^ㅡ^팀 개발 결과 보고서

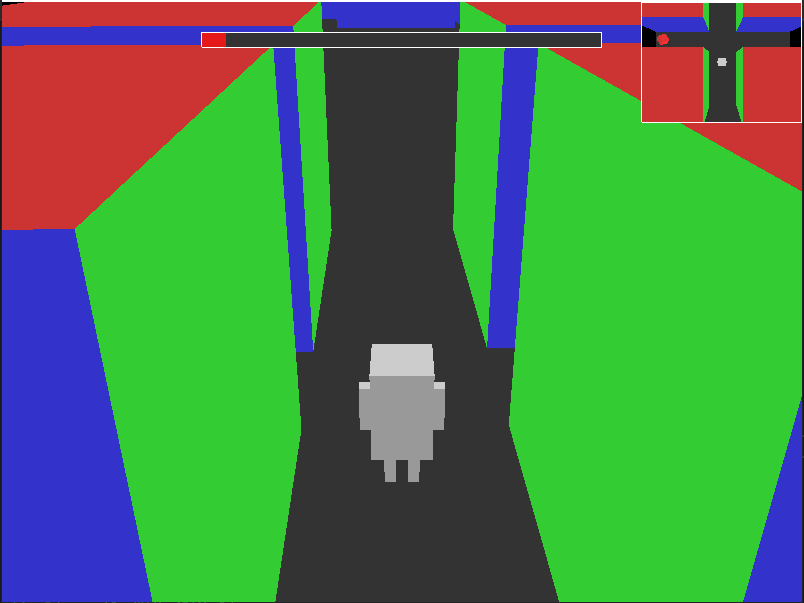
1. 프로젝트 소개
2. 프로젝트 구조
3. 팀원간의 작업 내용
4. 결과물
5. 명령어 및 실행 방법
6. 개발 소감 및 후기
7. 유튜브 링크

2017182023원윤식, 2017180045홍찬우

프로젝트 소개

저희 팀은 3D 런닝 게임의 형식으로 만드는 것을 목적으로 했습니다. 런닝 게임이다보니 특별한 3D 효과가 필요한 곳은 없다고 생각했습니다. 그래서 기본적인 모델, 투영, 뷰 변환만 적용했습니다. 여러가지 효과 대신, 이 프로젝트에서 신경썼던 것은 “카메라” 입니다. 카메라는 숄더뷰의 형태로, 마우스 드래그로 카메라를 이동시킬 수 있습니다. 좌우 드래그에는 각도 제한이 없고 위, 아래로는 각도 제한이 있습니다. 또한 마우스 휠로 카메라의 위치를 앞, 뒤로 움직일 수 있게 만들었습니다. 마지막으로 우측 상단에 탑뷰 미니맵을 표시했습니다.

중앙 상단에는 캐릭터의 HP, 우측 상단에는 미니맵이 있습니다.  
또한 게임은 숄더뷰로 진행됩니다.

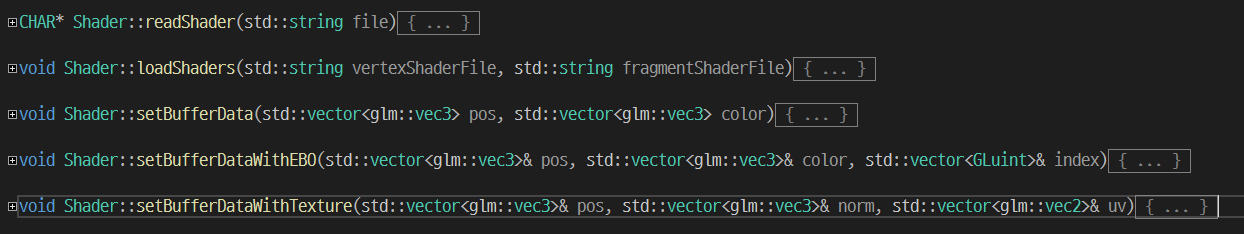


인게임 화면

프로젝트 구조

기본적인 쉐이더와 카메라는 FRAMEWORK로 분류했습니다.

