**개발 결과 보고서**

Project: golf

2023182016 박이언

2023184023 이수빈

**프로젝트 소개**

이 프로젝트는 2D 및 3D 그래픽 환경에서 골프 게임을 구현한 프로그램으로, OpenGL을 사용하여 개발되었습니다. 사용자 입력에 따라 골프공을 움직이며 여러 장애물을 피하고 스테이지를 클리어하는 목표를 가지고 있습니다. 프로젝트는 4개의 스테이지로 구성되어 있으며, 각 스테이지는 장애물과 특이한 환경을 포함하고 있습니다. 게임은 키보드를 사용하여 조작하며, 공의 물리적 움직임과 충돌 처리를 구현했습니다. 병맛 골프 게임인 ‘What The Golf?’를 모티브 삼아 만든 프로젝트이며 평범한 골프 게임과 다르게 다양한 상황을 만날 수 있도록 만들었습니다.

**구조 소개**

**-** 기본 화면 설정

800x800화면 비율

연한 청록색 배경

glClearColor(0.698f, 0.886f, 0.871f, 1.0f);로 설정해 모티브 게임과 유사한 배경으로 구성

- 클래스 및 구조체  
AABB 구조체

충돌 처리를 위해 AABB(Axis-Aligned Bounding Box)를 사용했습니다. AABB는 객체의 최소 및 최대 좌표를 저장하여 간단한 충돌 검사를 합니다.

- 골프공 및 장애물 관리

골프공과 장애물의 위치, 크기, 상태는 glm::vec3를 사용했습니다.

spherePosition: 공의 현재 위치

obstacle1Position, obstacle2Position: 장애물의 위치

GoalTransForm: 골대의 변환 행렬

cameraPos: 카메라 위치

- 행렬과 변환 설정

projectionTransform(원근 투영), viewTransform(카메라 위치), modelTransform(객체 변환) 행렬을 계산하고 이를 셰이더에 전달합니다. 이를 통해 3D 공간에서 카메라와 객체의 위치 및 크기를 설정합니다.

- 맵 스테이지별 렌더링

Stage 0: 타이틀 화면

Stage 1-4: 다양한 장애물, 목표 깃대, 구체 등을 렌더링합니다. 각 스테이지에서는 변환 행렬을 설정하고, 각 객체의 위치와 크기를 변경하여 그립니다.

Stage 5: 엔딩 화면

- 프로그램 흐름

초기화

OpenGL 초기화와 셰이더를 설정한 후, 버텍스 데이터를 GPU로 전송합니다.

입력 처리

키보드 및 마우스 입력을 처리하여 공의 이동, 카메라 회전, 스테이지 전환을 구현했습니다.

렌더링

각 스테이지의 맵과 장애물을 렌더링하며, 공의 위치에 따라 카메라가 움직이도록 설정했습니다.

충돌 처리

공과 장애물, 골대 간의 충돌 여부를 검사하여 게임 진행을 관리했습니다.

**프로젝트 진행 사항**

- 타이틀 및 엔딩

타이틀: 게임 이름(Project:golf)과 조작 키 설명이 있습니다.

엔딩: 플레이 타임 시간이 나옵니다.

- 스테이지 구성

스테이지 1: 단순한 직육면체의 맵과 골대가 배치되어 있습니다.

스테이지 2: 맵이 조금 더 길어지고, 장애물이 추가됩니다. 공은 장애물을 피해서 골을

넣어야 합니다.

스테이지 3: 스테이지 2과 같은 맵과 장애물이지만 깃대가 도망가는 기믹이 추가되었습니다.

스테이지 4: 스테이지 1과 같아 보이지만 가짜 골대와 높이 차가 있는 맵을 이용하여 골을 넣도록 하였습니다.

- 주요 구현 사항

충돌처리: AABB기반 충돌 검사를 통해 공과 깃대, 장애물 간 충돌 처리를 하였습니다.

카메라 이동: 공을 중심으로 카메라가 이동하도록 구현했습니다.

애니메이션: 공이 이동하도록 보이게 애니메이션을 사용하였습니다.

- 추가 구현 사항

플레이 타임 시간 추가: 엔딩 화면에 뜹니다.

