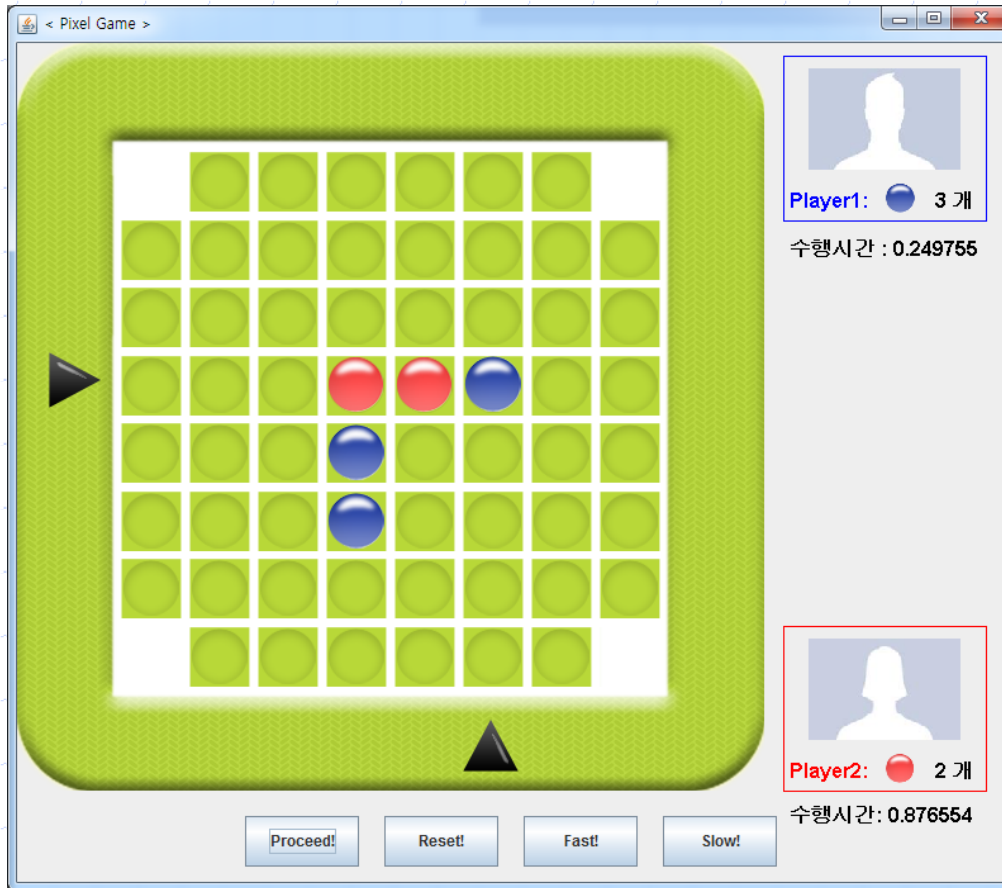


# Term Project 안내서

알고리즘및실습  
2018년 1학기

# 목적

- ◆ 픽셀게임의 돌을 두는 알고리즘을 개발하시오.
  - [픽셀 프로그램]



# 주어지는 프로그램 설명

## ◆ 픽셀 프로그램 소스 설명

### ■ PixelTester.java

- ◆ main 함수가 존재하는 Class

### ■ PixelBoard.java

- ◆ 그래픽 관련 처리 Class

### ■ Player.java (1/2)

- ◆ PixelPlayer를 위한 Abstract Class
- ◆ 멤버변수
  - currentPosition – 최근 자신이 둔 돌의 위치가 저장됨
  - **map** – 현재 픽셀판의 정보가 담겨 있음
    - 크기: 8 X 8 (인덱스 - 0~7 X 0~7)
    - map[i][j]에서 i는 행, j는 열을 의미 ex) i=3, j=4이면 4행의 5열의 위치
    - 값의 의미 - 0: 빈 공간, 1: Player1의 파란돌, 2: Player2의 빨간돌

