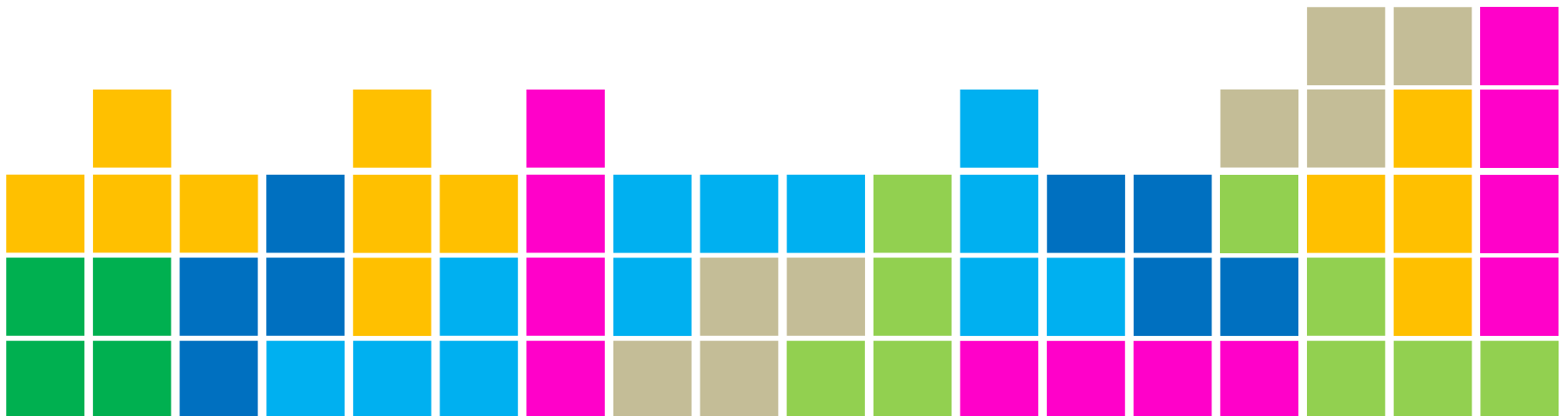


작성자 : 곽동운

# 테트리스 개발

## In C언어



# Contents

---

01 구상 및 설계

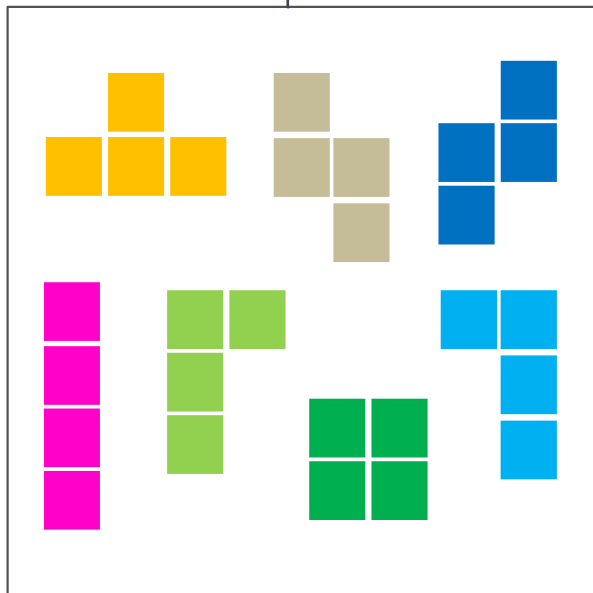
02 구현

03 테스트

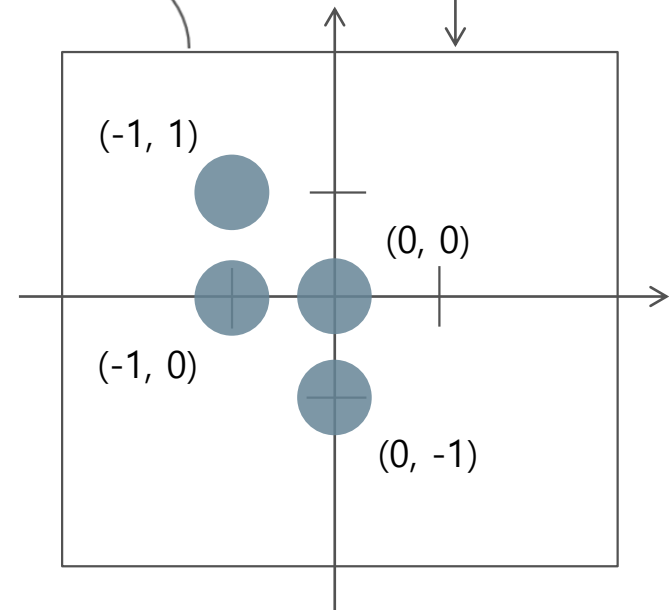
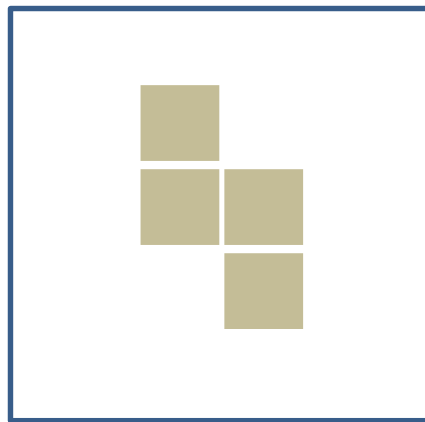
# 구상 및 설계 - 블록 구현

