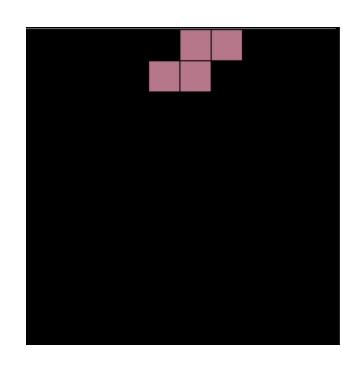
목차

- 프로젝트 소개
- 프로젝트 구조
 - GameLoop
 - Game
 - Renderer

프로젝트 소개

키 종류

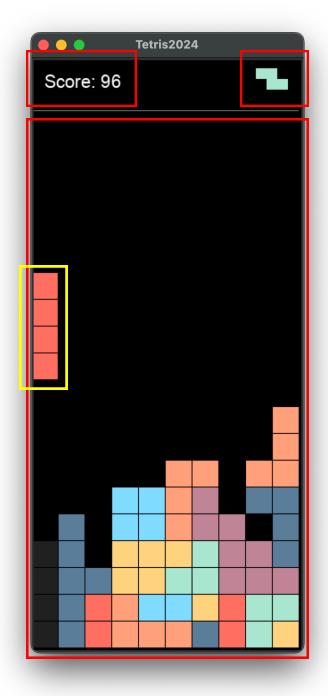
- 방향키(←,↓,→) : 블록 이동
- 방향키(↑) : 블록 회전(시계 방향)
- 스페이스바 : 낙하



프로젝트 소개

화면구성

- 현재 블록
- 보드
- 다음 블록 큐
- 점수 착지 시, 1+(지워진 줄)**2



프로젝트 소개

게임종료

• 현재 블록이 보드 상단을 나가면 종료

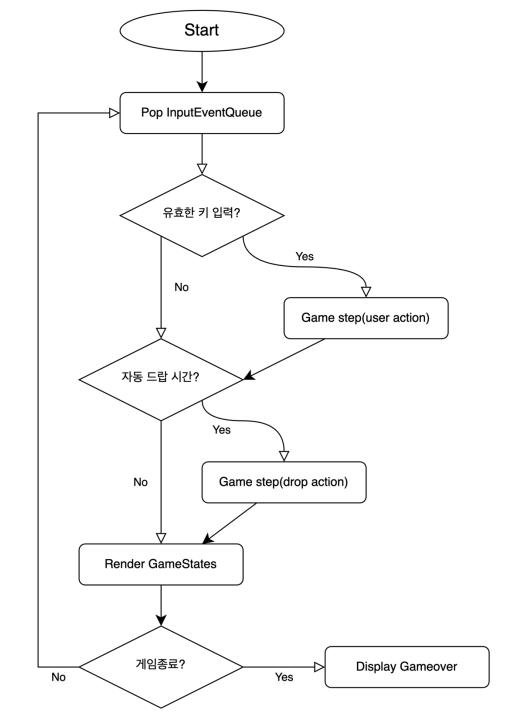


```
def main(w, h, bs) -> None:
    tk_manager = TKManager(
        width=w, height=h, block_size=bs, input_event_queue=InputEventQueue()
    game = Game(width=w, height=h, canvas=tk_manager.canvas)
    game_loop = GameLoop(
        game=game,
        tk_manager=tk_manager,
    game_loop.run()
```

- core/: 게임 로직 구현
 - game.py
 - tetromino_queue.py
 - randomizer.py
- graphic/ : 게임상태를 이미지화 수행
 - renderer.py
- io/: tkinter 관련 입출력 관리
 - tk_manager.py
 - input_event_queue.py
- → game_loop.py에서 위 모듈들을 이용하여 게임 실행

GameLoop

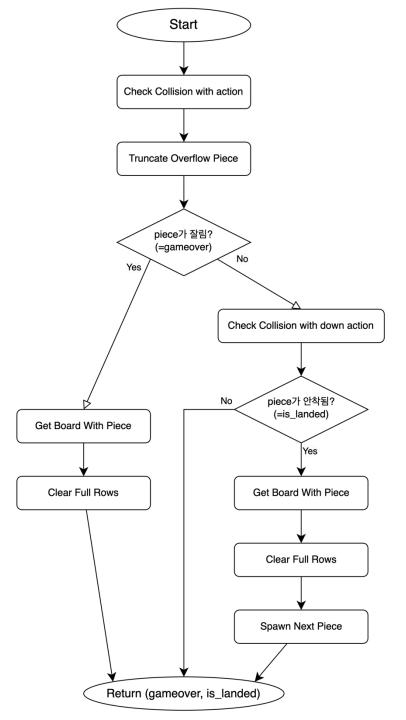
def update(self):



Game Game PIECES: List[List[List[int]]] DROP INDICATOR: int width: int def step(action: GameActions): height: int renderer: Renderer queue: TetrominoQueue - board: List[List[int]] piece: List[List[int]] GameStates - x, y: int -score: int idx: int cleared lines: int gameover: bool has a has a TetrominoQueue Renderer (randomizer: BagRandomizer | RandRandomizer) (width: int, height: int, block size: int, canvas: object) uses uses BagRandomizer RandRandomizer

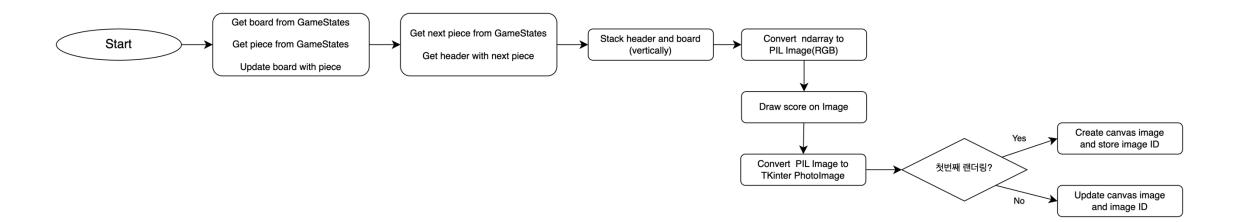
Game

def step(action: GameActions):



Renderer

• def render(gamestates: GameStates):



Renderer

• def render(gamestates: GameStates):

```
board = [[0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0,...]

board = [[0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0,...]

00 = [0, 0, 0, 0, 0, 0, 0, 0, 0]

01 = [0, 0, 0, 0, 0, 0, 0, 0, 0]

02 = [0, 0, 0, 0, 0, 0, 0, 0, 0]

03 = [0, 0, 0, 0, 0, 0, 0, 0, 0]

04 = [0, 0, 0, 0, 0, 0, 0, 0]

18 = [0, 0, 0, 0, 0, 8, 8, 0, 0, 0]

19 = [0, 0, 0, 0, 8, 8, 8, 0, 0, 0]

next_piece = [[5, 5, 5, 5]]

piece = [[0, 2, 0], [2, 2, 2]]

score = 0

x = 4
y = 0
```

