**REPORT**



|  |  |
| --- | --- |
| **과목명** | 객체지향프로그래밍2 |
| **학과** | 컴퓨터공학과 |
| **학번** | 12171661 |
| **이름** | 윤혁 |
| **제출일자** | 2020-11-07 |

목차

**설계 방법1**

**구현 과정2**

**결과 화면3**

**1. 설계방법**

1) 창 : 게임을 실행하면 사용자가 play를 할 수 있는 화면

- 창의 크기, 창의 이름, 프레임 등

2) 게임의 구성요소 : 벽돌깨기 게임을 하는데 창 안에서 구현해야 할 구성요소

- 벽돌, 공, 바(paddle) 및 이것들을 움직이거나 상호작용해서 게임을 진행하는 함수들.

3) 게임의 부가요소 : 점수, 난이도, 플레이 시간

- 벽돌을 깰 때마다 추가되는 점수를 실시간으로 표시, 플레이 시간을 실시간으로 표시하여 자신의 기록을 알 수 있음. 1단계를 clear하면 자동으로 2단계가 실행하여 조금 더 어려운 난이도의 벽돌 깨기를 play.

**2. 구현과정**

1)Paddle Class

paddle 클래스는 벽돌깨기를 할 때 공을 튕기는 역할을 하는 일종의 ‘바’입니다. paddle은 RectangleShapeclass 를 상속받았으며 매개변수가 2개의 float인 생성자와 update함수로 이루어져 있습니다. 생성자에서는 paddle의 기본 위치 x, y좌표를 인스턴스로 받아서 size는 {80, 20}, 위치는 인스턴스로 받은 위치 {x, y}, 색깔은 RGB color (0, 255, 64), 엔티티에 origin은 (0, 0)으로 설정합니다. update함수에서는 paddle을 움직일 때 y축 값은 고정으로 해 두고 좌우로만 움직일 수 있게끔 x 값을 할당합니다.

2)Ball Class

ball 클래스는 벽돌을 깨는 ‘공’의 객체입니다. 이 클래스는 Circle Class를 상속받았으며 생성자와 update, isIntersecting함수, handleCollision 함수로 이루어져 있는데 공의 속도는 전역변수로 선언된 5.0으로 설정이 되고 position은 매개변수로 받은 (x, y)값, 엔터티의 origin은 (0, 0), 색상은 RGB color(255, 128, 0)으로 설정합니다. update함수에서는 move함수를 통해 공의 속도만큼 x, y축으로 이동하게끔 설정해주고 그 공들이 벽에 부딪힐 때마다 튕겨 나오도록 if, else if문을 통해 window 크기를 벗어나지 않게 설정해줍니다. bool isIntersecting(Paddle& paddle)함수는 공이 paddle과 부딪힘을 감지하면 true를 반환하는 함수입니다. void handleCollision(Paddle& paddle)은 만약 paddle에 부딪히면 튕겨 나가게 해주는 함수입니다.

3)Brick Class

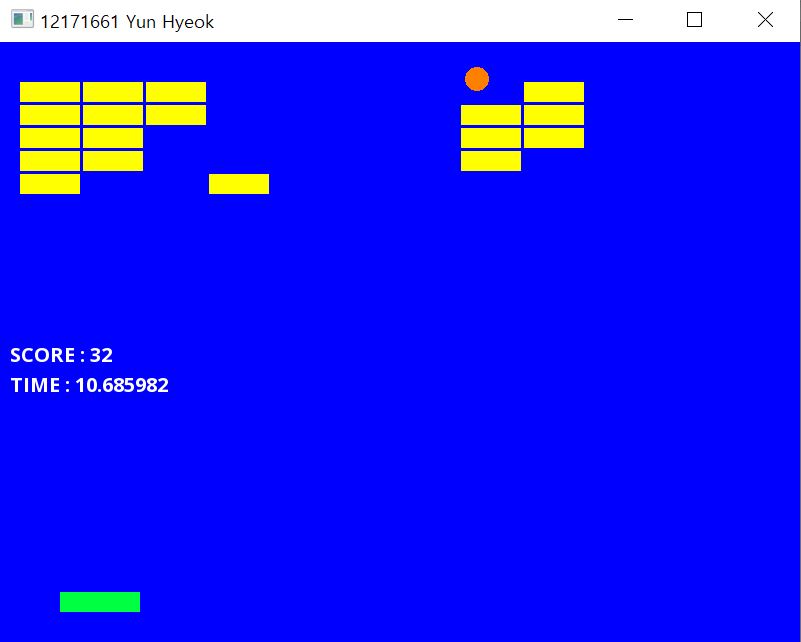
brick 클래스는 사용자의 공에 의해 깨지는 ‘벽돌’의 객체입니다. 이 클래스 또한 사각형 모양이기 때문에 RectangleShape Class를 상속받고 있으며 생성자, 그리고 Ball Class에서와 같이 isIntersecting함수, handleCollision함수가 있습니다. 생성자에서는 크기를(60.0, 20.0)으로 기본 설정해주고 있으며 position은 매개변수로 받은 (x, y)값을 설정합니다. 색상은 Yellow색으로 설정하고 엔터티의 origin은 (0, 0)으로 설정합니다. bool isIntersecting(Ball& ball)함수 에서는 벽돌이 ball와 부딪히게 되면 true를 반환하는 함수입니다. void handleCollision(Ball& ball)함수는 공이 벽돌과 부딪히면 일어나는 일을 구성한 함수인데, 부딪히면 Brick 클래스의 멤버변수인 bool deleted= false;로 초기화 되어있는데 true로 바꿔줍니다.

