하는 문자의 데이터 형, 인코딩 차이 같은 문제로 힘들어 다른 방법으로 구현했습니다. 하지만 지금은 아스키코드만 그리기 때문에 이런 방식이면 가능할 것 같습니다.

굳이 왔다 갔다 구현해서 시간이 더 오래 걸렸지만 새로 알게 된 점이 많았습니다.

이해하기 까지는 시간이 더 걸릴 것 같습니다.

자동 비율 UI, UI 커서, 욕심

내부 행,열 지정

크기 비율 조정

테두리 그리기

내용물 그리기

이벤트 할당

Ui컨테이너에 자식들을 넣으면 자동으로 크기가 정해지는 방식을 원했습니다. 실제로 구현은 가장 큰 컨테이너의 크기에서 내용물의 행과 열의 비율을 지정해주고 그 크기에 맞게 추가되는 방식이었습니다.

하지만 UI를 그리는 부분이나 내용물의 출력 범위 모두 어설프게 만들어져 있어 조금만 어긋나도 제대로 그려지지 않고 그려지는 위치가 다른 문제가 있습니다. 내용물에 따라 버그가 많아 여러번의 시도 끝에 그럴싸한 장면만 보여지게 만들었습니다.

이곳에 많은 고민과 노력이 들어간것에 비해 아쉬운 결과라고 생각합니다. 일단 구조도 너무 허술하고 복잡해서 수정할때마다 애먹었고 아직도 수정할 부분이 많습니다.

처음 설계를 잘해야하고 왠만하면 클래스들 다양하게 하지만 직관적으로 접근할 수 있게 나누는 편이 더 좋을거라고 생각하게 되었습니다.

전투, 로직

이 부분은 그냥 기초적인 공격, 적의 죽음, 아이템 드랍, 인벤토리, 확률적 도망등의 기능을 구현했지만 아직 거래, 스킬, 장비, 레벨, 포획 같은 부분들은 하나도 구현하지 못했고 결과적으로 플레이가 가능한 상태의 게임이 아닌 결과가 나와 아쉬웠습니다.