

## IQ Link

COMP1110/1140/1510/6710 major assignment

Alex Smith (u5838800), Yicong Du (u6127601)

Week 11, 2016

# Table of Contents

1 Summary

2 Design

3 Screenshots

# Summary

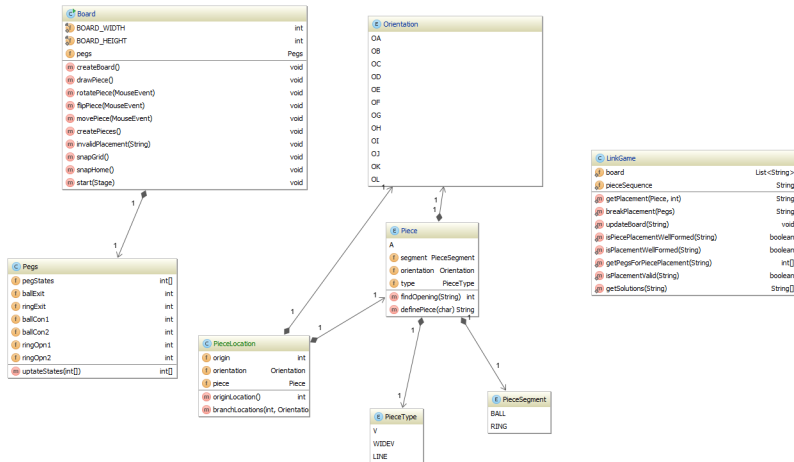
- Created Piece class with enumerators
- Added placement methods
- Implemented Viewer with one, then multiple, piece viewer
- Implemented full GUI for game
  - Piece drawing and movement
  - Game selection
  - Hint viewer as string, then image
  - Other QoL additions

Design

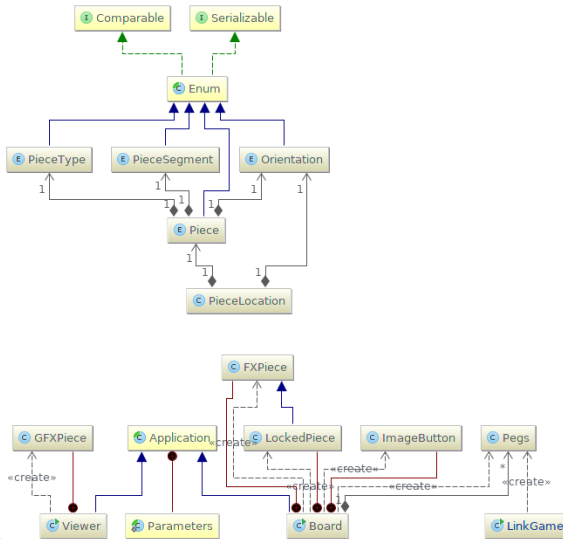
# Structure

- Enumerators for pieces and info, int array for pegs
- Store piece rotation, info, etc. in the enumerator
- Store ball/ring, connectors/gaps, in peg info
- Handle all GUI-related things in Board, all back-end in LinkGame