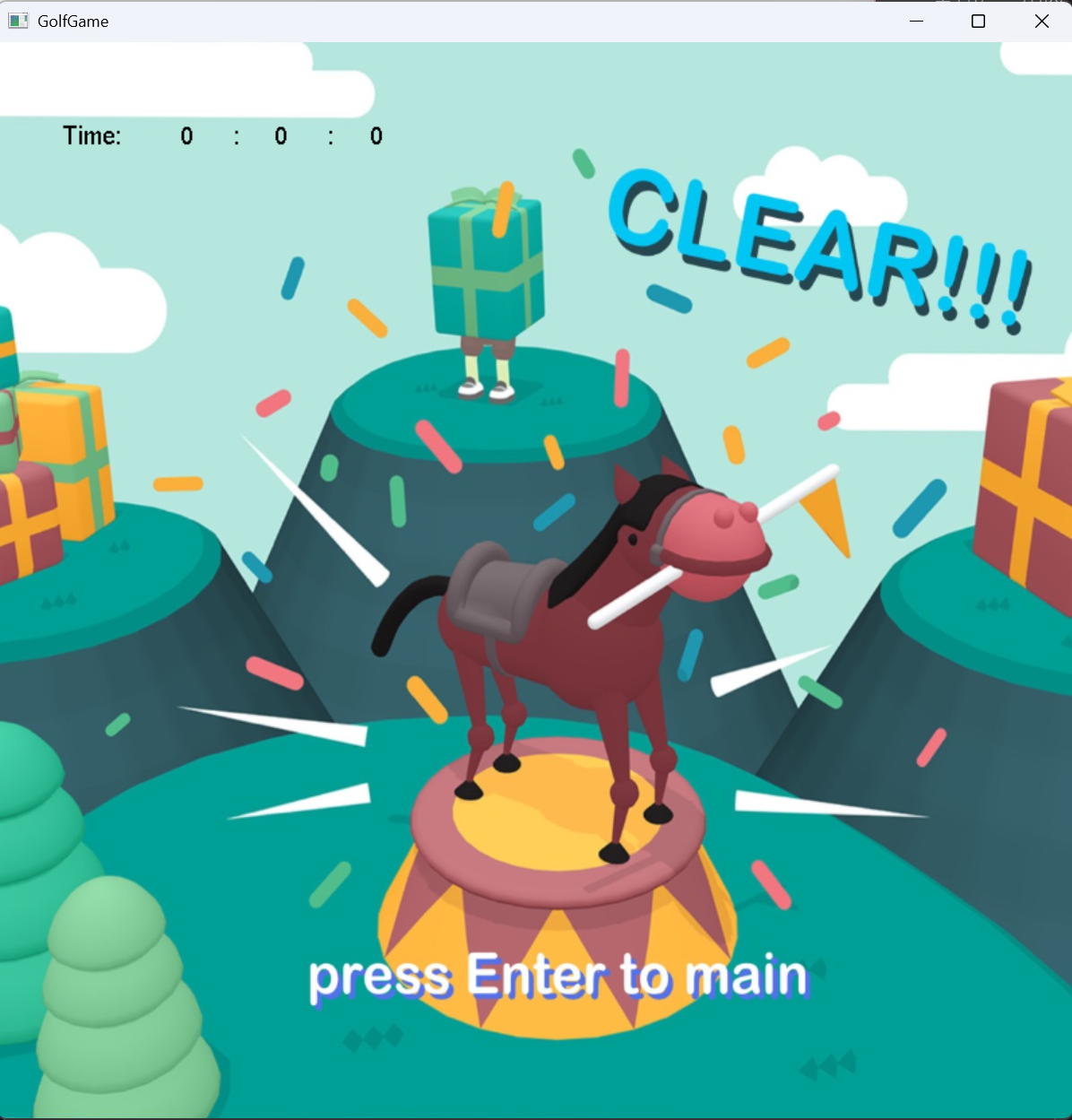
Q: 종료만 되는 게 아니라 타이틀로 이동도 가능

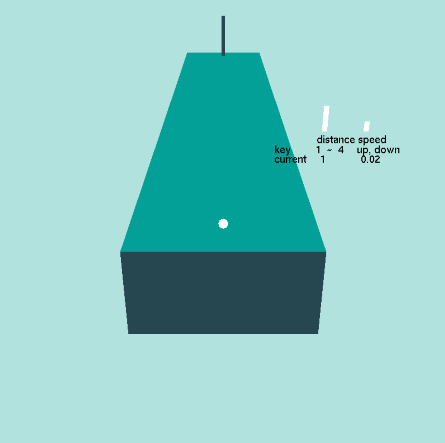
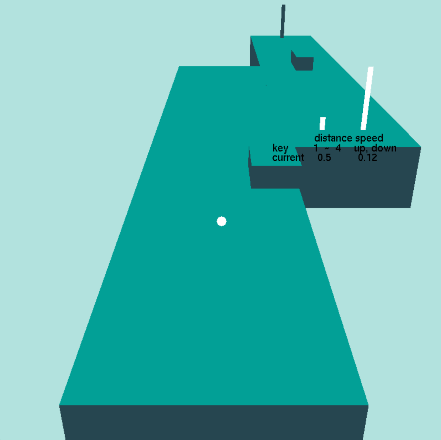
- 이스터에그 명령어

