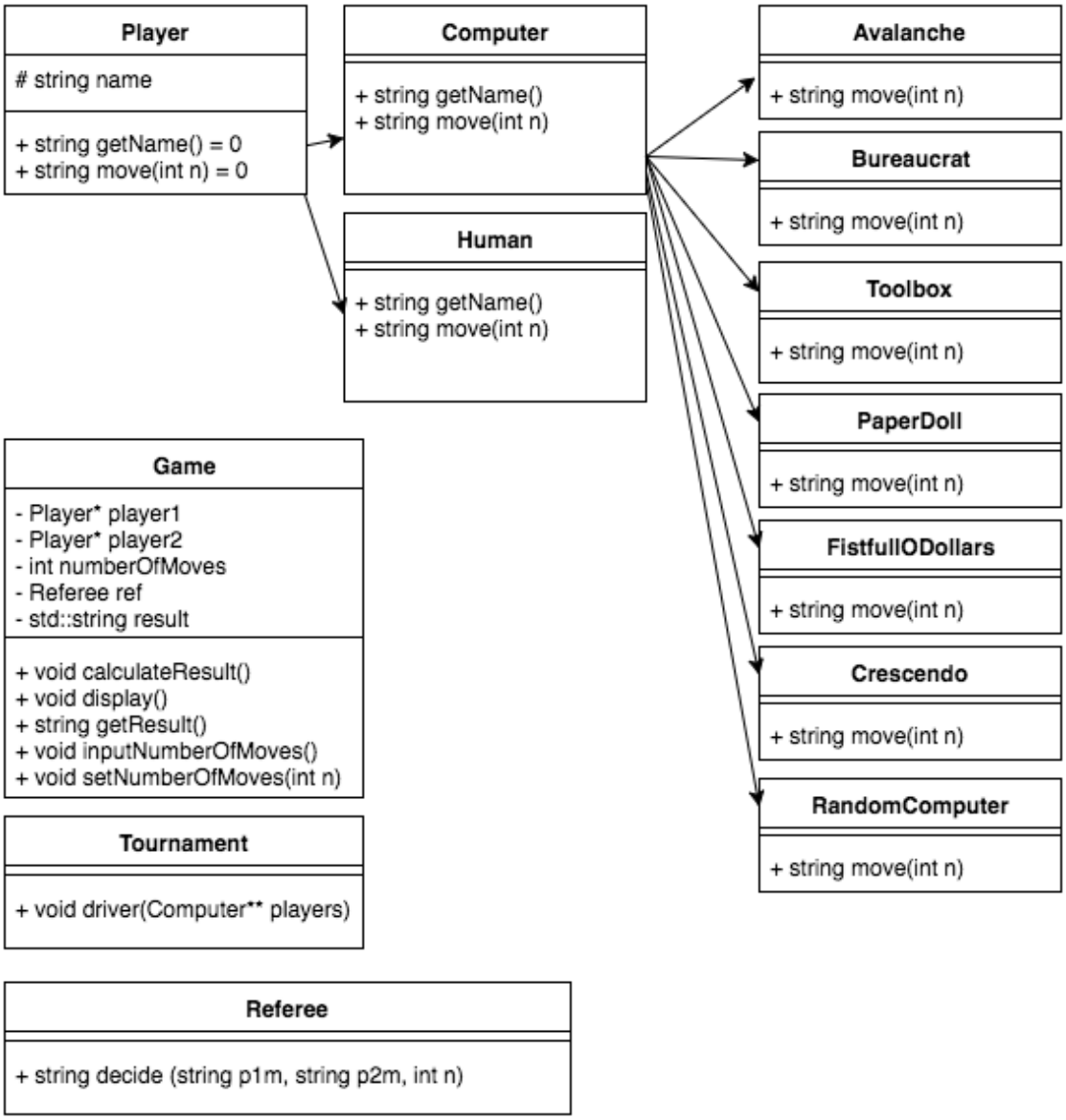
Rock, Paper, Scissors

UML Diagram



Description

Game:

When a game is created it takes two player pointers and stores them and also creates a Referee object.
It has 5 methods:

calculateResult() passes the two players moves to the referee and stores the results

display() outputs the results to console

getResult() returns the stored result

inputNumberOfMoves() takes the users input to set the number of moves

setNumberOfMoves() sets the number of moves to a given number

Referee:

Referee's behavior is to decide the outcomes of the moves given from both players

decide()
- creates a string to hold the results
- loops through each pair of moves and decides if player1 has won lost or tied (W, L or T) and stores the result in the corresponding place in the string.
- If the strings of moves it was given contain a character other than R, P or S, the result is set as 0.
- Returns the string of results

Tournament:

A Tournament object takes 8 players and competes them against each other in a tournament structure. It has one method:

driver()
- It takes an array of 8 players
- It then conducts 3 rounds and outputs a winner of the tournament
- The winner of each round is the one with the most wins out of 5 plays, if there is an even number, then the player with the lowest index in the array advances

Player:

Player is an abstract class, the parent of Human and Computer. It exists so that its children can both be used as one base class.

move() is a pure virtual method to be implemented in any children.
getName() is also a pure virtual method.

Human:

Is a player whose moves, Rock, Paper or Scissors (R, P or S) are entered by the user.

move()
- takes a number of moves
- creates a string to contain the moves represented a characters
- gets each move from the user and stores them in the respective spot in the string
- returns the string of moves

Computer:

Computer is a player that is fully automated, it always plays rock.
It has 7 child classes: Avalanche, ToolBox, Bureaucrat, PaperDoll, Crescendo, FistfullODollars and RandomComputer.

move() is a virtual method:
- takes a number of moves
- creates a string to contain the moves represented a characters
- sets each character in the string to 'R' for rock using a loop
- returns the string of moves

getName() returns the Class name which is different for all of it's children.

Avalanche, ToolBox, Bureaucrat, PaperDoll, Crescendo, FistfullODollars and RandomComputer:

These are all fully automated rock paper scissor players. Each is a child of Computer. They all have their own move() method:

Avalanche always plays rock
ToolBox always plays scissors
Bureaucrat always plays paper
PaperDoll plays paper, scissors, scissors
Crescendo plays paper, scissors rock
FistfullODollars plays rock, paper, paper
RandomComputer plays a random move every turn

When each of these classes are created their name is set to the classes name.

Main:

The main function starts a tournament, gets input for what 8 players should play in the tournament, creates each player and stores them in an array. That array is then given to the tournaments driver method to output the winner. The array of players is then deleted.

Testing

Test cases:

Input:
Avalanche Toolbox Bureaucrat PaperDoll Crescendo FistfullODollars Avalanche Toolbox
Expected output:
Avalanche

Input:
Avalanche Avalanche Avalanche Avalanche Avalanche Avalanche Avalanche Avalanche
Expected output:
Avalanche

Input:

(empty string)
Expected output:

(Blank and ready to input again)

Input:
Blah
Expected output:

(output error message and ready to input again)

Input:

(Blank)
Expected output:

(doesn't output until players are entered)

Input:
23098oihf']/32
Expected output:

(output error message and ready to input again)