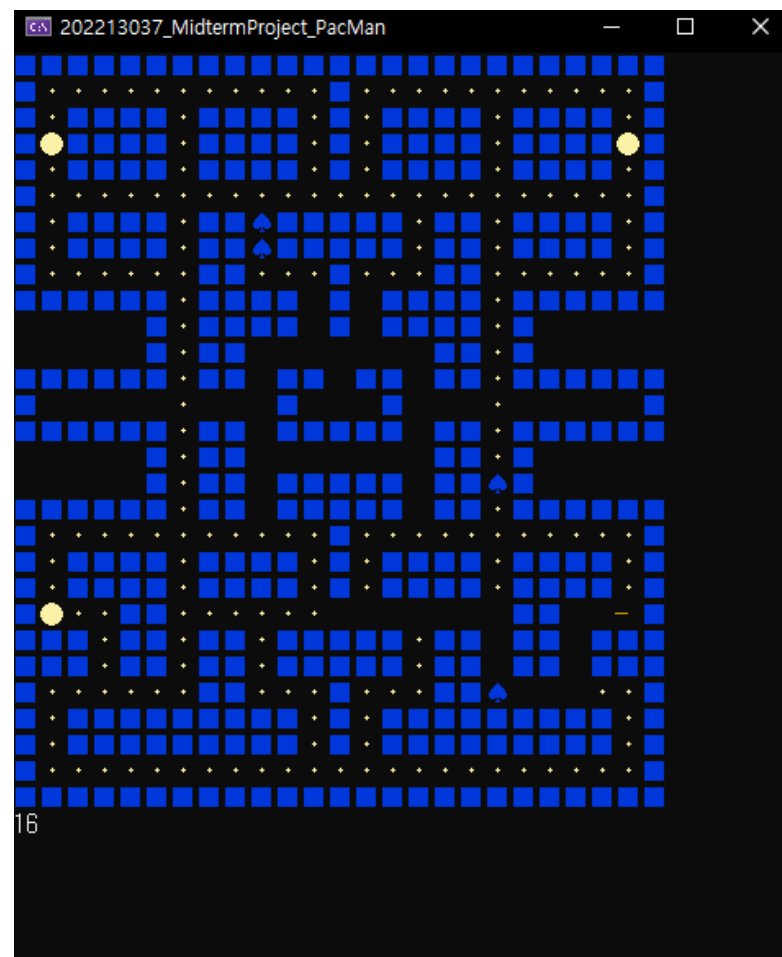
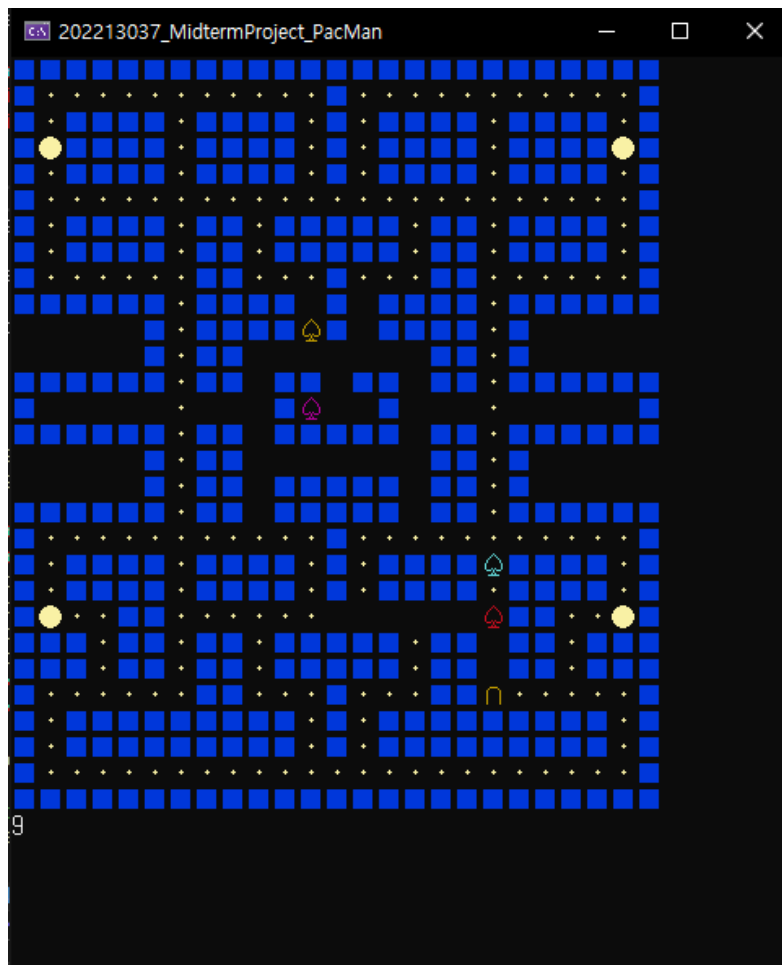




