CSE 232 Spring 2016

Honors Project #2

Assignment Overview

This is the second of two projects needed to get Honors credit. This one is due before midnight on April 28th, 2016 (a Friday).

Overview

OK, you know all about dots and boxes https://en.wikipedia.org/wiki/Dots_and_Boxes and you wrote the game playing elements and a random player to play the game. Now you want to write a much better player that should win the game.

Winning

How do you win the game? Can you always win the game. It turns out that d&b is a kind of game called a combinatorial game.

- there are two players moving alternately;
- there are no chance devices and both players have perfect information;
- the rules are such that the game must eventually end; and
- there are no draws, and the winner is determined by who moves last.

Because of this, it is possible to come up with a winning strategy.

Winning Player

You wrote a random player, now I want you to a winning player function called winning_play. You are going to pit winning_play against random_play (your old random player) and see who wins. The outcome should be obvious.

Rather than explain how to win, I would like you to look online and see if you can find a winning strategy. As a start, check out Elwyn Berlekamp's (he wrote a book on dots-and-boxes, no lie) youtube video https://www.youtube.com/watch?v=KboGyIilP6k He lists some good strategies there. You can also find more such info.

Do as much as you can to create a winning player. You may not have time to "solve" the game but you should be able to do ***very well*** against a random player.

Main Program

You are going pit `winning_play` against `random_play` to play dots-and-boxes. Your main program will operate in two modes:

- single play: there are two games here.
  - One with `random_play` first
  - One with `winning_play` first.
  - Generate the two games in great detail in the file `single_play.txt`. That exact name!

- multiple play: play a lot of games, (flip a coin who goes first) and see who wins statistically.

More detail

- prompt for a random seed integer
- prompt for a number of rounds the multi-player (shown below) will play
- in multiple play "flip a coin" to see who goes first and report it
- your program will play two rounds using `winning_play` and `random_play` players and report each individual play, stored in a file called `"single_play.txt"` to be created in the executable's directory.
    - For each turn:
        - draw the grid
        - the score
        - whose move it is
        - what move was made
        - was that a box just made, did they go again
        - anything else that might be useful
- it then will play the provided number of rounds using a `winning_play` and a `random_play` players and report statistics of the results to a file called `"multiple_play.txt".` The statistics to include are:
    - the number of rounds played
    - average score overall and for each player
    - the median score overall and for each player
    - the highest and lowest score achieved for each player
    - anything else you think might be a good idea

Deliverables

As before, I want just one code file named "honors2.cpp" with everything in it. Makes it easier to grade. Handin will be to project13 named honors2.

One additional thing. I want a text file (no more than a page) that describes the strategy you implemented and how I can tell that it is working. You might want to embed in the `single_play.txt` file some info about when you used the strategy so it is clear to me.

Text file name should be `strategy.txt` Exactly that name!

Notes

- your `winning_play` statistics should be awesome, though not necessarily a clean sweep.
- as before, you should be able to play thousands of games in multiple play to get some good stats.
    - if it takes minutes to do that, you need to rethink it.