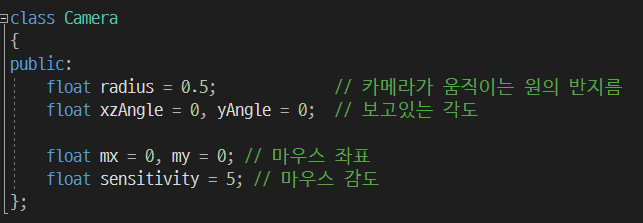
“쉐이더”



src/FRAMEWORK/Shader.cpp

쉐이더 class에서 가장 많이 사용되는 것은 setBufferData(std::vector<glm::vec3> pos, std::vector<glm::vec3> color) 함수입니다. 정점 벡터와 색상 벡터를 받아서 버퍼에 데이터를 입력합니다. EBO를 사용하는 함수와 텍스쳐를 사용하는 함수도 있지만 이 프로젝트에서는 사용되지않았습니다.

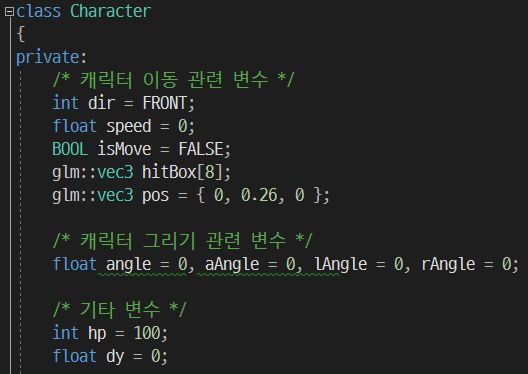
“카메라”

