Chain Reaction

(https://github.com/amrandahvur/AP-Project)

Madhur Tandon (2016053) | Aayush Aggarwal (2016002)

Design & Implementation

Design Patterns:

Chain of Responsibility:

The State of the board is deserialized and then passed into a function which builds the GUI from the raw function.

Iterator:

This is used to iterate over an arraylist of players.

State:

The State of the Board is seen and different operations are performed accordingly

The whole structure of the code implemented is modular and divided into different sections. There are 3 main components

1) **withoutGUI** part (used for serialization)
2) **application** part (used for GUI, includes JavaFX components)
3) **network** part (used for network play)

All 3 components are linked together using different class relationships.

Most commonly used in the design of the game is 2 way association.

Problems Faced

- Animation
  - Splitting / Exploding of balls after exceeding their critical mass, it used to happen only on the first instance of explosion. The animation was not carried forward for consecutive splitting. The final state of the board would appear correctly though.
- Serialization for Resume, Saving States for Undo
  - The initial design of the game was highly linked with JavaFX components. Since, JavaFX components when used as attributes inside a class make the class unserializable, we had to make new classes to accommodate only for keeping track of the corresponding raw form of the game data. This raw form was then used for re-building the GUI through a one-way function for both Undo and Resume.
- Network Play Mode (through Sockets)
  - Dynamic Addition of new players in the game was an issue (create **x** - player game if **x** people join the network). This, along with timing issues with respect to synchronisation of various threads usually lead to wrong state in some of the clients.

Individual Effort

Madhur Tandon

- Main Logic
- Splitting and Rotation Animations
  - Integrating Main recursive logic within the animation.
- Resume and Undo functionality
- Very Minimal GUI
  - Only for initial grid and orbs

Aayush Aggarwal

- Main GUI
  - Use of Colour Picker, linking different pages
- Major Bug Fixes
  - Problems associated with Undo and Animation when used together.
  - Problems associated with disabling of clicking of tiles between animation.
- Network Play Mode
  - Using Socket Programming

Bonus Components

Network Play

- Designed a network play feature, allows multiple players to play from different PCs on the same game.
- The first machine to start the game on the given IP acts as a server, the rest become clients.
- The server dynamically increases the player size as more clients join before the first move is made.
- Whenever an action happens on a machine (mouse click on tile or undo button), it broadcasts it over the network and the same action is performed on all machines. Since the game logic is a deterministic algorithm, the final state formed is same on all.