CS221 Project02 – Mancala again

Paul Haskell

INTRODUCTION

During Project01, you wrote a program that allowed two people to play Mancala against each other. In this Project02, you will write a program that lets one person play Mancala against your computer program. Of course, you can reuse code from your own Project01.

A significant portion of this project is to figure out how to make the computer play intelligently. From your playing, did you figure out any strategies? Since a computer is so good at, uh, computing, are there strategies that a computer could use that are not reasonable for a person? You will have to experiment with strategies, and software development for those strategies, to complete Project02.

Your program

Your program should be called **mancala2.c** and should be saved to a **Project02** directory in your GitHub repo. As with Project01, the program should print out a Mancala board that looks something like this:

```
      6     5     4     3     2     1
B:    5     5     5     5     5     5
0                                       0
      5     5     5     5     5     5   :A
      1     2     3     4     5     6
```

Now the program prompts Player A to "make a play" i.e. to enter the index of one of her nonempty basins. The program should check player A's input for validity and should re-prompt Player A if the input is not valid.

Once Player A's turn is complete, your computer program must decide on the best move for Player B. The computer then makes the chosen move for Player B, tells Player A what the move was, and prompts Player A to make her next move. This repeats until the game is over. As with Project01, your program should print out the final scores for Players A and B.

Some hints

Captures can swing a player's score a lot, either up or down. Your computer strategy should look for the possibility of a capture, and the risk of a capture by the human opponent.

How can you test your strategy for goodness? One way is to have two versions of your Program02, each with its own strategy. You have a "champion" and "challenger", and whenever the challenger wins, it becomes a new champion, and you then can look for even more strategy improvements.

CONCLUSION:

Please push your program to GitHub before the project deadline. The deadline for completion of Project02 is 11:59pm Monday March 31.

Wouldn't it be fun to have your program play against your classmates' programs? We will do that in Project03!

Task	Score, points
mancala2.c compiles, runs, and prints properly formatted game board and prompts	5
mancala2.c follows the rules of the game, runs games to their proper conclusion, and prints a correct final score	15
mancala2.c uses an intelligent and effective playing strategy, as judged subjectively by the grader	15
Software and design quality, as judged by the grader	20