

CSED 232 Object-Oriented Programing (Spring 2021)

Programming Assignment # 5

- GUI -

Due date : 6월 11일

담당 조교 : 김성태 (seongtae0205@postech.ac.kr)

주의사항

- 각 클래스마다 header file과 cpp file을 나누어 구현하여야 합니다.
- 문제에 명시되어 있지 않더라도 각 클래스마다 생성자(Constructor), 소멸자(Destructor)는 필수입니다. 기본 생성자의 경우, 별다른 언급이 없다면 멤버 변수에 int, double은 0, string은 "", pointer는 NULL로 초기화해 주십시오.
- 클래스마다 클래스명과 파일명을 통일하여 주십시오.

감점

- 제출 기한이 지나면 얻은 총점의 20% 감점
- 추가로 하루(24시간) 늦을 때마다 20%씩 감점
1일 이내 지연: 20% 감점, 2일 이내 지연: 40% 감점, 5일 이상 지연: 0점
- 컴파일이 정상적으로 이루어지지 않을 경우 0점

제출방식

채점은 Qt Creator 4.15환경에서 이루어집니다. 작업하신 환경이 있는 프로젝트 폴더에 Report를 포함하여 zip파일로 압축 후 제출해 주시기 바랍니다. 프로젝트 폴더 안에는 *.pro 파일이 꼭 들어가야합니다. 폴더명은 '#학번'으로 만들어 주시고, Report는 pdf 형식으로 제출해주세요. 반드시 PLMS를 통해 제출해주시기 바랍니다. 이메일 제출은 인정되지 않습니다. 5일이 지날 경우 0점이므로 과제 제출 마감일로부터 5일후인 6월 16일 23시 59분 59초 이후는 PLMS를 통해 제출하실 수 없습니다.

예) 20209999.zip

채점 기준

1. 프로그램 기능

- 프로그램이 요구 사항을 모두 만족하면서 올바르게 실행되는가?

2. 프로그램 설계 및 구현

- 요구 사항을 만족하기 위한 변수 및 알고리즘 설계가 잘 되었는가?
- 설계된 내용이 요구된 언어를 이용하여 적절히 구현되었는가?

3. 프로그램 가독성

- 프로그램이 읽기 쉽고 이해하기 쉽게 작성되었는가?
- 변수 명이 무엇을 의미하는지 이해하기 쉬운가?
- 프로그램의 소스 코드를 이해하기 쉽도록 주석을 잘 붙였는가?

4. 보고서 구성 및 내용, 양식

- 보고서는 적절한 내용으로 이해하기 쉽고 보기 좋게 잘 작성되었는가?
- 보고서의 양식을 잘 따랐는가?

다른 사람의 프로그램을 복사(copy)하거나 간단히 수정해서 제출하면 학점은 무조건 'F'가 됩니다.
이러한 부정행위가 발견되면 학과에서 정한 기준에 따라 추가의 불이익이 있을 수 있습니다.

F 피하기

본 과제에서는 'F 피하기' 게임을 구현해보면서 Qt GUI programming을 익히도록 한다. 본 과제를 수행하기 전에 아래 링크로 들어가 6강까지 필히 숙지하기를 권장한다. 강의를 시청하지 않을 경우 과제 진행에 큰 어려움이 있을 수 있다.

