CSIS, BITS Pilani K. K. Birla Goa Campus

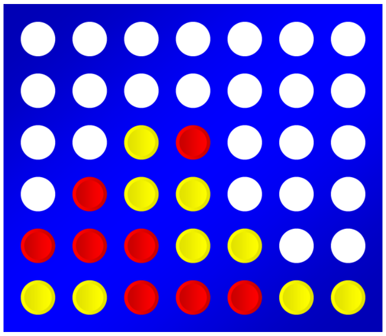
Artificial Intelligence (CS F407)

Programming Assignment 2
Total Marks: 15
Submission Deadline: 9 PM on 14/11/2023 (Tuesday)

Each student must individually do this programming assignment. Your program must be written in Python and should run (without errors) on Python 3.6 or later.
Take the honour pledge by filling this form: `https://forms.gle/oNdMC6pKkjmdY3gL8`
Your assignment will be evaluated **only** if you take the above honour pledge.
Any violation of academic integrity will result in -5 marks being awarded to everyone involved. There will be no differentiation between minor and major integrity violations.
Note that the deadline is **9 PM** and not midnight. Five marks per day will be deducted for submissions after the deadline. It will be your responsibility to submit the assignment well in advance and avoid unforeseen problems like power failures etc.

Question 1 (15 marks)

Consider the Connect 4 game shown below. More details about the rules of Connect 4 game can be found here:
https://en.wikipedia.org/wiki/Connect_Four.



This assignment includes a program for playing against a Myopic player that looks only one move ahead before choosing the best action. Run the "ROLLXYZ_FIRSTNAME.py" program file. (You can play against the Myopic player and see the code.)

Write a program that uses Game Tree based search to play against (and beat) the Myopic player. The Game tree based program should look 5 moves ahead (i.e. game Tree has a cut-off depth of 5).

Note that the Myopic player *always* makes the first move. Myopic player is always Player 1 and Game Tree player is always Player 2.

(a) What evaluation function should be used by the Game Tree player program? Compare different evaluation functions you have considered. The comparison can be based on the number of games won (out of 50 games) and average number of moves before each win. (If an evaluation function is better then the average number of moves before winning should decrease and number of games won should increase.)

(b) Initially, let the Game Tree based program look only 3 moves ahead so that the program runs faster. Implement alpha-beta pruning and compare different evaluation functions. Once your program runs as expected, increase the depth of the game tree to 5.

(c) Implement *move ordering* heuristic. Could you observe any reduction in the total recursive function calls? Or reduction in average time (i.e. average duration of the game) needed to beat the Myopic player?

(d) Does increasing the cut-off depth to 5 increase the frequency of winning? Does the average number of moves before a win decrease?

Create a report for the above findings. The report will carry 7 marks.

The program you submit should read the state of the game from “testcase.csv” file and win against the Myopic player in **five** moves. The code for this is included in “ROLLXYZ_FIRSTNAME.py” file. Uncomment RunTestCase() function in “ROLLXYZ_FIRSTNAME.py” file and see the output. Testcase based evaluation will carry 8 marks.

You can use only the following five functions and one variable of the FourConnect class in your submitted program:

- MyopicPlayerAction()
- GameTreePlayerAction()
- PrintGameState()
- SetCurrentState()
- GetCurrentState()
- variable winner

You should **not** use any other functions or variables in the FourConnect class. During evaluation, we will be using a different FourConnect class in which the variables and functions (that you are not supposed to use) will be renamed.

Read the comments in “ROLLXYZ_FIRSTNAME.py” and run the program to understand what is expected.

Instructions for submission

- You must submit a single program file with the name “ROLLXYZ_FIRSTNAME.py”. Your program should read the “testcase.csv” file and start playing against the Myopic player. The game tree player (Player 2) must win in five moves to pass the testcase. You must add comments so that your code is readable. You can remove all the comments that were initially present in the file.

- Your report must be named "ROLLXYZ_FIRSTNAME.pdf". The report must contains details of the choices you made for parts (a) to (d), and your findings. You can use graphs and/or tables to report the number of wins and average number of moves before winning. In the report, you can also include other ideas that you have explored.
- Please use only capital letters for the three file names. Eg. 2020H1030999G_ADARSH.py, 2020H1030999G_ADARSH.dat and 2020H1030999G_ADARSH.pdf.
- Submit **only** the two files mentioned above. **Don't** zip the files. The assignment submission will be through quanta.
- Let the IC know if you notice some bugs/problems with hte program files that are provided.