

# 오픈소스SW 과제중심수업 보고서

문화인류학과  
2014044120 권기홍

GitHub repository 주소 : <https://github.com/gitul0515/osw>

## 1. 각 함수들의 역할

1) main 함수: 프로그램 실행을 위한 시작 지점 역할을 한다. 전역 변수들을 선언 및 초기화하고, pygame 모듈을 초기화한다. 현재 창의 캡션을 설정하고, 스크린에 텍스트를 표시한다. game loop 안에서는 배경음악을 랜덤으로 불러들여 재생하고, runGame 함수를 호출한다. runGame 함수가 종료되면 음악을 종료하고 스크린에 텍스트를 표시한다.

2) runGame 함수: 게임을 실행하는 함수이다. game loop 안에는 게임의 주요 규칙들이 정의되어 있다. 이를 간략히 정리하면 다음과 같다.

- fallingPiece가 없으면 piece를 새롭게 생성하기
- 보드가 가득찼을 경우 게임을 종료하기
- KEYUP 이벤트로 p가 눌렸을 때 게임을 일시 정지하기
- K\_LEFT 또는 K\_a가 눌렸을 때 piece를 왼쪽으로 이동하기(이동할 공간이 있을 때)
- K\_RIGHT 또는 K\_d가 눌렸을 때 piece를 오른쪽으로 이동하기(이동할 공간이 있을 때)
- K\_UP 또는 K\_w가 눌렸을 때 piece를 회전하기(회전할 공간이 있을 때)
- K\_q가 눌렸을 때 다른 방향으로 회전하기(회전할 공간이 있을 때)
- K\_DOWN 또는 K\_s가 눌렸을 때 piece를 빠르게 밑으로 내리기
- K\_SPACE가 눌렸을 때 piece를 완전히 밑으로 내리기
- 키를 누르고 있는 채로 piece를 움직이기
- 시간이 흐름에 따라 piece를 서서히 떨어지게 하기
- 현재 게임의 상태를 스크린에 모두 그리기

3) makeTextObjs 함수: 텍스트를 만드는 함수이다. render 함수를 통해 텍스트가 표시된 Surface를 만든 후, Surface와 Rect 객체를 반환한다.

4) terminate 함수: 게임 프로그램을 종료시키는 함수이다.

5) checkForKeyPress 함수: KEYUP 이벤트가 발생하였는지 확인하는 함수이다. KEYDOWN 이벤트는 무시하고, KEYUP 이벤트가 있을 경우 이를 반환한다.

6) showTextScreen 함수: 사용자가 키를 누를 때까지 화면 중간에 큰 텍스트를 그려서 보여주는 함수이다. 키를 누를 때까지 화면을 계속 업데이트한다.

7) checkForQuit 함수: 게임 종료의 조건을 확인하는 함수이다. QUIT 이벤트가 발생하였는지 혹은 Esc key가 눌렸는지 확인하여 terminate() 함수를 호출한다.

8) calculateLevelAndFallFreq 함수: score에 따라 플레이어의 level을 계산해서 반환하는 함수이다. 또한, piece가 떨어져야 하는 속도를 계산해서 반환한다.

9) getNewPiece 함수: 새로운 piece를 무작위로 만드는 함수이다. 무작위로 회전시키고 무작위로 색깔을 설정하여 새로운 piece를 반환한다.

10) addToBoard 함수: piece의 위치, 모양, 회전 상태에 따라 보드를 채우는 함수이다.

11) getBlankBoard 함수: 비어 있는 보드 데이터 구조를 새롭게 생성하여 반환하는 함수이다.

12) isOnBoard 함수: 해당하는 x,y 좌표가 보드 위에 있는지 확인하는 함수이다.

13) isValidPosition 함수: piece가 유효한 자리에 있는지 확인하는 함수이다. piece가 보드 위에 있고 (다른 상자와) 충돌하지 않았다면 True를 반환한다.

14) isCompleteLine 함수: 보드에서 줄이 완성되었는지 확인하는 함수이다. 상자들로 채워진 줄에 빈 공간이 없다면 True를 반환한다.

15) removeCompleteLines 함수: 보드에서 완성된 줄을 제거하는 함수이다. 위에 있는 모든 상자를 밑으로 한 줄 내리고 맨 위의 줄은 빈칸으로 설정한다. 밑으로 내린 줄도 완성된 줄이라면 똑같이 제거한다. 제거한(완성된) 줄의 개수를 반환한다.

16) convertToPixelCoords 함수: 주어진 보드 좌표를 픽셀 좌표로 변환하는 함수이다.