<https://youtube.com/playlist?list=PLMgDVla0Pg8Wrl9WmZR09xAbfXyfkqKWY>

1. 게임설명

1.1 시작화면

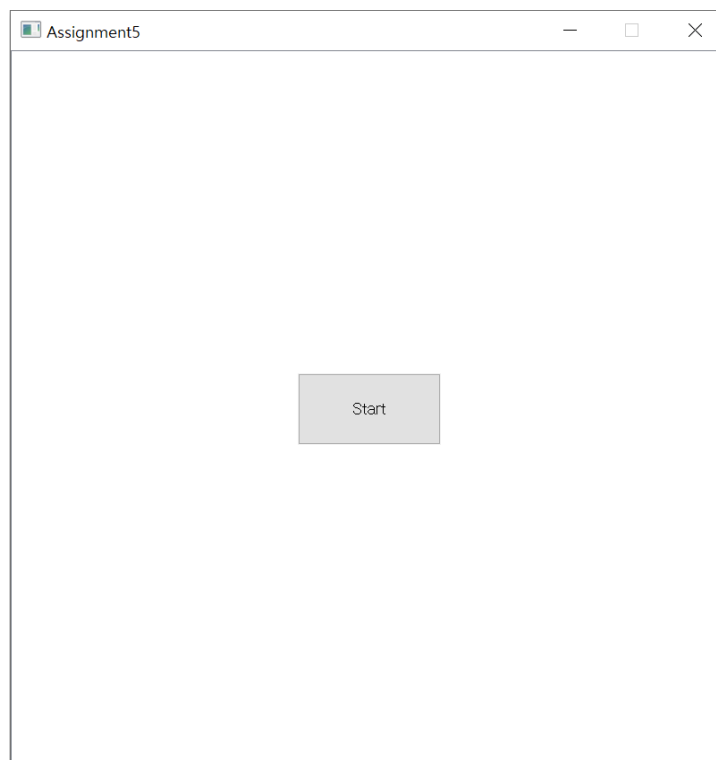


그림 1. 시작화면

프로그램이 시작되면 그림1과 같은 시작화면을 보여준다. 가운데 'Start' 버튼을 클릭하면 본 게임이 시작된다.

1.2 게임

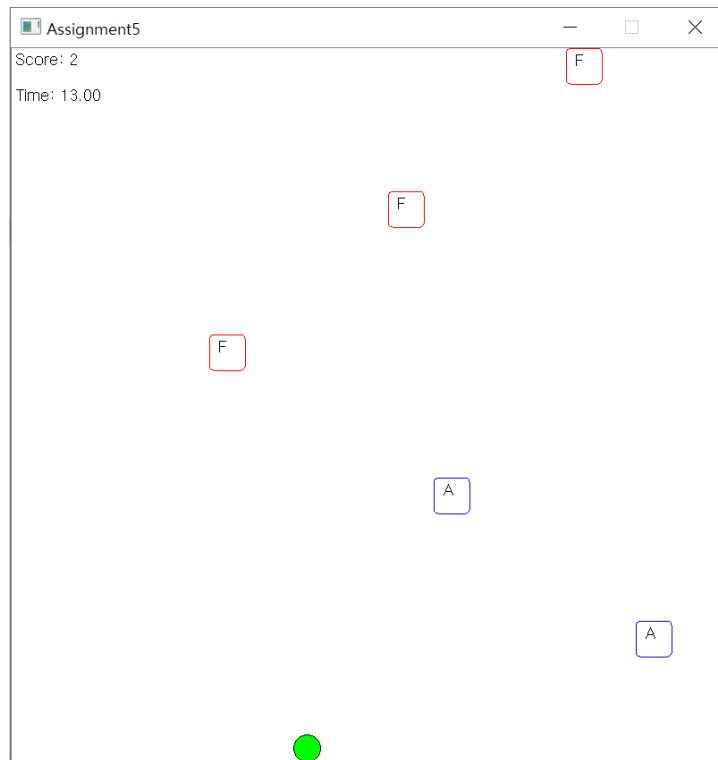


그림 2. 게임 진행 예시

본 게임의 목적은 제한시간 내에 Player가 F를 피하면서 A를 최대한 많이 획득하는 것이다. A를 획득하면 Score가 증가하며, F를 획득하면 게임이 종료된다.

Grade

Grade에는 F와 A가 있다.

F



그림 3. F 예시

F는 Player가 피해야 할 대상이다. Player가 F와 충돌하면 게임이 종료된다.

A



그림 4. A 예시

A는 Player가 획득해야 할 대상이다. Player와 A가 충돌하면 Score가 1 증가한다.

Player



그림 5. Player 예시

Player는 사용자가 직접 조종할 수 있다. 이동은 좌우로만 가능하다.

1.3 게임종료

게임이 종료되는 상황은 GameOver와 TimeOver 두 가지이다. 게임종료 상황이 발생하면 해당 상황에 맞는 문구를 가진 MessageBox를 띄운다. Message의 Title은 'End'로 설정한다. MessageBox의 'OK' 버튼을 클릭하면 프로그램이 종료된다.