4개의 블록으로 이루어진  
7개의 블록 모양을 각각  
4개의 x, y 좌표로 구현

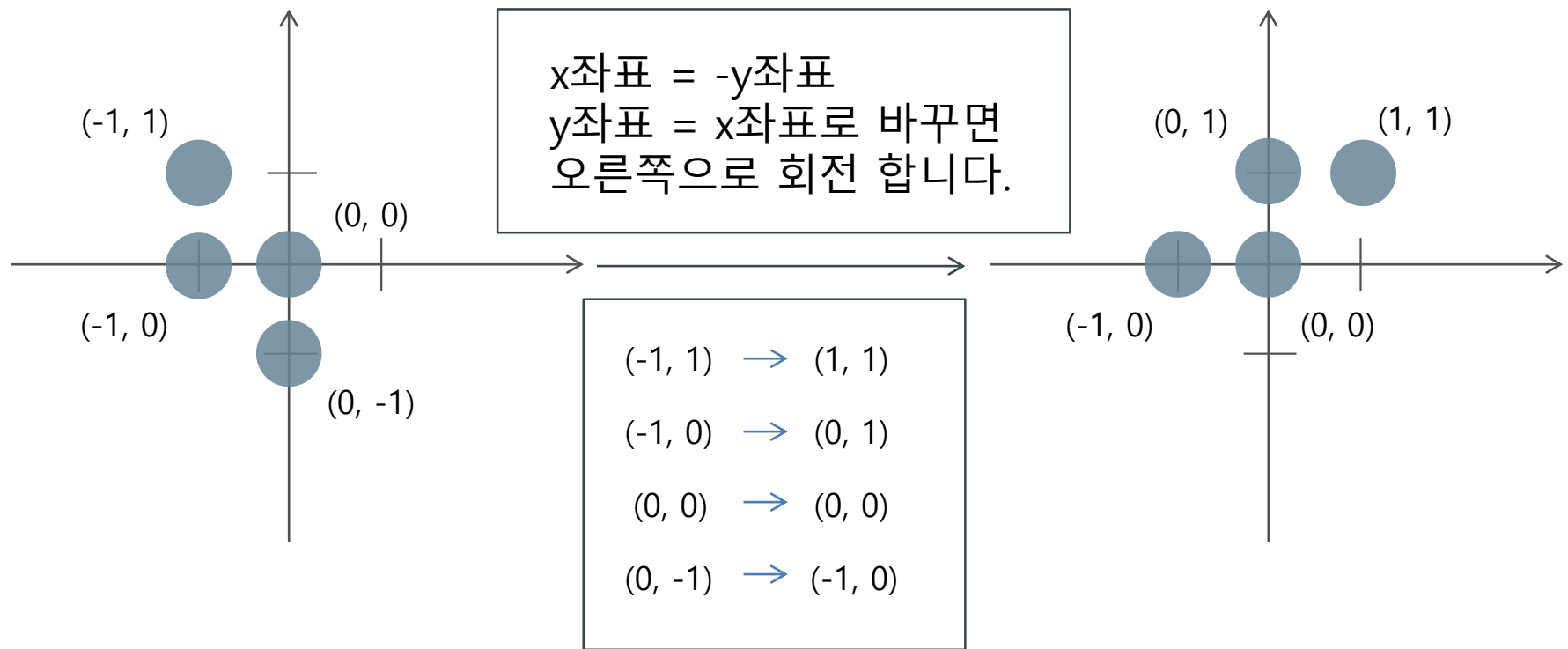
```
int shape[7][4][2] = {  
  {{0, -1}, {0, 0}, {-1, 0}, {-1, 1}},  
  ...  
}
```