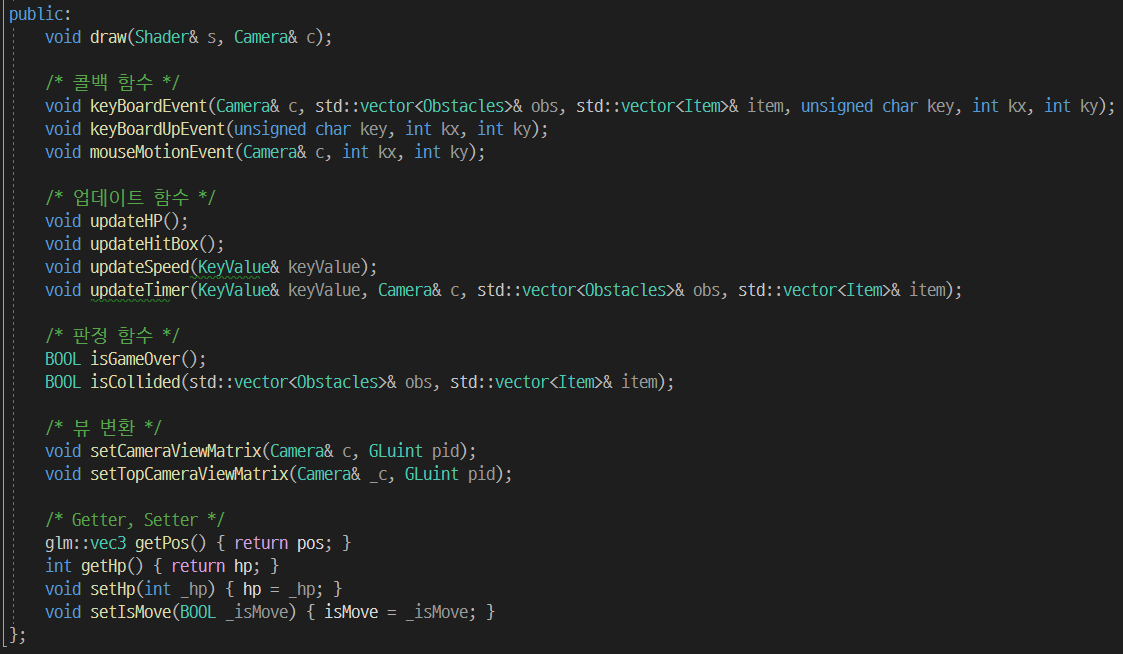


src/FRAMEWORK/Camera.h

class Camera에는 당연히 카메라에 관련된 변수가 있습니다. 밑에 마우스 좌표와 감도는 마우스 드래그로 카메라를 움직일 때 사용되는 변수입니다.

“캐릭터”

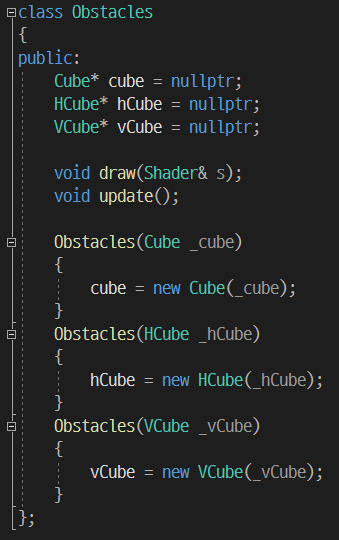




src/INGAME/Chracter.h

가장 중요한 class Character입니다. 캐릭터와 관련된 것은 모두 여기에 있습니다. ‘캐릭터 관련 키보드 콜백 함수’, ‘체력, 히트박스, 속도 업데이트’, ‘게임 종료와 충돌 판정‘, ‘캐릭터 위치에 따른 카메라의 뷰 변환 적용’을 이 class에서 처리합니다.

“장애물”

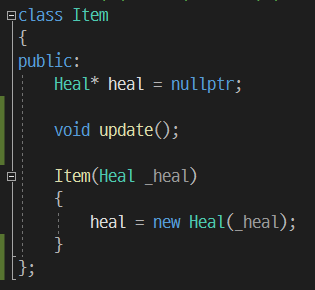


src/INGAME/Obstacle.h

장애물을 관리하는 class Obstacles입니다. Cube는 정육면체, HCube는 x축으로 움직이는 정육면체, VCube는 z축으로 움직이는 정육면체입니다. VCube는 게임 난이도에 직접적인 영향을 끼치지 않는다 판단하여 구조물로 넣진 않았습니다. 3개의 class는 기본적으로 [중점의 좌표(glm::vec3), 반지름(float)]을 입력받아 생성됩니다.

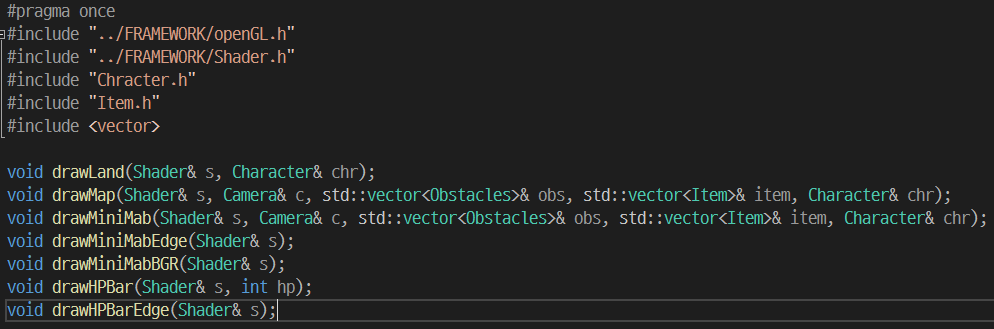
이렇게 모든 종류의 장애물을 관리하는 class를 만든 이유는 std::vector에 담아서 쉽게 관리하기 위해서입니다.

