**Computer Graphics Programming Assignment #2**

20170869 컴퓨터공학과 이정훈 [ljh453@postech.ac.kr](mailto:ljh453@postech.ac.kr)

20170185 컴퓨터공학과 최석환 [seokhwan98@postech.ac.kr](mailto:seokhwan98@postech.ac.kr)

1. **프로그램 개요**

배경이 움직이면서 점프로 장애물을 피하고 별을 먹어 점수를 획득하는 게임. 플레이어는 점프를 해서 날아오는 FireBall과 구멍을 피해야 한다. FireBall에 맞거나 구멍에 빠질 경우 게임이 종료되며 게임오버하기전까지 최대한 많은 별을 먹어 높은 점수를 얻는 것을 목표로 한다. 지형은 여러 높이로 구성되며 낮은 지형에 대해서는 점프 없이도 진행할 수 있다.

1. **프로그래밍 환경**

개발환경:Visual Studio & GitHub

버전 정보 : freeglut-MSVC-3.0.0-2

glew-2.1.0-win32

glm-0.9.9.7

1. **프로그램 설계 및 구현**

3.1. 자료구조

|  |
| --- |
| class Object |
| private:  float x;  float y;  float width;  float height;  public:  virtual bool isCollided(Object object);  virtual bool isCollided(Object object, float margin); |

virtual bool isCollided(Object object) : 매개변수 object와 충돌 여부 반환

virtual bool isCollided(Object object, float margin) : 매개변수 object와 충돌 여부 반환, margin : 충돌 범위 추가 넓힐 때 사용

|  |
| --- |
| class MovingObject |
| private:  float x\_viewport;  public: |

float x\_viewport : viewportX 기준 x 축 방향으로 어느 위치에 있는지

|  |
| --- |
| class Terrain |
| private:  bool isHole;  public: |

|  |
| --- |
| class Star |
| private:  bool collected;  float outerRadius;  float innerRadius;  public:  void enableStar();  void disableStar();  bool isCollected(); |

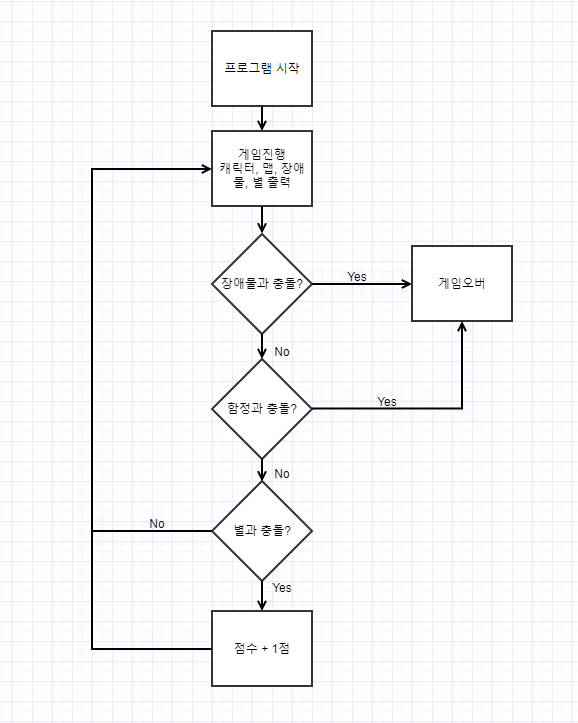
void enableStar() & void disableStar() : 별을 먹으면 안보이게 만들어주고 점수를 추가하기 위해 collected를 수정하기 위한 함수

bool isCollected() : 별을 먹고 수정된 collected를 알려주기 위한 함수

|  |
| --- |
| class Fireball |
| private:  float speed;  float radius;  public: |

|  |
| --- |
| class Character |
| private:  float jumpSpeed;  JumpState characterJumpState;  public:  bool isCollided(Object object) override;  bool isCollided(Object object, float margin) override;  bool isOnGround(std::list<Terrain> terrainList\_);  float surfaceY(std::list<Terrain> terrainList\_); |

3.2 플로우차트



3.3 애니메이션

애니메이션은 총 4프레임으로 구성되어 있다.



