컴퓨터 그래픽스 텀프로젝트

개발 결과 보고서



2013180021 원주연

2103180044 최진호

CHAPTER 1 게임 소개

<게임 시작 화면> <게임 종료 화면>

이번 텀 프로젝트 게임으로 여러 개의 미니게임으로 이루어진 ‘파티패닉’이란 게임에서 나오는 미니게임에 영감을 받아 제작하게 되었습니다.

저희 게임은 2인용이 가능하게 제작되었고 초기화면때 마우스로 1p와 2p를 선택할 수 있습니다.

한 사람 혹은 두 사람이 각자의 캐릭터를 5X5 블록으로 구성 되어있는 xz평면위에서 뛰어다닐 수 있습니다. 위에서 내려오는 보드판의 구멍을 찾아 이동하여 보드를 통과하는 게임입니다. xz축만을 이동하는 게임이라 3D 느낌이 나지 않아 고민하던 와중 보드가 회전하며 내려올 때 그에 맞게 그림자를 그려서 y축 느낌이 나도록 하였습니다. 두 축만 이동하지만 충분히 3D 느낌이 나는 게임입니다.

Chapter 2. 구조 소개

1. 클래스 및 구조체

typedef struct Pattern // 패턴을 읽어서 담는 구조체

{

int m\_pattern[5][5]; // 5x5 이차원배열로 뚫려있으면 1

}; // 막혀있으면 0

typedef struct Shadow //그림자 구조체

{

float m\_xpos = 0.f;

float m\_ypos = 2.f;

float m\_zpos = -400.f;

float max\_zpos; // 그림자의 z위치 한계선

// 보드의 맨위는 max\_zpos를 크게 맨 밑은 작게

float m\_velocity; // z축마다 속도를 다르게 함.

bool b\_real = false; // 비어있으면 그리지 않게

};

class Board //내려오는 보드 판 클래스

{

private:

float Xpos;

float Ypos;

float Zpos;

float Rotate;

bool b\_Ready = false; // 준비가 되면 맵에 그려짐

bool b\_Collision = false;

int Hole[5][5] = { 0 }; // 구멍을 뚫기위한 2차원 배열

int Hole\_Level = 0; // 패턴의 인덱스 번호(랜드값으로 부여받는다)

Pattern m\_Hole;

Shadow m\_shadow[5][5];

bool b\_DrawShadow = false;

public:

static Pattern pattern[18];

public:

void GetPos(float\*, float\*, float\*);

void SetPos(float, float, float);

float GetRotate();

void SetReady();

void SetState();

bool GetReady();

bool GetState();

void SetHole();

void GetPattern(Pattern\*);

void SetShadow();

bool GetShadow();

public:

void Update();

void Draw();

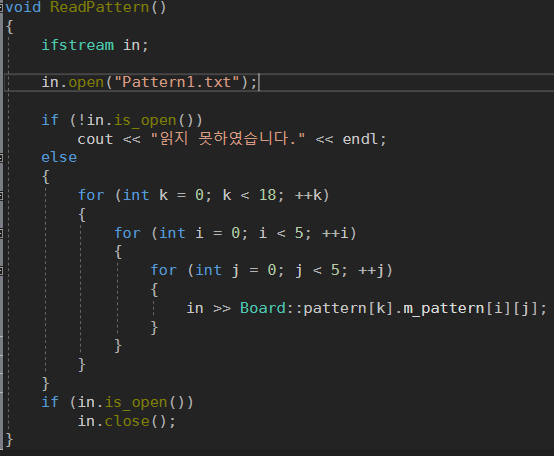
void ShadowInit();

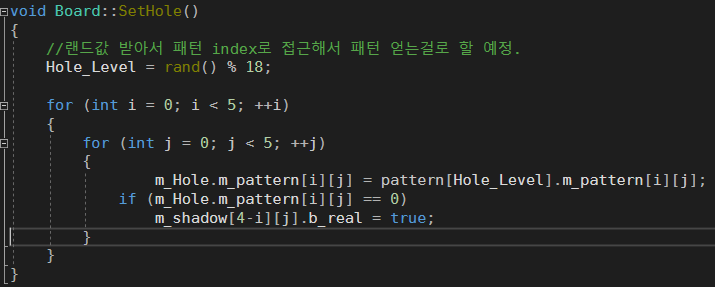
void Shadow\_Update();

void Shadow\_Draw();

};

chapter3 구조

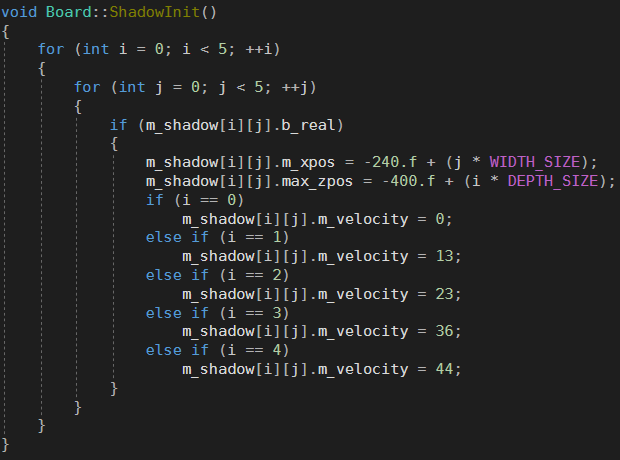
구조체 Pattern의 배열에 모든 패턴을 저장한다.