블록 구현



# 구상 및 설계 - 블록 회전



# 구상 및 설계 - 화면구현

Width : 11, Height : 22

|   |    |    |    |   |
|---|----|----|----|---|
| 1 | 0  | 0  | 0  | 1 |
| 1 | 0  | 3  | 0  | 1 |
| 1 | 0  | 3  | 3  | 1 |
| 1 | 0  | 0  | 3  | 1 |
| 1 | 0  | 0  | 0  | 1 |
| 1 | 0  | 0  | 0  | 1 |
| 1 | 0  | -2 | 0  | 1 |
| 1 | -2 | -2 | -2 | 1 |
| 1 | 2  | 2  | 2  | 1 |

게임 기본 화면

- 좌우 벽: 1
- 바닥: 2
- 공백: 0

움직이는 블록

- 움직이는 블록: 3





충돌 후 멈춘 블록

- 멈춘 블록: -2

Screen에 저장

Screen[22][11]

변환 및 구현

|    |   |   |
|----|---|---|
| 1  | → |    |
| 2  | → |  |
| 0  | → | " "   |
| -2 | → |  |
| 3  | → |  |

# 구상 및 설계 - 화면 반짝임 처리

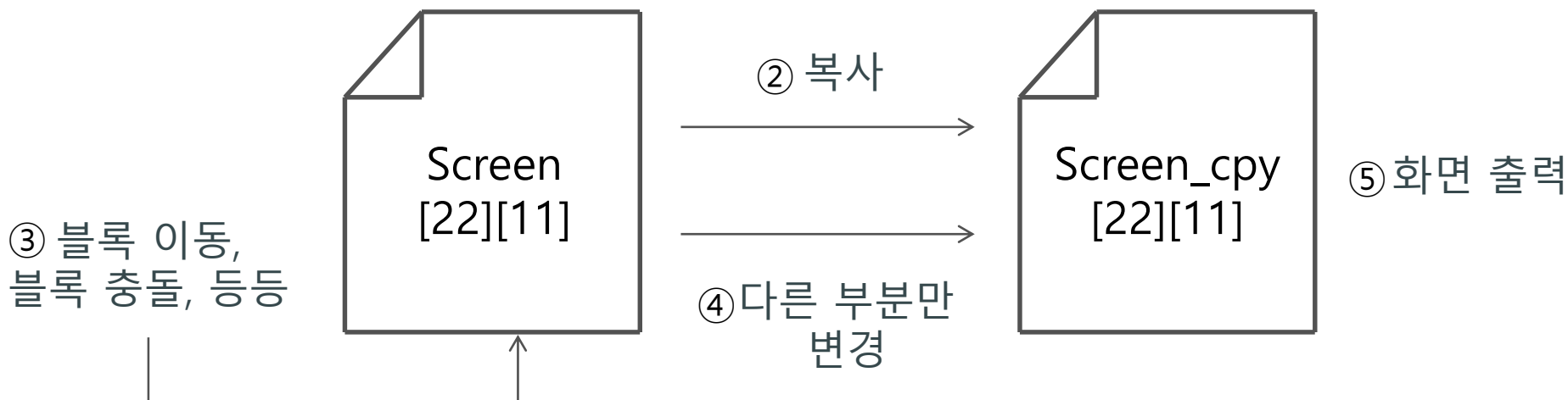
## ※ 문제점 ※

블록이 움직일 때 마다 `system("cls")`를 이용해서 모두 지워주고 새로 그려줄 경우 게임 창이 반짝이는 현상이 발생.

## ※ 해결 ※

두 개의 Screen 배열을 만들고, 블록 이동, 블록 회전 등 바뀐 부분만 새로 그려줌으로써 `system("cls")`를 사용하지 않고도 게임 진행이 가능합니다.