“아이템”

  
src/INGAME/Item.h

아이템을 관리하는 class Item입니다. Obstacles와 비슷한 구조입니다. 구현 아이템은 체력을 회복하는 한 종류의 아이템이지만 혹시 더 만들 수 있을 때도 쉽게 대응할 수 있는 구조를 만들었습니다.

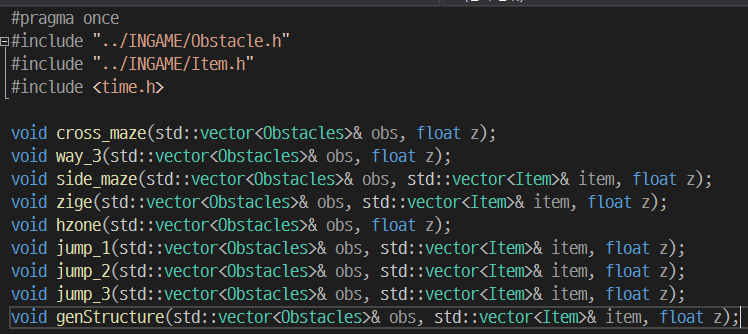
“그리기”



src/INGAME/Painter.h

그리기 관련 함수는 모두 여기에서 처리됩니다. HP바, 테두리를 그릴 때는 투영 변환이 없는 쉐이더를 사용해서 2차원으로 그립니다. 또한 미니맵을 그릴 때 인게임 화면을 그린 후 glClear(GL\_DEPTH\_BUFFER\_BIT)를 호출함으로서 미니맵이 인게임 지형에 가려지는 상황을 제거했습니다. 그리고 장애물과 아이템은 캐릭터와의 거리가 멀리 떨어져있을 경우 그리지 않습니다. UI들은 배경 택스처에서 배운것 처럼 변환 및 이동에 적용되지 않게 설정햇습니다.

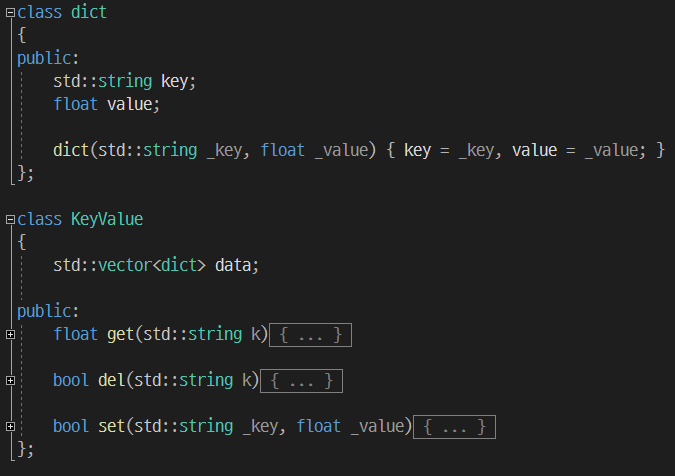
“장애물, 구조물 생성”



src/INGAME/Generator.h

이곳에서 미리 지정된 구조물들을 생성합니다. 구조물은 캐릭터가 일정 거리에 들어왔을 경우 랜덤하게 생성됩니다. 즉 캐릭터 일정거리 이상은 생성되지 않은 상태 입니다. 몇몇 구조물들은 랜덤하게 좌우가 대칭되어 생성될 수도 있습니다.

“KeyValue”



src/keyValues.h

파이썬에 있는 dict와 비슷한 기능을 하는 class KeyValue입니다.

get(key)는 value를 return하고,

set(key, value)는 key에 해당하는 데이터가 없는 경우엔 생성,

있는 경우엔 기존 데이터의 value를 새로운 value로 수정합니다.

del(key)는 key와 같은 dict를 삭제합니다.

캐릭터의 HP 닳는 속도, 카메라의 반지름 값을 조정 등을하는데 사용합니다.