패턴을 보드의 멤버변수 Hole에 복사한다.

보드는 회전하여 떨어지면서 z축의 위 아래가 반전된다.

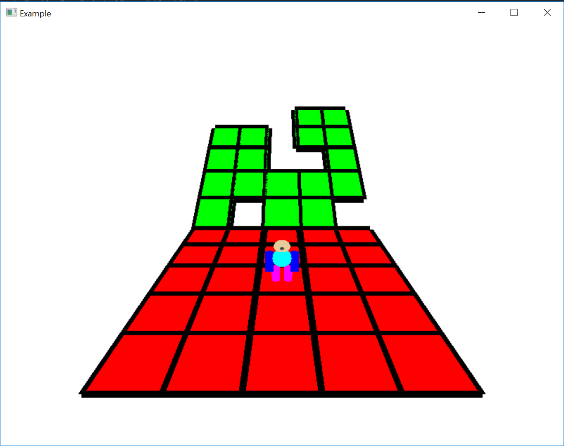
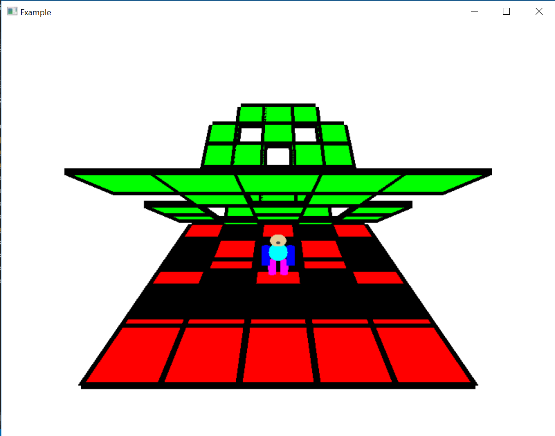
그래서 shadow엔 4-i번째 인덱스에 저장을 하였다.

그림자의 x축 z축 위치에 따라 속도와 좌표를 설정

Chapter 4 작업분담표

|  |  |
| --- | --- |
| **최진호** | **원주연** |
| 게임 프레임워크 제작 | 게임 맵 제작 |
| 플레이어 클래스 제작 | 보드 제작 |
| 텍스쳐 구현 | 그림자 구현 |
| 1p, 2p 구현 | 게임 사운드 구현 |
| 플레이어끼리 충돌체크 구현 | 플레이어 보드판 충돌체크 구현 |
| 버그 수정 및 게임 마무리  개발보고서, 발표자료 제작 | |

Chapter 5 결과물 분석

< 보드 대기 상태> < 보드가 내려올 때>

우리의 게임의 가장 큰 특징인 그림자를 찍은 스크린샷입니다. 보드의 모양, 각도에 따라 그림자를 Animate하였습니다. 캐릭터는 8방향으로 이동이 가능하며 2인플레이일 경우 먼저 구멍을 차지한 플레이어를 보드가 덮치기 직전 밀어내서 상대방을 죽일 수도 있는 경쟁구도를 만들어보았습니다.

Chapter 6 명령어 소개

1. 마우스 명령어

게임시작 시 마우스로 1p 2p 둘 중 하나 선택하여 클릭

1. 키보드 명령어

1p 이동 : w a s d

2p 이동 : I j k l

Chapter 6 프로젝트 개발 소감 및 후기

원주연 [2013180021]

그림자를 구현하는데 있어서 많은 버그가 발생하였고 모양이 대칭인 패턴으로만 테스트를 해서 버그를 늦게 발견하였다. 이 버그를 픽스하느라 부가적인 요소들을 추가하지 못하였다. 앞으로 있는 테스트에 있어서 꼼꼼히 테스트를 하여 버그를 빠르게 발견 이 버그를 픽스하고 일정에 차질이 없게 할 것이다.

개발도 무작정 시작하다보니 텍스쳐를 입히기 힘들었다. SceneManager를 활용하여 만들었다면 텍스쳐를 입힐 수 있었을 것이고, 스테이지도 구현할 수 있었을 것이다. 기본 전체적인 틀을 짜고 개발을 시작하는 습관을 들여야겠다는 생각을 하였다.

최진호[2013180041] :

이번 프로젝트를 시작 할 당시에 저번학기에 수강한 윈도우 프로그래밍 프로젝트를 망치게 된 이유를 생각해 보았다. 우선 주기적으로 프로젝트를 진행하지 않은 것과 많은 시간을 투자 못한 것이 실패의 원인이였다. 그래서 이번 컴퓨터 그래픽스 프로젝트에는 주기적으로 진행하고 시간도 많이 들였다. 윈플 프로젝트 보다 결과물이 좋아서 만족하고 프로젝트를 진행하면서 전반적인 내용을 복습한거 같아서 기억에 많이 남는다.

또한 협업의 기회를 통해 의사소통의 방법과 중요함을 깨달았다. 성장하는 모습을 조금이라도 보여서 스스로 만족한다.