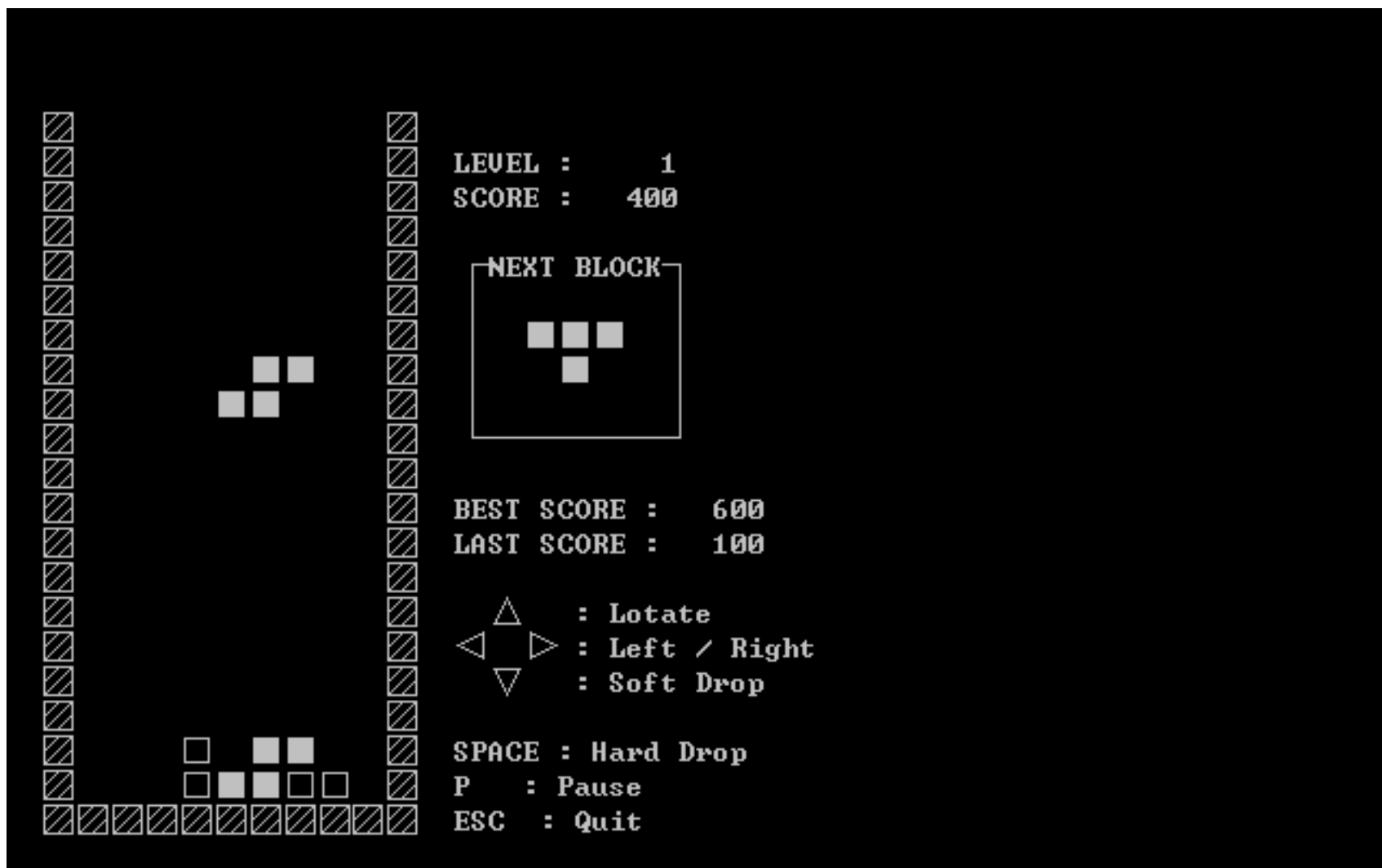
- ① Screen, Screen\_cpy 두 개의 배열을 선언



# 구상 및 설계 - 주요 메소드

| 메서드명             | 기능                 |
|------------------|--------------------|
| system("cls")    | 첫 화면 초기화           |
| setMain()        | 게임 화면 값들을 배열에 저장   |
| drawMain()       | 게임 화면 그리기          |
| getShape()       | 블록 모양 가져오기         |
| Piece()          | 블록 그리기             |
| rotateRight()    | 블록 회전              |
| key_check()      | 키 입력 확인 및 블록 이동    |
| check_line()     | 라인 체크 및 가득 찬 라인 제거 |
| check_crush()    | 충돌 확인              |
| check_gameOver() | 게임 오버 확인           |

# 구현





# 테스트

| 테스트                      | 결과 |
|--------------------------|----|
| 1. 화면 반짝임 여부             | ○  |
| 2. 방향키 입력 시 블록 이동이 잘 되는가 | ○  |
| 3. 블록 회전이 잘 되는가          | ○  |
| 4. 라인이 채워지면 지워지는가        | ○  |
| 5. 부가기능이 충분한가            | △  |
|                          |    |
|                          |    |
|                          |    |
|                          |    |
|                          |    |

# 개발 일정

| 개발내용        | 일정                    |
|-------------|-----------------------|
| 구상 및 설계     | 2019.04.22~2019.04.24 |
| 구현          | 2019.04.25~2019.04.28 |
| 테스트 및 수정 보완 | 2019.04.29~2019.04.31 |

!THANK YOU

Any questions?