1. **프로그램 실행방법**

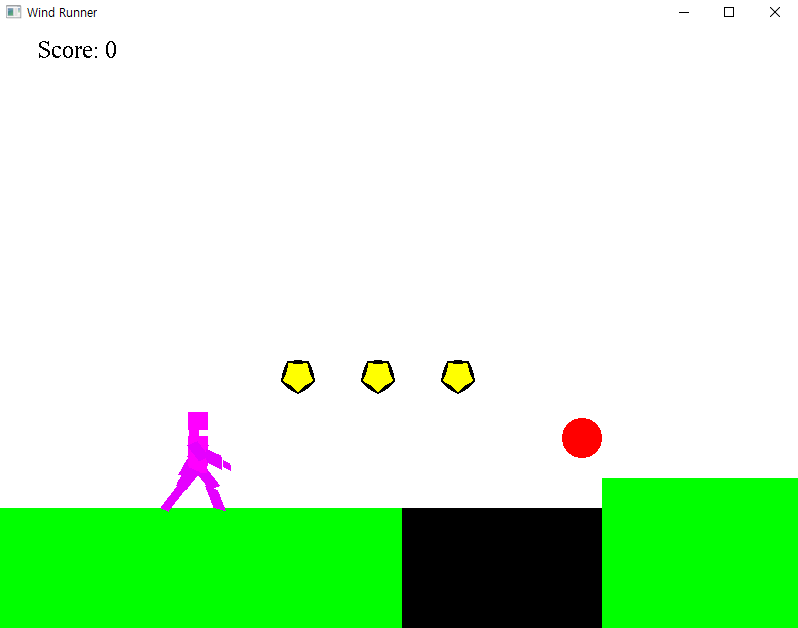
Visual Studio를 이용하여 프로젝트 파일을 실행한다.

프로그램을 컴파일 할 경우 게임이 바로 시작된다.

게임이 시작되면 플레이어는 키보드의 스페이스를 이용하여 캐릭터를 조작할 수 있다.

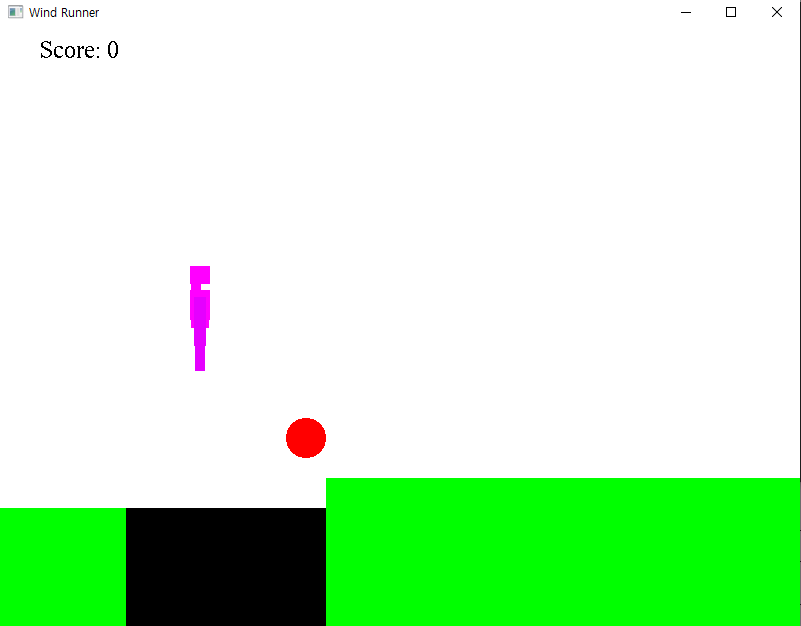
스페이스로 점프를 해서 장애물과 함정을 피해 높은 점수를 획득하면 된다.

1. **예제**

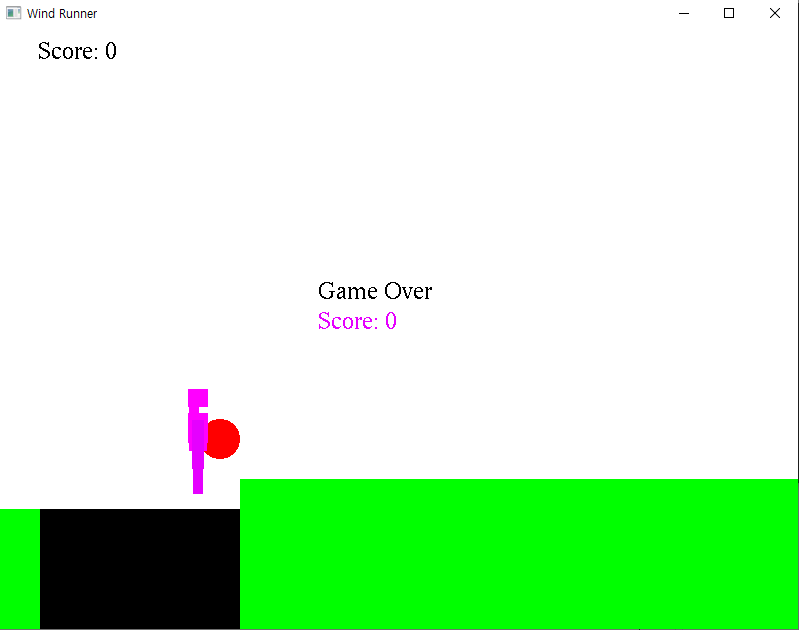


게임을 시작하면 다음과 같은 화면이 나온다.