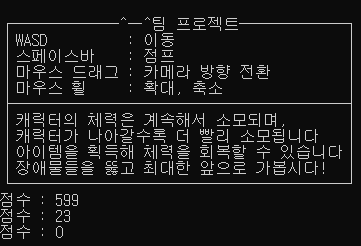
팀원간의 작업 내용



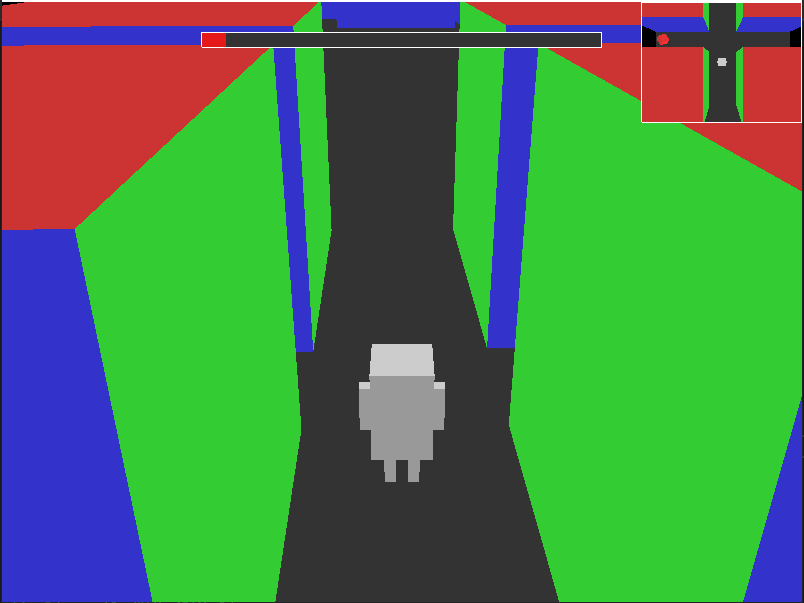
팀 프로젝트를 위해 Git을 이용했으며 각자 브랜치를 따서 진행한 뒤 나중에 약간의 수정을 거쳐 Main 브랜치로 합병했습니다. 또한 커밋할 때 어떤 내용을 구현했는 지 적음으로서 상대방에게 어떤 것이 바뀌었는 지 확인할 수 있게 했습니다. 원윤식, 홍찬우의 작업 내용은 다음과 같습니다.

|  |  |  |
| --- | --- | --- |
|  | 원윤식 | 홍찬우 |
| 작업 내용 | 프레임워크 제작 | 프레임워크 분석 |
| 카메라 이동 | 장애물 |
| 캐릭터 애니메이션 | 구조물 |
| 충돌 판정 | 구조물 생성 알고리즘 |
| 최종 디버깅 | 피드백 |

결과물

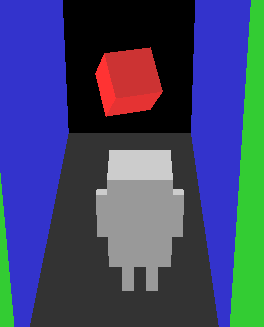


콘솔창에 조작법과 게임 설명을 출력하게 했습니다. 또한 체력이 다 닳거나 떨어지면 콘솔창에 점수를 출력합니다.



인게임 화면

정해진 구조물들이 랜덤으로 생성되면서 앞으로 나아갑니다. 몇몇 구조물 사이에는 체력을 획복할 수 있는 아이템도 있습니다. 사진의 경우 캐릭터 시점으로는 볼 수 없지만 미니맵을 보면 미니맵 기준 좌측에 아이템의 위치가 있다는 것을 알 수 있습니다. 플레이어는 숄더뷰만으로 진행할 수도 있지만 이렇게 미니맵을 활용해서도 진행할 수 있습니다.



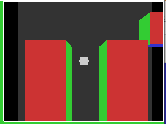
체력 회복 아이템

체력 회복 아이템은 y축으로 회전하고 있으며 위아래로 반복적으로 이동함으로 둥둥 떠다니는 느낌을 줬습니다. 캐릭터와 충돌할 시 아이템은 사라지며 체력을 모두 회복합니다.



