BlackJack Benchmark Project

Team

Benjamin Lieu
Ian Hamilton
Sidhant Karamchandani
Yugraj Singh
Prajan Chauhan

Proposed Project

Black Jack Game between AI players created by different programming languages.

Dealer would be programmed in Java. He will follow set rules: for example, ALWAYS hit on a total that is less than 17. There will be multiple players: one programmed in Haskell, one programmed in Java, and one programmed in Python. The board itself will be programmed in Java, and will display a simple, green felt table with the players' cards' values showing to the user. Each language will be programmed with set strategies created by us. For example, ALWAYS hit when the player's total amount is 12 or less, at a total of 15 look at the cards around to see whether there are low cards or high cards still in the deck, completely stop at 18, etc. We will keep track of two things: how long each AI takes to make each move, and how many wins/losses/busts each player has. We can use this data over a period of many games to benchmark and compare the performance of these different AI's.

Time Budget

Board Implementation	6 Hours
Haskell Player	6 Hours
Java Dealer	3 Hours
Python Player	5 Hours
Deck/Cards Implementation	5 Hours

Total Hours: 40+ Hours (including connection issues and debugging)

Timeline (when will you do which parts?)

We would need to set up who would do what in the project. We would have a team that work on the board and another that would work on the A.I. for Haskell. Once one of the teams finish their part they would work on another A.I. in a different language.

Week 1: Algorithm Set Up
Week 2: Board/Deck and Cards Implementation

Week 3: Deck and Cards Implementation/Player and Dealer Implementation
Week 4: Player and Dealer Implementation/Communicating between segments
Week 5: Fine Tuning/Bug Fixes

• include time for presentation and report write-up

– **Risks** (always consider – and mitigate - risks!)
We have two main problems, but they both stem from the same one problem: *HOW TO COMMUNICATE DATA BETWEEN MULTIPLE LANGUAGES?*

1) How will we send data between the players to notify each one that a turn is complete, and how will we update the table display based on this?

2) How will we send card data between the dealer, deck, and players, before a player makes a decision?

Our proposed solution is to use socket programming through a local port.