

그래픽스 PONG게임 제작 리포트

B477026 유현지

1) 게임 기능

- 스테이지 1,2로 구성되어 있다.
- AI모드와 PVP모드가 존재한다.
- ready 상태에서 왼쪽 shift를 눌러 모드를 변경할 수 있다.
(디폴트는 AI 모드, 현재 모드를 상단에서 확인할 수 있다.)
- ready 상태에서 space bar를 누르면 게임이 시작된다.
- AI 모드에서는 오른쪽 컨트롤 바를 방향키로 조절, PVP 모드에서는 왼쪽은 WASD, 오른쪽은 방향키로 조절할 수 있다.
- 왼쪽, 오른쪽의 스코어와 현재 스테이지, 모드가 상단에 표기된다.
- 공이 양 옆의 벽면에 닿으면 상대방이 점수를 얻고, 3점을 먼저 달성한 쪽이 승리한다.
- 공이 벽면에 닿은 순간 스코어가 올라가며 ready상태(오브젝트 포지션 초기화)로 전환된다.
- 어느 한 쪽이 승리하면 스테이지가 변경되며, 가운데에 [000 Win] 메시지가 나타난다.
(메시지는 space bar를 눌러 게임을 시작하면 사라진다)

2) 구현 과정

윈도우 크기 조정 -> 스카이 박스&카메라 세팅 -> 오브젝트 구현 -> 컨트롤 바 키보드 조정&갈 수 있는 범위 지정 -> 공과 컨트롤 바의 Bounding Box를 통한 AABB 충돌 처리 구현 -> 공의 사실적인 움직임 구현 -> 게임 스코어, 스테이지, 모드 Label 구현 -> 스테이지2 추가 -> AI 구현 -> 3점을 달성한 쪽의 Win(승리)메시지 구현 -> 스테이지 순환(1-2-1-2-1-2-....)구현 -> AI모드와 PVP모드 스위칭 구현

3) 어려웠던점

1. 윈도우 크기를 조정할 때 기본 BaseApplication에 autoCreateWindow가 true로 설정되어있어 createRenderWindow함수로 직접 윈도우를 생성할 수 있다는걸 찾는데 시간이 오래 걸렸다.
2. 충돌 처리에 대해 고민하다가 Unity의 BoundingBox를 떠올리고, BoundingBox를 통한 충돌 처리를 AABB라 칭한다는걸 알게되었다.
3. AABB로 단순 충돌은 구현했지만 좀 더 자연스러운 움직임을 위해 계속 검색을 하던 도중 <http://rembound.com/articles/the-pong-tutorial> 에서 공의 움직임, AI 구현에 큰 도움을 받았다.
4. 사이트에서는 2D로 충돌처리를 했는데, 오우거는 3D이므로 오브젝트들의 BoundingBox의 z좌표를 2D상의 y좌표로 두고 충돌, 움직임, AI를 구현했다.
5. Label구현에서 가장 큰 어려움이 있었다. win(승리) 메시지를 Label로 구현하고 싶었지만, Label을 완전히 없애는 방법을 찾지 못해 win(승리) 메시지를 panel로 구현하고, 스코어&스테이지를 Label로 구현했다.
6. 스테이지 1,2가 계속 순환되게 하는게 어려움이 있었다. 스테이지, 스코어 Label 변경과 win(승리) 메시지 변경이 계속 꼬여서 bool 변수를 많이 썼고, 결과적으로 코드가 지저분해졌다.
7. AI-PVP모드 변경 시 shift가 한 번만 눌리게 하는걸 실패해서 Label로 현재 모드를 보이게 했다.