Fall 2022 CS3330 Final Project Documentation

Zach Thomas (zrtfcf)

December 2022

I. Overview

One of the most widely played card games around the world is Blackjack. Everyone loves it, but few know how to really play it well. Far too many people have and will walk into a casino with no game plan. They will play however they feel and bet whatever they feel like betting. It's a recipe for disaster and can ruin even the simple fun of Blackjack if you go in without a plan and lose all of your money.

II. Purpose of the project

The purpose of this project is to improve the way people play Blackjack and possibly prevent them from sitting at a table without a plan and losing money. This project aims to teach the basics while exposing the user to the best decisions possible to help them learn when to do what, along with showing them the odds of success. Listed are bullet points that provide clear advantages to using my program to learn Blackjack basics.

1. The program tells players the general rule of thumb for what they should do before they start a game of Blackjack.
2. User's hands are held every step of the way and they are talked through every decision that needs to be made.
3. Users are able to get real practice playing Blackjack before they even go to a real table. This is valuable so they don't make the wrong decision when it comes time to play for real since stress can be high in a casino environment.

III. Functionalities of the project

There are two main functionalities in the app: Custom buy ins and wagers, and a cheat sheet

1. Custom buy ins and wagers: Users are able to simulate a realistic betting system to provide enhanced practice for the real deal. They can buy in for pretty much any realistic amount and wager what they wish. Although, users are guided and wagers are suggested to them.
2. The trainer provides a chart to the player which exposes them to the statistically best move possible. This allows the user to see the best options possible based on the current hands and learn from it.

Overall, the program provides a simulated betting environment while allowing the user to practice while seeing what they should do in a situation if they thought otherwise.

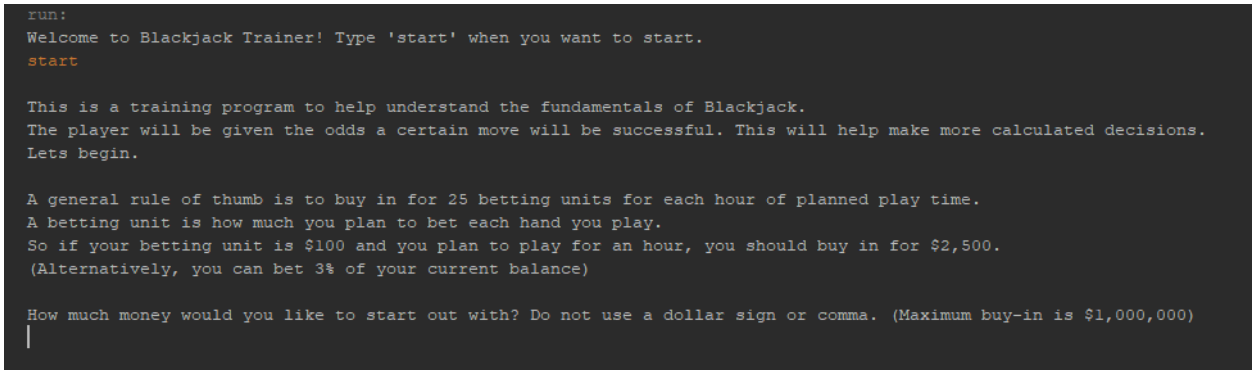
IV. Required elements

1. Object Oriented Elements
    a. Classes
        i. FinalProject.java
        ii. Card.java
        iii. Deck.java
        iv. Hand.java
        v. Player.java
        vi. Dealer.java
        vii. Game.java
    b. Subclasses
        i. Player is a subclass of Hand
        ii. Dealer is a subclass of Hand
2. Code Elements
    a. Multiple Collections
        i. ArrayList in FinalProject.java (Line 143)
        ii. ArrayList in Deck.java (Line 12)
        iii. ArrayList in Hand.java (Line 12)
        iv. ArrayList in Game.java (Line 15)
    b. Exception Handling
        i. Try-Catch statement in FinalProject.java (Lines 54-115)
            1. Throws: InterruptedException for interrupted thread
        ii. Try-Catch statement in Game.java (Lines 28-62)
            1. Throws: InterruptedException for interrupted thread
        iii. Try-Catch statement in Game.java (Lines 68-72)
            1. Throws: InterruptedException for interrupted thread
        iv. Try-Catch statement in Game.java (Lines 173-180)
            1. Throws: InterruptedException for interrupted thread
        v. Try-Catch statement in FinalProject.java (Lines 243-277)
            1. Throws: InterruptedException for interrupted thread

V. Program In Use

Usage of the program is rather simple as the user is walked through what to do for each step or given a custom option for what they want to do. Although, they are barred from entering invalid characters into the field.

1.



   a. Users are welcomed into the program and asked to type start. (The word “start” is only accepted. Not case sensitive.)
   b. Afterwards they are introduced to the purpose of the program and how it will work.
   c. The user is given the general rules of thumb and alternatives to optimize their betting strategy.

2.

   a. The user is prompted on what amount they want to buy in for and how much they want to bet.
   b. Afterwards simulating shuffling and dealing happens.
   c. The user is told the basic premise of Blackjack before they play and the basics of how to play. They are also given a key to the cheat sheet they are provided after the cards are revealed.
   d. The cheat sheet provides a full list of moves a user can do and the optimal decision for each hand.

```
Would you like to hit or stand? (48% chance to bust at this value)
hit
///Bet: 3//////////////////Balance: 97///
Player hand is:
[Queen of Spades, 6 of Clubs, King of Clubs](26)
Dealer wins.

Dealer hand is:
[Queen of Hearts, 4 of Clubs](14)


Play Again? (yes/no)
no
You left the table with $97.
Thanks for playing!
BUILD SUCCESSFUL (total time: 10 minutes 24 seconds)
```

3.
    a. According to the chart, hitting was the most statistically probable to provide a win. Although, we got unlucky this time.
    b. The player is given another card and it causes them to go over 21. Making them lose.
    c. After each round the user is confronted with a prompt asking them if they want to play again (yes or no is only acceptable). For this purpose we didn't want to, so the program told us are ending balance and thanked us for playing. Then terminating the program.
    d. Had I chosen to play again, it would have pretty much replicated photo 2.