게임인공지능

202213037 이수연

플레이 화면



기본 시스템

- GameLogic : 게임의 전반적인 실행
- MapLogic : 게임의 기본 맵 관리
- ScreenLogic : 게임을 화면에 표시
- GameObject : 움직이는 오브젝트들의 부모
 - Player : 플레이어블 캐릭터 
 - Ghost : 플레이어 적대시하는 캐릭터 
 - FSM : State 패턴을 이용해 구현
 - A* : 최단거리 알고리즘

게임 로직

- 기본적으로 맵에 대한 파악이나 전체 시스템을 담당합니다
- Update() : 게임의 흐름을 담당 (매 프레임 실행)
- Draw() : 맵 및 GameObject들의 좌표를 스크린 로직에 전달
- Check() : 게임이 진행될 때의 변수들을 확인
- Input() : 입력 확인

맵 로직

- 파일 I/O를 이용하여 맵 파일을 불러옵니다. (PacMan.dat)
- 기본 구조물(벽, 먹을 것등 정적 오브젝트)들의 위치들을 관리합니다
- 외에도 정적 함수로 위치와 방향관련 변환을 가지고 있습니다

스크린 로직

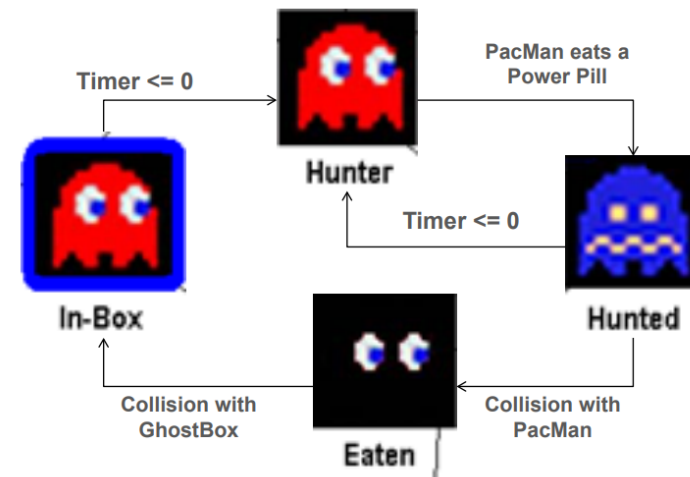
- 더블 버퍼링을 이용하여 게임 화면을 관리합니다
- 모든 화면 출력을 담당합니다

GameObject

- 움직이는 객체들의 부모 클래스입니다
- Update() : 매 프레임 실행
- Move() : 객체들의 움직임을 담당
 - 기본적으로 direction을 파라미터로 받아 움직이므로 움직이는 방식이 다른 고스트와 플레이어 모두 해당 함수를 통해 움직입니다
- StateUpdate(), SetState() : 맵에 출력되는 오브젝트의 상태관리
 - 이를 통해 플레이어가 애니메이션을 가질 수 있습니다
 - 고스트의 아이콘 변화 또한 해당 함수를 통해 이루어집니다

Ghost

- State 패턴으로 구현된 FSM을 통해 현재 상태와 아이콘, 변수등의 상태 해제 (bool) 를 관리합니다.
- 길찾기의 경우 AStar클래스에서 알고리즘 실행 후, GetDirection() 함수로 방향을 받아와 Move()로 움직입니다



수정된 부분

- 실습시간에 진행했던 A* 코드를 약간만 수정하여 사용하려 했으나, 알고리즘이 너무 무거워 게임 자체가 진행이 불가할 정도로 렉이 걸렸었습니다.
 - 함수 호출 빈도 문제가 아닌, 게임 실행부터 렉이 발생했습니다.
- 결국 우선순위 큐를 이용하여 해당 알고리즘의 최적화 작업을 진행하였고, 이후로는 ghost의 개수를 늘려도 더 이상 렉이 걸리지 않았습니다.