GameOver

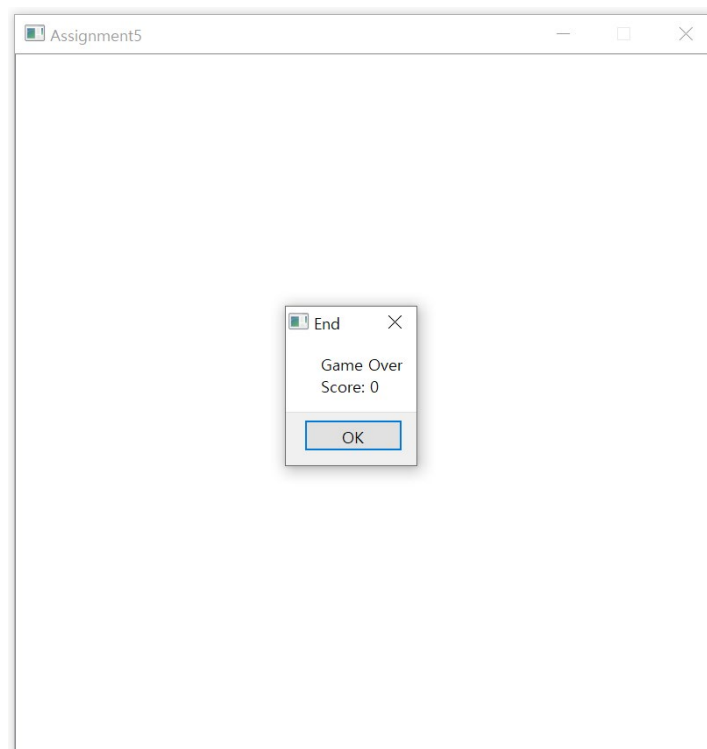


그림 6. GameOver

Player와 F가 충돌할 시 발생한다. 'Game Over' 문구와 현재까지 획득한 Score를 같이 MessageBox에 출력한다. 'OK'버튼을 누르면 프로그램이 종료된다.

TimeOver

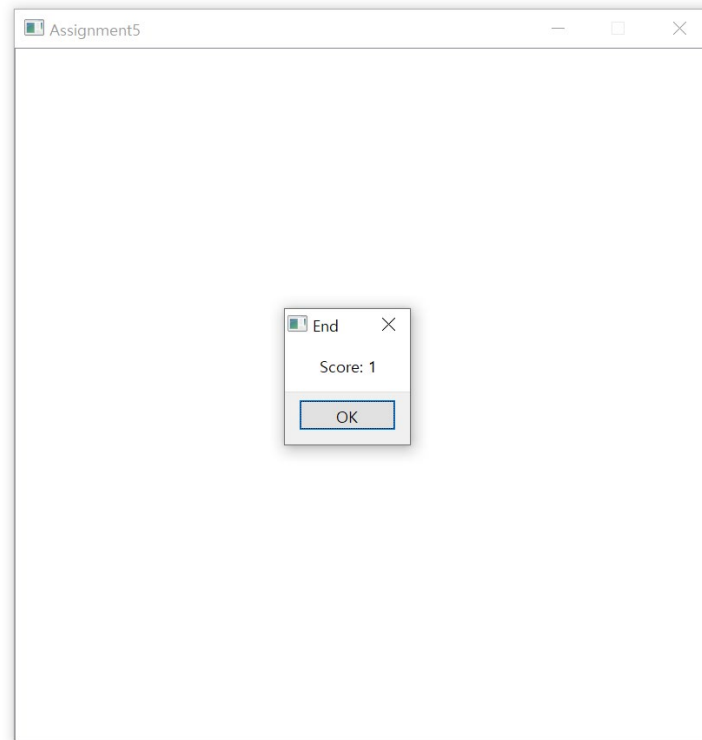


그림 7. TimeOver

제한시간이 지나면 발생한다. 현재까지 획득한 Score를 같이 MessageBox에 출력한다. OK버튼을 누르면 프로그램이 종료된다.

2. 게임구현

언급되지 않은 구현 및 설정에 대해서는 자유로운 구현이 가능하다. Mainwindow.ui를 통한 ui 구현을 허용한다. 첨부된 링크를 통해 구현에 필요한 class를 확인하길 바란다. 첨부된 class 이외의 다른 class사용이 필요할 수 있다. 구현에 필요한 자세한 함수 사용법들은 Google 검색을 최대한 활용하도록 한다.

2.1 시작 화면

시작 화면은 800x800으로 설정한다. 'Start' 버튼은 화면 중앙에 정렬되어 위치하도록 한다. 시작 화면 Title은 'Assignment5'로 설정한다.

2.2 게임

게임화면

게임 화면은 800x800으로 고정한다. Scene의 크기와 위치를 View에 맞춘다.