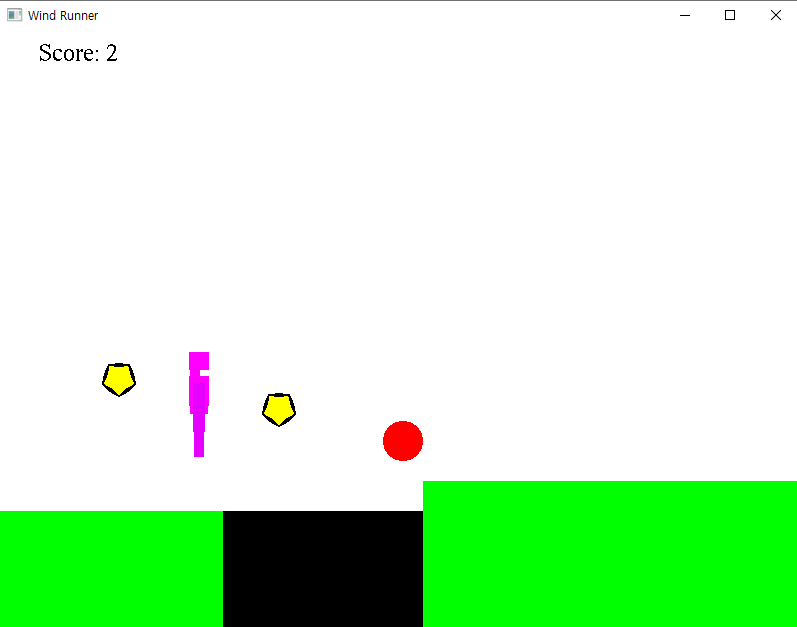
화면에는 플레이어(보라색 상자), 파이어볼(빨간색 타원), 별(노란색 별), 구멍(검은 사각형), 맵(초록색 사각형)이 배치되어 플레이어 방향으로 배경이 이동한다.



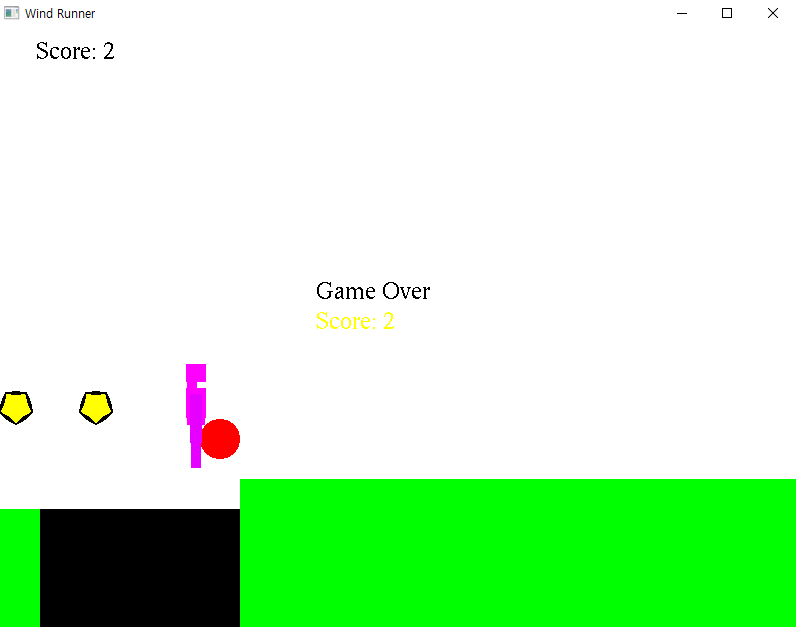
키보드로 조종을 하며 스페이스를 누르면 플레이어가 점프를 할 수 있다. 점프 중, 스페이스를 한 번 더 누르면 더블점프를 할 수 있다.



만약에 파이어볼, 높은 벽에 접촉하게 되거나 구멍에 빠지면 게임오버가 뜨고 점수가 출력이 되며 게임이 종료된다.



플레이어가 별과 접촉하게 되면 점수가 1점 오르고 별은 사라지게 된다.



스페이스를 이용하여 점프를 하고 장애물과 함정을 피해 최대한 별을 많이 먹으면 되는 게임이다.

1. **토론**

현재 애니메이션은 frame이라는 변수를 2개를 쓰는데 frame2라는 변수는 프로젝트가 끝날 때까지 무한히 증가하는 형태이다. 이럴 경우 프로그램을 길게 유지할 수 없다. 프레임이 지나가는 속도를 조절하여 프레임을 조절하면 더 메모리를 적게 쓸 수 있을 것 같다.

1. **결론**

이 프로젝트를 통해 2D 애니메이션을 어느정도 구현할 수 있게 되었다. 추가적으로 지난번 ASSN 1을 진행하면서 부족했던 캐릭터 충돌에 범위와 판정에 대해 더 자세하게 알 수 있었다.

1. **개선방향**

현재 프로그램에서는 버섯과 낮은 점프가 구현되어 있지 않다. 지난번 프로젝트를 완성하고 제시했던 개선방향을 개선하여 프로젝트가 좀 더 직관적으로 돌아가게 만들었다. 그렇게 캐릭터의 정면 충돌에 대해서는 충돌이 잘 이루어지게 되었으나 위아래 충돌이 직관적이지 않아 버섯이 잘 작동하지 않았다. 충돌에 대한 알고리즘을 다시 살펴볼 필요가 있다.