

Sean Burke

CPSC 224

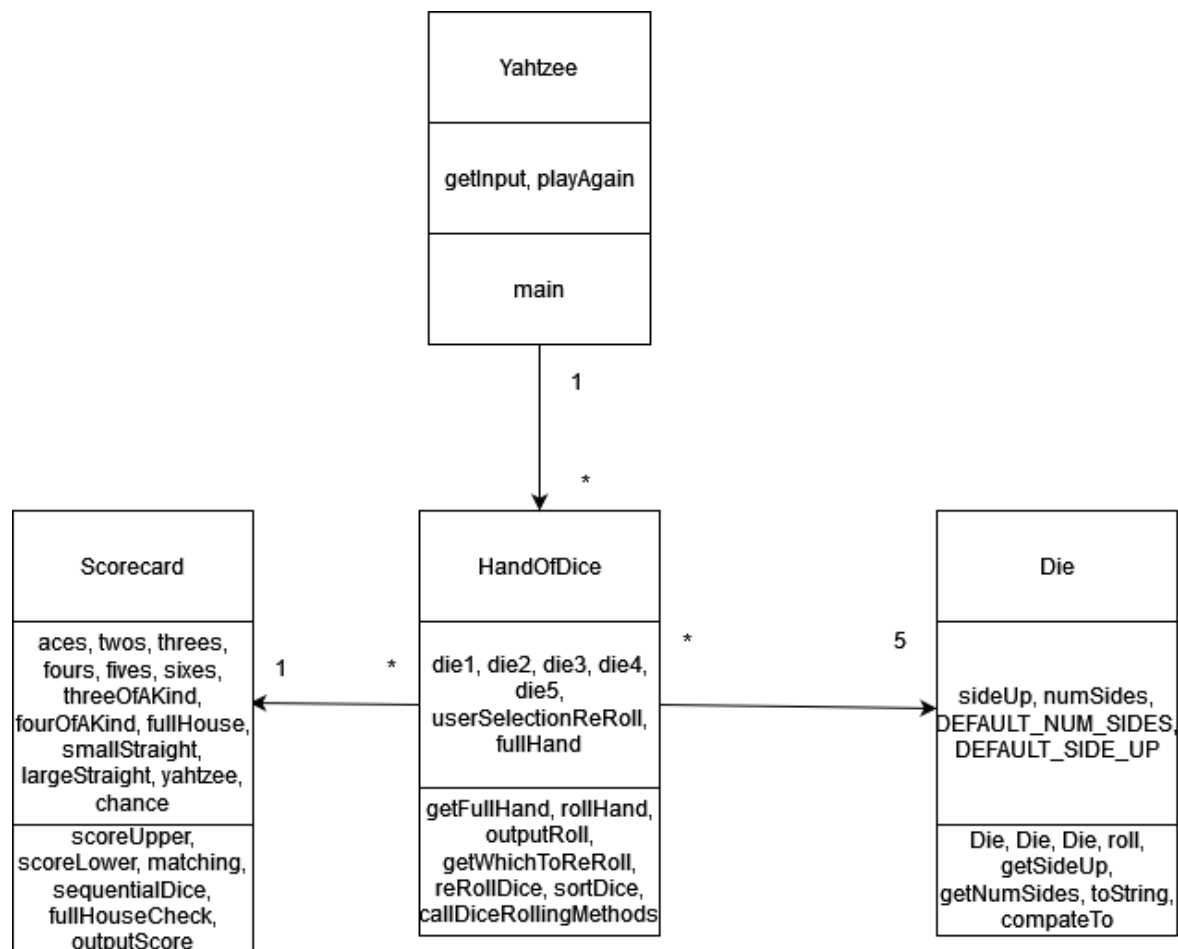
Crandall

1/26/2022

Summary of the goal: The goal is essentially to play through a single round of Yahtzee. This includes the rolling of the dice, and the 2 re rolls. It also includes an output of all possible points the user could have received with their final hand. Finally, it can be replayed as many times as the user wants but the scores do not carry over to the next round.

Design Overview: I made classes that splits the game into its logical parts. The classes are as follows Die which is responsible for the manipulation of a single dice. HandOfDice which is responsible for manipulating the entire hand of dice as well as outputting each hand to the console. Next, Scorecard is responsible for calculating all of the relevant scores as well as outputting those scores to the console. Finally, Yahtzee contains the main method and it is what controls the entire game.

UML Class Diagram:



Major issues: One of the major issues was there were some errors in the calculations of the scores. I incrementally tested when I wrote the code to see what worked but there were some special cases where certain aspects of my code would not work. I had to comment out sections to identify the issue and then rework the logic to make sure it worked properly. I troubleshoot this way because I find that it is the quickest method for me. After identifying where the problem was, I realized that there were some simple typos causing the errors such as ending a loop at the wrong index in the array for the dice. There were also some issues making sure that the necessary information was communicated across different classes. I knew it was due to improper syntax due to the fact that I am new to Java. I looked over the book to find the proper syntax to make sure that everything worked properly.

In Retrospect: I would rearrange some aspects of my code to clean it up and make it easier to read. I also would likely add another class responsible for handling a full round of Yahtzee (in anticipation of future assignments) so that main controls each round. This will make handling the game easier as it gets more complicated. I also would split the scoring into separate methods for each score so there doesn't need to be so many variables.