

PROGRAMMING ASSIGNMENT 2

Subject : GoPY Board Game

Teaching Assistant : Bahar Gezici

Submission Date : 15.11.2019

Due Date : 29.11.2019 23:59

[Click here to accept your Programming Assignment 2.](#)



1 Introduction

In this assignment, you are expected to implement a simplified "GoPy Board" game using Python programming language and to get familiar with control flows, collections and functions. The software that you will implement is going to fulfill some basic operations that are included in similar board games like Gomoku, Yonmoku etc.

2 Assignment

Gomoku game is a board game for two players who take turns in putting black and white stones on the board. Each players' goal is to create an unbroken row of five stones horizontally, vertically, or diagonally [1]. In this experiment, you are expected to implement a small and similar version of Gomoku game called GoPy Board, in Python programming language.

GoPy Board is an abstract strategy board game, which is played with 'X' and 'O' Go stones on a dynamic sized Go board. The first player starts the play by placing a 'X' stone on the board, and then the second player makes his move by placing an 'O' stone on an empty cell in the board. This process then alternates. The winner is the first player to obtain an unbroken row of stones that is considered as length of the board size of same stone horizontally, vertically, or diagonally.

Simply, your program will take players' moves as inputs and then displays the game board after each turn. The game specifications are as follows:

- The game starts out with an empty game board.
- The size of game board corresponds to a dynamic matrix. User will give the size of the board as an argument at the beginning of the game.

- There are two players and name of the players are fix: Player 1 and Player 2.
- Player 1 (the first player to move) will always be X and player 2 (the second player) will always be O.
- Dealing with handling user input is another logical step. When a player (say player 1, who is X) wants to place a X on the screen, they cant just click on a terminal. So we are going to approximate this clicking simply by asking the user for a coordinate of where they want to place their piece.
- The player 1 plays first, and the players alternate in placing a stone of their own on an empty cell on the board.
- The purpose of each player is to place an unbroken row of stones of their own horizontally, vertically or diagonally. When unbroken row of stones is reached by the length of board size, the relevant player will win the game.
- At each move of "GoPy Board" game, the player give the cell values (0,1,2 up to board size) as an input.
- You should draw the current status of the board after each turn.
- When game ends you should print " Winner ? ", with ? denoting the name of the winner, 'X' or 'O'.
- If you enter a value that isnt between 0 and size of the board, you should give an error message like this: " Please enter a valid number".

A sample output of program is provided below:

```
[bahargezici@rdev ~]$ python3 assignment2.py
What Size Game GoPy?3
 0 1 2
 3 4 5
 6 7 8
Player 1 turn--> 2
 0 1 X
 3 4 5
 6 7 8
Player 2 turn--> 1
 0 0 X
 3 4 5
 6 7 8
Player 1 turn--> 5
 0 0 X
 3 4 X
 6 7 8
Player 2 turn--> 4
 0 0 X
 3 0 X
 6 7 8
Player 1 turn--> 8
 0 0 X
 3 0 X
 6 7 X
Winner: X
```

- You should also add a function to check whether the availability of a square. If you put your stone to the full of cell which was selected by another player, you should give an error message like this: "The other player select this cell before." If you put your stone to the place that you have already selected, you should give an error message like this: " You have made this choice before".

```
What Size Game GoPy?4
0 1 2 3
4 5 6 7
8 9 10 11
12 13 14 15
Player 1 turn--> 2
0 1 X 3
4 5 6 7
8 9 10 11
12 13 14 15
Player 2 turn--> 3
0 1 X 0
4 5 6 7
8 9 10 11
12 13 14 15
Player 1 turn--> 6
0 1 X 0
4 5 X 7
8 9 10 11
12 13 14 15
Player 2 turn--> 3
You have made this choice before
0 1 X 0
4 5 X 7
8 9 10 11
12 13 14 15
Player 1 turn--> 10
0 1 X 0
4 5 X 7
8 9 X 11
12 13 14 15
Player 2 turn--> 2
The other player select this cell before
0 1 X 0
4 5 X 7
8 9 X 11
12 13 14 15
Player 1 turn--> 14
0 1 X 0
4 5 X 7
8 9 X 11
12 13 X 15
Winner: X
```

3 Grading Policy

Task	Point
Submitted	1
Compiled	10
Dynamic board size	30
Output	49
Error handling	10
Total	100

4 Important Notes

- Do not miss the submission deadline! Use the submit system. Submissions as e-mail attachments will not be accepted.
- Save all your work until the assignment is graded.
- Your assignment must be original and you must do it with your own individual work. Duplicate or very similar assignments are both going to be considered as cheating. You can ask your questions via Piazza and you are supposed to be aware of everything discussed on Piazza. At Piazza, you cannot share your source code completely or partially. Assignments will be checked for similarity, and there will be serious consequences if plagiarism is detected.
- Your programs will be executed in DEV machine, please make it work on dev before submitting. For example, you should use

```
python3 assignment2.py
```

syntax to execute your first program in DEV machine. If you use python assignment2.py syntax, DEV machine executes your program according to python 2.7.5.
- It is your duty to check the Piazza platform against any possible update about this assignment. If any instruction written by the TA violates any condition against this document, the new instruction(s) on Piazza is/are valid!
- You will submit your work as stated below:
 - assignment2.py
- You will lose points if you don't follow the rules defined above.

References

- [1] Gomoku game, <http://www.vcpr.cz/en/help-and-rules/gomoku-rules/>.