체력바

플레이어의 체력은 중앙 상단에 표시되어있고 일정 주기마다 감소됩니다. 이 주기는 플레이어가 앞으로 가면 갈수록 줄어듭니다. 따라서 플레이어는 더 자주 체력 아이템을 획득해야 할 것입니다.



미니맵

우측 상단에 플레이어의 머리 위에서 내려다보는 시점의 미니맵이 위치해 있습니다. 플레이어는 이 미니맵을 통해 숄더뷰로 얻을 수 없는 정보를 얻어서 장애물을 파훼할 수 있습니다.

실행 명령어 및 실행 방법 소개

openGL라이브러리 이외에 필요한 라이브러리는 없습니다. 압축 해제 후 exe파일을 실행하면 됩니다. WASD로 캐릭터 이동, 스페이스바로 점프, 마우스 드래그로 카메라 방향 변경, 마우스 휠로 캐릭터와 카메라 사이의 거리를 조절할 수 있습니다. 이 명령어는 실행했을 때 콘솔창에도 나오게됩니다.

개발 소감

2017182023 원윤식

3D와 관련된 프로그래밍은 처음이라 막상 다 만들고보니 특출난 것을 구현한 것도 아닌데 코딩하는데 꽤나 어려움을 겪은 것 같습니다. 가장 힘들었던 것은 충돌 판정이였는데 여길 고치면 저게 안되고, 저걸 고치면 이게 안되고 하는 상황이 있었습니다. 만족할만한 충돌 판정이 구현된 것은 아니지만 납득할만한 수준이라고 생각합니다. 그 외에는 프레임워크를 만들 때 팀프로젝트이다보니 class 구조를 이쁘고 깔끔하게 나누는 것에 집중했습니다. 특히 장애물과 구조물은 홍찬우 학우가 담당했기 때문에 쉽게 장애물을 생성할 수 있도록 구조를 구축해 놓았습니다. 각 소스의 종류마다 폴더로 분류하고, class마다 소스파일을 나눠서 관리했습니다. 이것은 저번학기 윈도우 프로그래밍 팀프로젝트 때 파일 분할을 안했다가 관리가 어려워진 경험이 있었기 때문에, 이번에는 제대로 분류해서 관리했습니다. 이번 openGL을 배우면서, 더 많은 3D 효과들을 배우고싶어졌고 정말 게임다운 게임을 만들고 싶다는 생각이 들었습니다.

2017180045 홍찬우

앞이 안보엿다. 한마디로 표현하면 이거였습니다. 실습도 어려워서 따라가지 못해서 물어물어 하드코딩을 해가며 하던 저에게 팀 프로젝트는 정말 힘든 산이였습니다. 그런데 제게 빛이 나타났습니다. 윤식… 그는 도대체… 먼저 프레임 워크를 다 구성하여 코딩을 못하던 저도 깔끔하게 알아볼 수 있게 해주었습니다. 그리고 혼자서 하는게 아닌 프로젝트답게 변수 이름 하나를 정할 때도 알아볼 수 있게 잘 해준 것을 보고 습관을 바꿔야겟다고 생각햇습니다. 하면서 중간중간 서로 피드백 하면서 자잘한 버그를 고치거나 새로운 것을 만들어 낸 것도 좋았고, 무엇보다 정말 좋앗던 것은 제가 생각한 난이도와 구성, 기획쪽을 전부 구현할 수 있었던 것입니다. 또 프로젝트를 하면서 여러가지를 배워간다는 것이 정말 좋았습니다. 기획을 하면서 느낀점은 역시 생각을 남에게 알려주는 것은 어렵다 생각햇고 그걸 해결하기위해 엑셀과 사진, 그림 ,글 등으로 해결 할 수 있다는 것을 알았던 것 같습니다.

유튜브링크(검색은 안되고 링크있는 사람만 접근 가능)

https://youtu.be/t35DvMpnZm4