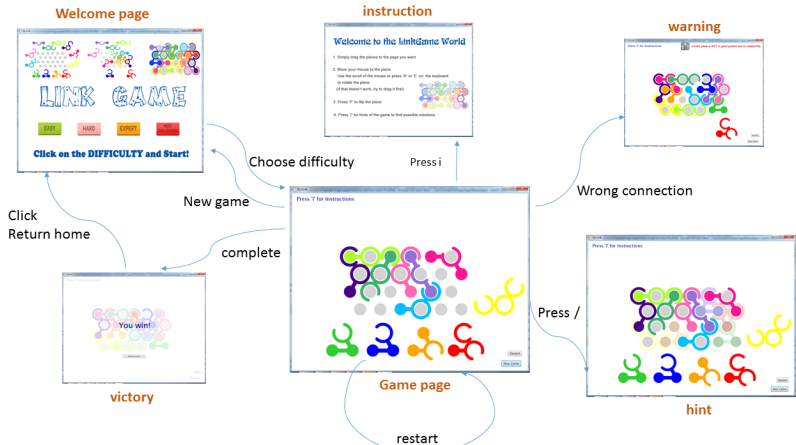
# Structure



# Structure



# Structure





# Piece and Peg classes, mk. I

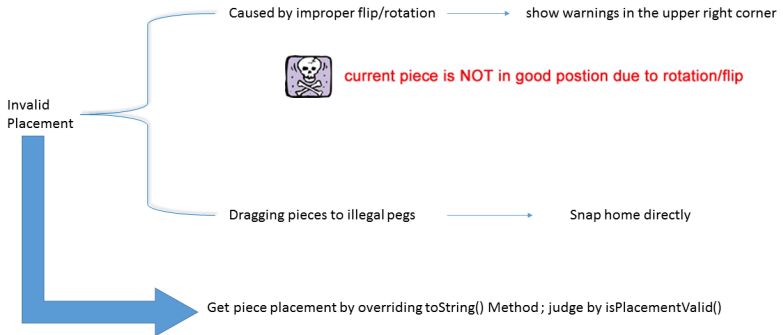
## Enumerator declaration

```
enum Piece {  
  A (LINE,0A,  
      RING,2,3,  
      BALL,4,0,  
      BALL,1,0),  
  ...  
}
```

## Peg declaration

```
public Pegs(int[] states) {  
    this.ballExit = states[0];  
    this.ballCon1 = states[1];  
    this.ballCon2 = states[2];  
    this.ringExit = states[3];  
    this.ringOpn1 = states[4];  
    this.ringOpn2 = states[5];  
}
```

# Invalid placements



# Game remake buttons



Two buttons

**New Game** → back to the welcome page

If the user select the same difficulty level as the former game

Can start with different initial placement (totally random)