# 주어지는 프로그램 설명

## ◆ 픽셀 프로그램 소스 설명

### ■ Player.java (2/2)

#### ◆ 멤버메소드

- void setCurrentPosition(Point currentPosition)
  - 멤버변수인 currentPosition에 현재 Player 자신의 돌 위치를 저장
- **abstract Point nextPosition(Point lastPosition)**
  - PlayerTester 에서 호출하는 함수
  - 상속 클래스에서 직접 구현해야 하는 함수
  - 입력 파라미터 lastPosition: 상대 Player가 가장 최근에 둔 돌의 위치
  - 리턴해야 할 정보: 현재 자신이 두고자 하는 다음 돌의 위치

### ■ PixelPlayer100.java, PixelPlayer101.java

- ◆ Player.java를 상속한 샘플 Pixel 플레이어 코드

# 주어지는 프로그램 구조

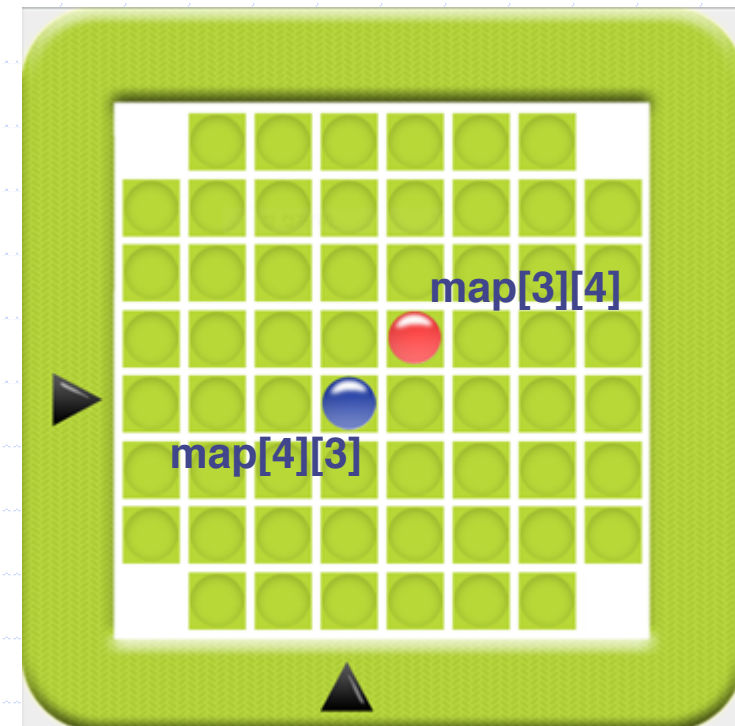
## ◆ 픽셀 프로그램 구조

- 학생들이 작성해야 할 클래스 소스: `PixelPlayer**.java`
  - ◆ `PixelPlayer**.java` 에서 \*\*은 각 조 숫자
    - 1조: `PixelPlayer01.java`
    - 2조: `PixelPlayer02.java`
    - ...
    - 10조: `PixelPlayer10.java`
    - ...
  - ◆ Player Class를 상속받아 작성
  - ◆ `Point nextPosition()` 함수를 작성
    - 이차원 배열로 구성된 현재 바둑판 정보인 `int[][] map`을 분석
    - 자신의 돌을 어디에 두어야 할지를 결정
    - 즉, 현재 주어진 `map` 정보에서 내가 이길 수 있는 가장 최적의 다음 돌 위치를 계산해 내는 알고리즘 구현
      - 예를 들어, Greedy Algorithm 또는 Dynamic Algorithm 등을 적용할 수 있음

# Pixel 게임 규칙

## ◆ 규칙

- 1. 시작은 파란색(Player1) 돌의 위치인  $\text{map}[4][3]$ , 빨간색(Player2) 돌의 위치인  $\text{map}[3][4]$ 에 하나씩 두어진 상태로 시작한다.
  - ◆  $\text{map}[4][3] = 1$
  - ◆  $\text{map}[3][4] = 2$
- 2. 슬라이더의 초기 좌표는  $\text{map}[4][3]$
- 3. 파란색 돌이 먼저 돌을 놓음(선공)