<https://doc.qt.io/qt-5/qgraphicsview.html>

<https://doc.qt.io/qt-5/qgraphicsscene.html>

F, A

자유로운 모양으로 구현하되, 크기는 40x40으로 고정한다. 원으로 구현할 경우에는 지름을 40으로 설정한다. F와 A는 서로 다른 색상을 가져야 하며, 안에는 Grade를 표현하는 Text가 표현되어야 한다. QGraphicsItem의 하위 class (QGraphicsRectItem 등)를 사용하여 구현하도록 한다.

<https://doc.qt.io/qt-5/qgraphicsitem.html>

<https://doc.qt.io/qt-5/qtimer.html>

<https://doc.qt.io/qt-5/qpainter.html>

0.5초(=500ms)마다 F 또는 A가 랜덤으로 생성된다. Grade가 생성되었을 때 x축 위치는 Scene 내부에서 랜덤으로 설정되며, y축 위치는 0으로 설정된다. x축 생성범위는 Grade의 오른쪽(Right)이 Scene의 끝과 일치하도록 설정한다. Grade는 0.2초(=20ms)마다 y축으로 10만큼 이동한다.

Class	생성 timeStamp	초기 위치	이동 timeStamp	이동 단위
F, A	0.5s(=500ms)	(rand()%(SCENE_WIDTH – object.rect().width()), 0)	0.2s(=20ms)	(0, 10)

표 1. Grade 설정

Player

원으로 구현하며 지름은 30으로 고정한다. 원 내부 색상은 자유롭게 선택한다. 시작 x축 위치는 Scene의 가운데로 위치시킨다. 시작 y축 위치는 Player의 아래(Bottom)이 Scene의 끝과 일치하도록 설정한다.

<https://doc.qt.io/qt-5/qgraphicsellipseitem.html>

<https://doc.qt.io/qt-5/qtimer.html>

키보드 왼쪽 방향키를 입력하면 왼쪽으로 10만큼 이동하며, 키보드 오른쪽 방향키를 누르면 오른쪽으로 10만큼 이동한다. 이동시 Player가 Scene을 벗어나면 안 된다.

Class	초기 위치	이동 단위	x축 범위
Player	(SCENE_WIDTH/2-object.rect().width()/2, SCENE_HEIGHT-object.rect().height())	(-10,0) 또는 (10, 0)	[0, SCENE_WIDTH-object.rect().width()]

표 2. Player 설정

충돌

Player와 Grade가 충돌하면, Grade에 따라 다른 함수가 호출된다. F와 충돌하면 GameOver가 발생하며, A와 충돌하면 Score가 1 증가한다. Grade는 y축 위치가 Scene을 벗어나거나, Player와 충돌하면 삭제된다.

Score

시작 시 0으로 설정된다. Score는 화면 왼쪽 상단에 위치한다.

<https://doc.qt.io/qt-5/qgraphicstextitem.html>

Time

Score 아래에 표시된다. 게임 진행시간은 30초로 고정된다. 남은 시간을 소수점 둘째자리까지 시각화하여 보여준다.

<https://doc.qt.io/qt-5/qgraphicstextitem.html>

<https://doc.qt.io/qt-5/qtimer.html>

2.3 게임종료

게임종료 상황이 발생하면, Scene을 clear한다(= Scene 안에 있는 모든 Item들을 삭제시키고 Grade 생성을 중지한다). Scene을 clear한 후 각 상황에 따른 MessageBox를 띄운다.

GameOver

Player와 F가 충돌하면 발생한다. GameOver가 발생하면 'End' title을 가진 MessageBox를 띄운다. MessageBox에는 'Game Over' 문구와 현재까지 획득한 Score를 같이 출력한다. 'OK'를 클릭하면 프로그램 전체를 종료한다.

<https://doc.qt.io/qt-5/qmessagebox.html>

TimeOver

남은 게임시간이 0이 되면 발생한다. TimeOver가 발생하면 'End' title을 가진 MessageBox를 띄운다. MessageBox에는 현재까지 획득한 Score를 출력한다. 'OK'를 클릭하면 프로그램 전체를 종료한다.

<https://doc.qt.io/qt-5/qmessagebox.html>

채점 기준

1. 프로그램 기능 – 70%
 - A. 시작화면 - 5
 - B. 게임
 - i. Player구현 - 15
 - ii. F 구현 - 15
 - iii. A 구현 - 15
 - iv. Score 구현 - 5
 - v. Time 구현 - 5
 - C. 게임 종료
 - i. GameOver 구현 - 5
 - ii. TimeOver 구현 - 5
2. 프로그램 설계 및 구현 – 25%
3. 프로그램 가독성 – 5%
4. 보고서 구성 및 내용, 양식 – 10%