Final Project
– *Mancala* –

I. Driver class

Packages needed:
```
import java.util.*;
import javafx.*; // for 2d graphics
```

II. abstract Mancala class

Mancala	
`protected LinkedList[][] board;`	`// Board of stones`
<<constructors>> `public Mancala();` <<accessors>> *<none>* <<modifiers>> `public void move();` `public boolean isWon();` `public abstract void fillBackground();` `private void drawBoard();` `public void drawNumbers();`	 `// constructor for Mancala` `// move method for changing stone location (depends on game type)` `// return true or false depending on whether the user has won (depends on game type)` `// fills the background with a color (depending on game type)` `// draws the board onto the screen` `// draws the number of stones onto the screen next to eat pit`

III. Capture class

Capture	
<<constructors>> public Capture();	 // constructor for Capture
<<accessors>> <*none*>	
<<modifiers>> private void captured() public void move(); public boolean isWon(); public void fillBackground();	 // helper method for when stones are captured // move method specifically for capture game type // determines the winner based on rules for stones in pits // fills the background of the screen depending on the type

IV. Avalanche class

Avalanche	
<<constructors>> public Avalanche();	 // constructor for Avalanche
<<accessors>> <*none*>	
<<modifiers>> public void insertRecursion(TreeNode node, Object data) public void insert(Comparable element) public void fillBackground();	 // inserts element into the tree // calls recursive method for insert

V. Stone class

Stone	
private int xLocation private int yLocation private Color myColor	// stores the x location of the stone // stores the y location of the stone // stores the color of the stone
<<constructors>> public Stone(); <<accessors>> *<none>* <<modifiers>> public static void draw(Graphics g, int x, int y) private void setColor(Comparable element)	 // constructor for Stone // draws stone // sets stone color