17) drawBox 함수: 상자 하나를 보드 좌표에 그리는 함수이다. 만약 픽셀 좌표가 주어진다면, 상자를 해당 픽셀 좌표에 그린다.

18) drawBoard 함수: 스크린에 보드를 그리는 함수이다. 보드 주변의 테두리를 그리고, 보드의 배경을 채우며, 보드의 각 상자를 그린다.

19) drawStatus 함수: 현재 score와 level의 텍스트를 그리는 함수이다.

20) drawPiece 함수: 보드 또는 스크린에 piece를 그리는 함수이다. 특정한 값이 좌표로 주어지지 않았다면, piece의 데이터 구조에 저장된 위치를 사용한다. piece를 구성하는 상자

들을 그린다.

21) drawNextPiece 함수: 다음 piece를 그리는 함수이다. Next: 라는 텍스트를 그리고 다음 piece를 그린다.

## 2. 함수의 호출 순서 또는 호출 조건

main: 프로그램이 실행될 때 최초로 호출된다.

=> showTextScreen: main 함수에서 호출

=> makeTextObjs: showTextScreen 함수에서 호출

=> checkForKeyPress: showTextScreen 함수에서 호출

=> checkForQuit: checkForKeyPress 함수에서 호출

=> terminate: checkForQuit 함수에서 호출

(조건: QUIT 이벤트가 발생했거나 KEYUP 이벤트로 Esc키가 눌렸을 경우)

=> runGame: main 함수에서 호출

=> getBlankBoard: runGame 함수에서 호출

=> calculateLevelAndFallFreq: runGame 함수에서 호출

=> getNewPiece: runGame 함수에서 호출

=> getNewPiece: runGame 함수에서 호출 (조건: fallingPiece가 None일 경우)

=> isValidPosition: runGame 함수에서 호출 (조건: fallingPiece가 None일 경우)

=> isOnBoard: isValidPosition 함수에서 호출

=> checkForQuit: runGame 함수에서 호출

=> showTextScreen: runGame 함수에서 호출. (조건: KEYUP 이벤트로 p가 눌렸을 경우). 이하의 함수들은 생략.

=> isValidPosition: runGame 함수에서 여러 번 호출되며 호출 조건이 다양함.

조건:

1) KEYDOWN 이벤트로 K\_LEFT 또는 K\_a 또는 K\_RIGHT 또는 K\_d 또는 K\_UP 또는 K\_w 또는 K\_q 또는 K\_DOWN 또는 K\_s 또는 K\_SPACE가 눌렸을 경우. 즉 piece를 이동하거나, 회전하거나, 떨어뜨리려 할 때 유효한 공간이 있는지 확인하기 위해 호출된다.

2) 키를 누른 채로 piece를 이동하려고 할 때, 유효한 공간이 있는지 확인하기 위해 호출된다.

3) piece를 떨어뜨릴 시간이 되었을 때, piece가 바닥에 착지했는지 여부를 검사하기 위해 호출된다.

=> addToBoard: rungame 함수에서 호출. (조건: piece가 바닥에 착지했을 경우)

=> removeCompleteLines: rungame 함수에서 호출.

(조건: piece가 바닥에 착지했을 경우)

=> isCompleteLine: removeCompleteLines 함수에서 호출.

(조건: y >= 0일 경우)

=> calculateLevelAndFallFreq: rungame 함수에서 호출.

(조건: piece가 바닥에 착지했을 경우 )

=> drawBoard: runtime 함수에서 호출  
=> drawBox: drawBoard 함수에서 호출  
=> convertToPixelCoords: drawBox 함수에서 호출  
(조건: pixelx가 None이고 pixely가 None일 경우)  
=> drawStatus: runtime 함수에서 호출  
=> drawNextPiece: runtime 함수에서 호출  
=> drawPiece: drawNextPiece 함수에서 호출  
=> convertToPixelCoords: drawPiece 함수에서 호출  
(조건: pixelx가 None이고 pixely가 None일 경우)  
=> drawBox: drawPiece 함수에서 호출  
(조건: piece를 구성하는 상자가 비어있을 경우)  
=> convertToPixelCoords: drawBox 함수에서 호출  
(조건: pixelx가 None이고 pixely가 None일 경우)  
=> drawPiece: runtime 함수에서 호출  
(조건: fallingPiece가 None이 아닐 경우). 이하의 함수들은 생략.  
=> showTextScreen: main 함수에서 호출  
runtime 함수가 끝나면, 종료 메시지를 스크린에 출력함