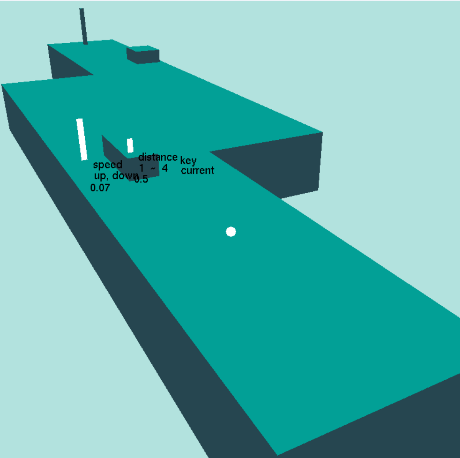
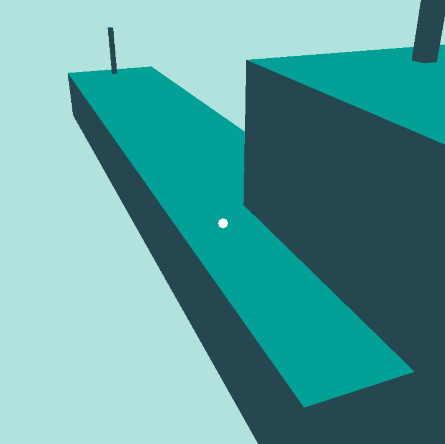
H: 은면제거 ONOFF

Y: 와이어 객체 ONOFF

**스크린샷 및 결과 소개**

**** 타이틀  엔딩

 1 스테이지  2 스테이지

 3 스테이지  4 스테이지

**명령어 소개**

키보드 입력으로 이루어져 있습니다.

Enter: 게임 시작

공 이동

W: 앞으로 이동, S: 뒤로 이동,A: 왼쪽으로 이동, D: 오른쪽으로 이동

1, 2, 3, 4: 이동 거리 설정

위/아래 화살표: 공 이동 속도 증가/감소

R: 공 위치 초기화

카메라 회전

왼쪽/오른쪽 화살표: 카메라 회전

C: 카메라 위치 초기화

모드 변경

H: 은면 제거 토글, Y: 와이어프레임 토글

Q: 타이틀에서는 종료, 스테이지에서는 현재 진행도 저장하고 타이틀로 이동

**개발 소감**

**박이언**: 좋아하던 게임을 모티브로 직접 구현할 수 있다는 점에서 이번 프로젝트는 매우 재밌었습니다. 개발 과정에서 예상치 못한 문제나 어려움도 많았지만, 팀원들과 함께 해결해 나가며 큰 뿌듯함과 성취감을 느낄 수 있었습니다. 특히 협업을 통해 서로의 아이디어를 공유하고 발전시키는 과정이 인상 깊었으며, 이러한 경험이 앞으로의 프로젝트에서도 큰 도움이 될 것 같습니다.

**이수빈**: 이번 프로젝트는 처음에는 어려움이 많았지만, 문제를 해결해 나가며 많은 것을 배울 수 있었습니다. 특히 충돌 처리, 애니메이션, 카메라 이동, 텍스쳐 매핑 등을 구현하며 컴퓨터 그래픽스의 원리를 깊이 이해하게 되었습니다. 이 중에서도 가장 큰 배움이 느껴졌던 것은 텍스쳐 매핑쪽이었던 것 같습니다. 그럼에도 조금 아쉬운 점을 꼽자면 조명을 구현하지 못한 것이 아쉬웠습니다. 이 게임에 조명을 넣으면 원하던 착시가 나오지 않아 넣지 않았는데 그래도 완성된 결과물을 보니 한 학기의 노력이 헛되지 않았음을 느낄 수 있었고, 앞으로 더 큰 도전을 할 수 있는 자신감을 얻었습니다.