**Restart** → back to the same initial placement

# Screenshots

# Welcome screen



## LINK

## GAME

EASY

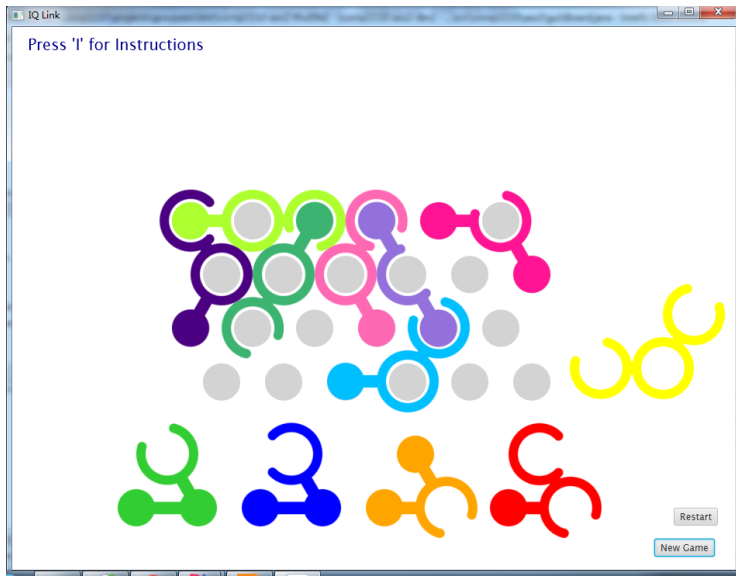
HARD

EXPERT

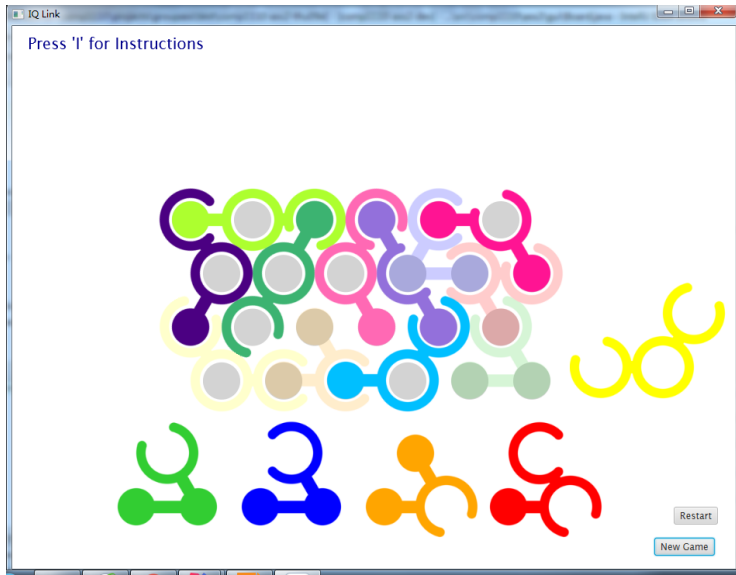
MOST  
CHALLENGING

**Click on the DIFFICULTY and Start!**

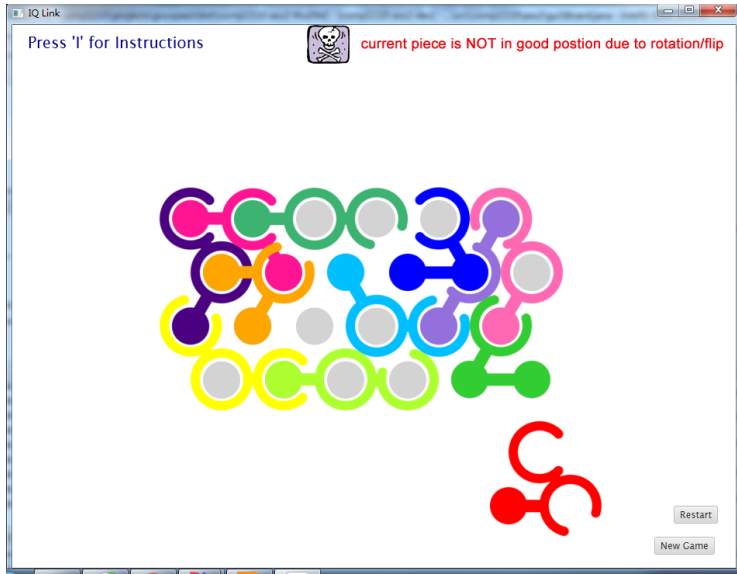
# Main game



# Hints

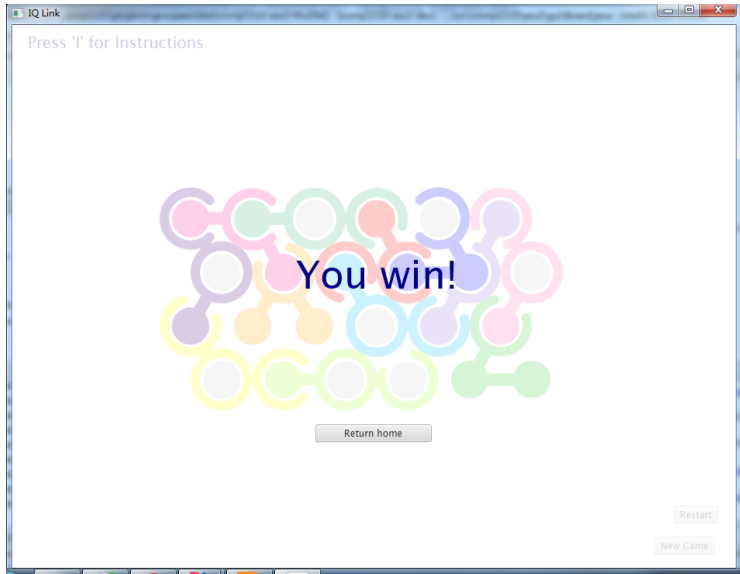


# Can't do that





# Victory



Questions?