4) main 함수

위에 구성한 class들을 활용해 게임을 실행합니다. ball과 paddle을 클래스 생성자 함수를 통해 800x600의 화면을 벗어나지 않도록 생성해 줍니다. 벽돌은 하나가 아니기 때문에 vector를 이용해 생성하고 이중 for문을 통하여 10\*5개의 벽돌들을 설정한 크기로 생성해줍니다. 게임의 부가요소를 구현하기 위해 Font, Text, Clock 객체를 생성해줍니다. Text의 font는 "OpenSans-Bold.ttf"로 설정합니다. RenderWindow객체로 800x600크기의 화면을 “12171661 Yun Hyeok”이름으로 생성해줍니다. 최대 프레임은 60으로 설정합니다. 화면이 open되면 배경은 Blue색상으로, 화면 좌측에는 text를 통해 벽돌을 깰 때마다 1점씩 올라가는 score와 플레이한 시간을 표시합니다. 당연히 시작점수는 0점입니다. 시간은 time객체와 clock객체를 사용해서 실시간 플레이 시간을 표시해줍니다. paddle객체는 사용자가 마우스를 움직인대로 이동하게 하되 x축으로만, 즉 좌우로만 이동할 수 있게 합니다. 그리고 ball도 움직임을 설정하는 update함수와 부딪힐 때 일어나는 일을 구현하는 handleCollision함수를 넣어줍니다. for문에 vector의 배열의 크기를 자동으로 인식하는 루프를 통해 벽돌을 깨는 동작을 구현합니다. 또한 벽돌이 깨질 때마다 vector 배열에서 erase합니다. 그리고 여기서 사용자의 점수를 올리는 부분이 있는데, bricks vector의 size가 기존의 size에서 1이 줄었다면, 즉 사용자가 벽돌을 1개 부실 때마다 score++를 해주고 이를 string status에 반영하여 text에 표시하도록 합니다. 만약 brick.size()가 0이 된다면, 즉 모든 벽돌이 부셔졌다면 “Level 1 success! This tis Level 2”라는 메시지를 띄우며 바로 2단계로 넘어갑니다. 2단계는 공의 속도를 7.5로 재설정해주고 score도 초기화 해줍니다. 그 이후는 기존의 main 함수와 동일한데 paddle의 크기로 줄여주므로 paddle.setSize({40,20}); 을 추가합니다. 또 다른 점은 1단계에서는 벽돌을 모두 부시면 2단계로 넘어갔지만 이번에는 “You have cleared Level 1 and 2! Great!!!@@” 라는 문구만 표시합니다. 그리고 2번 반복되는 main함수 모두 window.draw 함수를 통해 ball, paddle, brick, text를 화면에 그려줍니다. 마지막으로 window.display() 함수로 화면에 모두 display해줍니다.

**3. 결과화면**

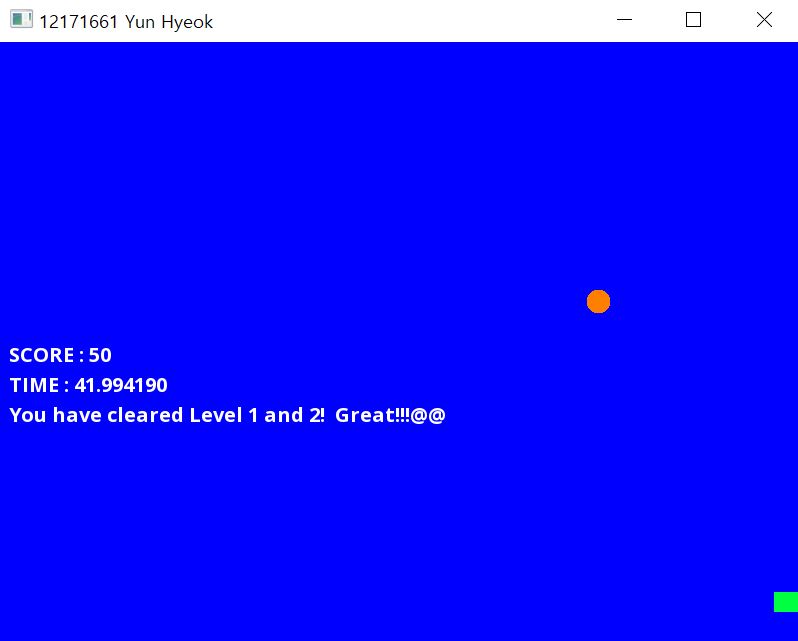
1) 벽돌깨기를 첫 실행 시 나오는 화면입니다. 벽돌 1개를 깰 때 마다 1점이 올라가며 실시간 플레이 타임도 흘러갑니다. 1단계 입니다.

****

2) 2단계로 넘어간 화면입니다. 공의 속도가 5.0->7.5로 빨라졌으며 바의 크기도 절반으로 줄은 모습입니다.

****

3) 1, 2단계 모두 클리어한 상태입니다. 더 이상의 단계는 없으며 축하메세지가 표시됩니다.

****