**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**

Batch No. :

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS**

**Artificial Intelligence (BITS F444/ CS F407)**

**I Semester 2017-18**

**Programming Assignment-2**

**Coding Details**

**(October 3, 2017)**

*Instruction: Type the details precisely and neatly*

1. ID \_\_\_\_\_2014A7PS0095P\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name \_\_\_\_\_\_Lakshit Bhutani\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Mention the names of Submitted files :
   1. driver.py
   2. game.py
   3. util.py
   4. 2014A7PS0095P.docx
2. Total number of submitted files: \_\_\_\_4\_\_\_\_\_
3. Name of the folder :\_\_\_\_2014A7PS0095P\_\_\_\_\_\_\_\_\_
4. Have you checked that all the files you are submitting have your name in the top?(**yes/**no) : YES
5. Have you checked that all the files you are submitting are in the folder as specified in 4 (and no subfolder exists)?(**yes**/no) : YES
6. Problem formulation
   1. State representation:

Object of a class containing variables state [4X4 matrix representing the board] marked with separate symbols for human, machine tiles and the empty tiles.

* 1. Pseudo code of your successor function

*succ = []*

*for j in 1..4:*

*for i in 1..4:*

*if state[i][j] is empty:*

*succ.append((i, j))*

*break*

* 1. Terminal states generation process

Terminal state is generated if either the machine or the human wins the game of the game is drawn. Each of these is checked by the *terminal\_test* function

* 1. Data structure to store terminal states

The data structure is the same to store all states. It is a dictionary indexed by the state and stores the tuple *(utility\_value, next\_move)* of the state. Here utility\_value is that of the state and next\_move is the move to be taken next by the machine if it reaches it. This value is None if the game terminates on the given state.

* 1. Method to access terminal states and corresponding utility values

All states are indexed into a dictionary which stores the tuple *(utility\_value, next\_move)* of the corresponding state.

1. Minimax Technique details
   1. Node structure:

Object of a class containing variables state [4X4 matrix representing the board] marked with separate symbols for human, machine marked tiles and the empty tiles.

* 1. Method to ensure the correctness of terminal test (describe in maximum 4 lines)

The terminal test checks for either a win or a draw. Win is detected by checking identical symbol (human/machine) in horizontal, vertical and diagonal directions in groups of three. A tie is detected if the entire grid is filled (no empty tiles ) and there is no win situation.

* 1. Total number of nodes generated to play one game: 42175 (distinct) 7262072 (overall)
  2. Write the statistics here as asked

R1 = 7262072 R2 = 64 R3 = 16

R4 = 18.76 R5= 0.0019

* 1. Code status (**implemented fully**/ partially/ not done) : Implemented Fully

1. Alpha Beta technique details:
   1. Explain the logic used for pruning (in maximum four lines)

Initially alpha = -1000 and beta = 1000. At each MAX node if the MAX\_VALUE so far is greater than or equal to beta, the node is not expanded further or else the value of alpha is updated to the maximum of alpha and the MAX\_VALUE so far. Similarly at MIN node, the expansion is pruned as soon as MIN\_VALUE becomes less than or equal to alpha or else beta is updated to minimum of beta and the MIN\_VALUE so far.

* 1. Total number of nodes generated to play one game : 8028
  2. Write the statistics here as asked

R6 = 8028 R7 = 99.89% R8 = 21.77

1. Code status (**implemented fully**/ partially/ not done) : Implemented fully

1. Comparative analysis

Fill in the following information based of 10 independent games

|  |  |  |
| --- | --- | --- |
|  | Minimax Algorithm | Alpha Beta Pruning |
| Average number of nodes created | 7262072 | 8028 |
| Average time taken [to make first move] | 5.039 | 0.775 |
| Number of times machine wins (player M) | 10 | 10 |

1. GUI details
   1. Created the GUI (**yes**/ No): YES
   2. Have created it according to the specifications?(**yes**/No) YES
   3. Which module of Python is used for creating graphics? *turtle*
   4. Is this under the standard Python library or not? Under standard Python library
   5. If not, why? NA
2. Graphics details:
   1. Is turtle graphics working fine for displaying the board and coins? YES
   2. How have you calibrated the board and accepted human input to play the game? Human input is accepted by click at one of the valid tiles. Calibration of the click is done on the basis of turtle window’s origin and relative position of the click.
   3. How are you showing the base line? Drawing a red line above the board with text ‘Base line’ written on tip of it.
   4. How are you showing the move of the machine? Machine move are green coloured circles in one of the valid tiles.
   5. How are you showing the move of the human player? Human moves are blue coloured circles marked by the click on one of the valid tiles.

1. Compilation Details:
   1. Code Compiles (**Yes**/ No):\_\_YES\_\_\_\_\_\_\_\_
   2. Mention the .py files that do not compile:\_\_\_\_\_\_\_\_None\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   3. Any specific function that does not compile:\_\_\_\_\_\_\_None\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   4. Ensured the compatibility of your code with the specified Python version(**yes**/no)\_\_\_\_YES\_\_\_\_\_
   5. Instructions for compilation of your files mentioning the multi file compilation process used by you (We may use the replica of these for compiling your files while evaluating your code) *python driver.py*
2. Driver Details: Does it take care of the options specified earlier(yes/no):\_\_\_YES\_\_\_\_
3. Execution status (describe in maximum 2 lines)

Code is completely functional. Both minimax and alpha-beta pruning algorithms are generating correct moves and the machine is winning in all games played so far.

1. Declaration: I, \_\_\_\_\_\_Lakshit Bhutani\_\_\_\_\_\_\_\_ (name) declare that I have put my genuine efforts in creating the python code for the given programming assignment and have submitted only the code developed by me. I have not copied any piece of code from any source. If the code is found plagiarized in any form or degree, I understand that a disciplinary action as per the institute rules will be taken against me and I will accept the penalty as decided by the department of Computer Science and Information Systems, BITS, Pilani.

ID\_\_\_\_\_2014A7PS0095P\_\_\_\_\_\_\_\_\_\_\_ Name:\_\_\_Lakshit Bhutani\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_03/10/2017\_\_\_\_\_

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Should not exceed three pages