EECS 204002 Data Structures 資料結構

Project #3: Critical Mass Chain Reaction 連環爆

1. Project Objective:

Apply the knowledge learned from the course of data structures and implement a competitive algorithm for the Critical Mass Chain Reaction game.

2. Project Description

The gameplay takes place in a 6x6 board of squares. For each cell in the board, we define a critical mass. The critical mass is equal to the number of orthogonally adjacent cells. Therefore, that would be 4 for usual cells, 3 for cells in the edge and 2 for cells in the corner. All cells are initially empty. The Red and the Green player take turns to place "orbs" of their corresponding colors. Each player can only place an orb of the chosen color either in an empty cell or a cell which already contains the same color orbs. When two or more orbs are placed in the same cell, they stack up. When a cell is loaded with a number of orbs equal to its critical mass, the stack immediately explodes. As a result of the explosion, the cell of explosion loses all orbs and becomes empty. At the same time, an orb is added to each of the orthogonally adjacent cells and all orbs of these orthogonally adjacent cells change color to the color of the orbs of explosion. The explosions might result in overloading of an adjacent cell and the chain reaction of explosion continues until every cell is stable. The winner is the one who eliminates every other player's orbs.

You are to design a competitive algorithm and TAs will integrate your code to act as one of the game players. Your code is evaluated based on whether your code plays the game correctly and how well it plays against predefined bots.

3. Project Submission Rules
   a. Submit your program to https://nthuds.azurewebsites.net/ and your project report ILMS.
   b. Please use GitHub for source code control and show your program development history. Please follow the version control rules when doing programming.

   c. Grading Policy
   The basic evaluation is as follows:
   - 20%—if code can be successfully compiled on the competition server and can successfully play one game (regardless of outcome);
   - 30%—if code could beat *randomMove*, a bot that performed random moves;
   - 40%—if code could beat *noLook*, a bot with a board evaluator but no game tree;
   - 50%—if code could beat *heithoff*, a bot with a board evaluator and game tree with four levels of look-ahead;

- 60%—if code could beat *rlawrenc*, a bot with a very strong board evaluator and game tree with four levels of look-ahead;
- Bonus credit awarded for the top 10 ranked students.

d. Report & Demo (20%)
  1. The report file should be named "**report.pdf**".
  2. Each person should reserve a 15-min demo with TA. During the 1-on-1 demo, TA's will ask you questions related to your project report, test case and your implemented code. TA may ask you to compile and execute your program on spot.
  3. Your project report is recommended to follow this outline:

```
1) Project Description
        1-1) Program Flow Chart
        1-2) Detailed Description
2) Test case Design
        2-1) Detailed Description of the Test case
```

Note: The project report is limited to 10 pages.
Note: Your report can be either in Chinese or in English, or mixed.

Etiquette
a. **Do not plagiarize others' work, or you will fail this course.**
b. **No acceptance of late homework.**
c. **Please frequently check the class website announcements for possible updates.**