Lab 4

CS 3793/5233 – Fall 2015 assigned October 25, 2016
Tom Bylander, Instructor due midnight, November 18, 2016

In Lab 4, you will improve a program that plays a solitaire version of Farkle. Your grade on the lab will depend on the performance of your program on 1000 turns of the game. The initial program is ailab4.zip, which you can download from the course web site.

Farkle

This description is based on the standard scoring described in http://en.wikipedia.org/wiki/Farkle.

You can play Farkle by running the main method in `Farkle.java`. Your part of the protocol is to enter the dice that you want to set aside (dice that add to your score) and to enter "bank" if you decide to finish the turn at that point.

At the start of a Farkle *turn*, six dice are rolled. After the roll, you may *set aside* any dice that score adding to your turn score. At this point, you can *bank* your turn score: the turn ends and your turn score is added to your total score. Or you reroll the remaining dice (the dice that are not set aside), set aside dice that score adding to your turn score, decide to bank or not, and so on.

After any roll (including the first roll), if no dice can be set aside, then you have *farkled*: the turn ends and your turn score is zero.

If you are able to set aside all six dice after one or more rolls, then you have *hot dice*, and the next roll will include all six dice.

The scoring is as follows. Considering only the dice that were rolled last, three 1s, three 2s, three 3s, three 4s, three 5s and three 6s score 1000, 200, 300, 400, 500, and 600, respectively. Any 1s not in a three of a kind score 100 each. Any 5s not in a three of a kind score 50 each. You can score combinations of the above on the same roll, e.g., a 1 and a 5 and three 6s adding 750 to your turn score.

The current `FarklePlayer.java` will only set aside 1s and 5s and always banks after the first roll.

Illustration of Farkle

These are examples of running `Farkle.java`.

In this run, the user asks to play one turn. Only dice number two is set aside, and the turn score is banked.

```
1
6 1 6 6 3 3
2 bank
score 100 banked, average score = 100.00 after 1 turns
quit
```

In this run, four dice are set aside, and the turn score is not banked. The next roll is a Farkle, so all 700 points were lost

```
1
6 1 6 6 3 3
1 2 3 4
score 700 continue
X X X X 4 6

_
score 0 farkled, average score = 0.00 after 1 turns
quit
```

In this run, the 700 is banked.

```
1
6 1 6 6 3 3
1 2 3 4 bank
score 700 banked, average score = 700.00 after 1 turns
quit
```

Finally, in this run, 2 turns are played. The first turn resulted in 700 points. The next turn has three rolls of the dice, eventually coming up with four 5s, and an additional 200 points.

```
2
6 1 6 6 3 3
1 2 3 4 bank
score 700 banked, average score = 700.00 after 1 turns
4 3 5 4 2 3
3
score 50 continue
6 5 X 5 2 2
2 4
score 150 continue
6 X X X 2 5
6 bank
score 200 banked, average score = 450.00 after 2 turns
quit
```

Your Task

The current `FarklePlayer.java` will achieve an average of about 214 over 1000 turns. Your task is to improve `FarklePlayer.java`:

1. Set aside three of a kinds. This should increase the average score to over 300.
2. Make better decisions about whether to bank or not. This can increase the average score to about 400.

To decide whether to bank or not, you should determine the *expected reward* of rolling the dice that are not set aside. You need to calculate the average turn score for the dice (the average over all possible values to the dice that are not set aside). For any possible dice roll, remember that either you will add to your current turn score, or you get a farkle and your turn score goes down to zero. You should bank if the expected reward is less than your turn score.

Your Grade

Your grade on this lab will range from 50 for an average score of 300 to 100 for an average score of 400. Just like the previous lab, there will be bonuses for the best performers. The five best performances in this lab will receive an additional 100, 80, 60, 40, or 20 points. Any ties will result in proportional allocation. For example, if 10 students all tie for the best performance, they will get an extra 30 points each.

Improvements

The expected reward described above is really the expected immediate reward, not the expected total reward. The expected total reward for any combination of turn score and number of dice to reroll can be determined by value iteration, which would be a preprocessing step before playing Farkle. See section 9.5 in the book and the notes.