# Pixel 게임 규칙

## ◆ 규칙

- 4. 4개의 돌이 일렬로 먼저 놓아지면 승으로 간주
- 5. `nextPosition()`이 반환하지 말아야 하는 위치
  - ◆ 이미 돌이 놓여진 위치
  - ◆ 행과 열이 동시에 바뀐 위치
  - ◆ 배열이 벗어나는 위치 (0~7 X 0~7 외부)
  - ◆ 코너, 즉 (0, 0), (0, 7), (7, 0), (7, 7) 위치
- 6. 위의 경우에 해당하는 위치가 반환될 시 패로 간주
- 7. 무승부(Draw) 결과가 산출되는 경우
  - ◆ 맵 전체에 모든 돌이 놓여진 경우
  - ◆ 두 개의 슬라이더 중 어느 것을 움직여도 놓을 위치가 없는 경우
- 8. 무승부(Draw)인 경우, `nextPosition()` 함수의 총 수행 시간을 비교하여 시간이 적게 걸린 Player를 “무승부승”으로 간주
- 9. 작성한 알고리즘이 에러 (Exception)를 발생시킬 경우, 패로 간주

## ◆ 평가방법

### ■ 승점 배점 방법

- ◆ 승 : 무승부승 : 무승부패 : 패 = 3 : 2 : 1 : 0

### ■ 각 조가 작성한 PixelPlayer간의 리그전으로 전체 픽셀경기를 조교가 수행한다.

- ◆ 임의의 한 조가 다른 모든 조들과 선후공의 순서를 바꿔가며 2번의 경기를 수행하게 되면 한 리그가 종료된다.
  - 예를 들어, 총 16조이면 한 조의 경기 회수는 총 30회
- ◆ 승, 무승부승, 무승부패, 패를 각 조별로 합산하여 전체 조별 순위를 정한다.
- ◆ 추후 경기 결과를 공지하고 Term Project 발표회를 시행한다.
  - 1~4등을 한 조에 대한 추가적인 평가
    - 가장 높은 Term\_Project 점수 부여
    - 수업시간에 전략에 대해 간략히 발표
    - 수업시간에 재경기를 하여 순위를 최종적으로 재조정할 수 있다.



# 팀프로젝트 수행 방법

## ◆ Tip

- 친구들이 작성한 PixelPlayer를 미리 받아서 스스로 경기를 수행해보면 자신이 만든 알고리즘의 수준을 체크해볼 수 있을 것임.

## ◆ Report 요구사항

- 픽셀 알고리즘에 대한 전략, 절차, 방법을 구체적으로 그림등과 함께 제시할 것
  - ◆ Greedy Method를 활용했다면 어떠한 규칙으로 자신의 돌 위치를 계산해 내었는지를 명확하게 제시할 것
- 프로그램 코드를 정확하게 Report에 제시할 것
  - ◆ 주석을 충분하게 넣을 것
- 자신이 취득한 기술적 사항을 생각한 바 그대로 적을 것
- 질문은 Q&A 게시판에 올릴 것

# 텀프로젝트 수행 방법

## ◆ 제출형태

- 보고서 파일(HWP 또는 MS-Word)과 프로그램을 게시판에 업로드
  - ◆ 서론, 본론, 결론의 구조를 가진 보고서
    - 표지 (Term Project 문구 추가)
    - 서론: Term Project의 내용 및 목적
    - 본론: 각각의 문제를 코딩한 이후 앞에 있는 요구사항에 따라 작성
    - 결론: 숙제를 한 이후의 느낀점, 하고싶은 말, 또는 불평~~
- 파일명: 알고리즘-TermProject-2018-01.zip
  - ◆ 보고서와 PixelPlayer\*\*.java 를 하나의 zip 파일에 묶어서 만듦
  - ◆ 맨 마지막 숫자 01은 조 번호를 나타냄
- 게시판의 게시물 제목: 알고리즘-TermProject-2018-01
  - ◆ 맨 마지막 숫자 01은 조 번호를 나타냄

## ◆ 제출기한

- 2018년 6월 13일 (수요일) 23:59:59 - 절대엄수