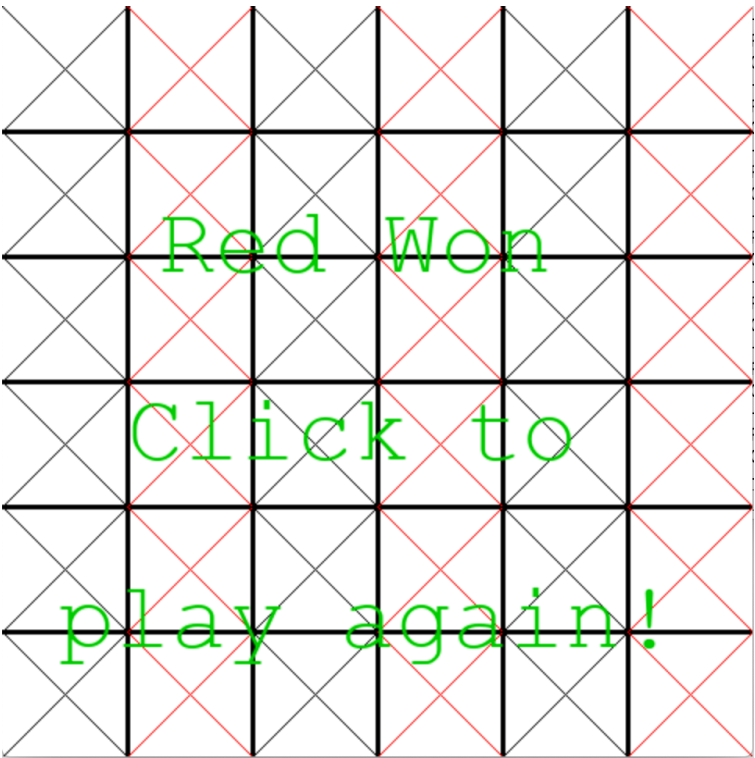# CS4386 Assignment 1

1 Game

There are two players in this game. In each turn, a player can put an 'X' in the empty cell of the 6*6 board. Any player who can connect 3 or 6 adjacent 'X' (color does not matter) in a row/column (**no diagonal**) will get 3 or 6 points respectively. The game ends when all the cells are taken by players and the player who gets higher points is the winner.

Case 1-3 illustrate the rules of this game (Player 1 is denoted by black and Player 2 is denoted by red).

<mark>Case 1:</mark>

Step i	Step i+1
X X ⬚ ⬚ ⬚ ⬚ (6×6 grid)	X X X ⬚ ⬚ ⬚ (6×6 grid)

In this case, player 1 will get 3 points in step i+1 since he connects 3 adjacent 'X'.

Step i	Step i+1
X X X X X	X X X X X **X**

In this case, player 2 will get 6 points in step i+1 since he connects 6 adjacent 'X'.

Step i	Step i+1
X X / X / X	X X **X** / X / X

In this case, player 1 will get 3+3=6 points in step i+1 since he connects 3 adjacent 'X' both in row and column.

2 Demo codes

Link

Github Link: https://github.com/sheilaya/CS4386_assignment1

Note: We will keep improving the demo codes and release the updated ones on github, which will not be much different from this version. Hence you can start implementing your AI based on this version **now**.

Meanwhile, if you find any bugs in our demo codes, you can ask questions on Github issues and we will fix them.

How to Run

To run the demo codes, you need to install python3.6 first.

We support 3 languages for implement your AI: C++, JAVA and Python. Python is the easiest one to implement since our server program is written in Python. Hence we **highly recommend** you to use Python. If you are not familiar with Python, you can use Java or C++.

1. If your AI is implemented by C++:

```
cd cpp
g++ AIPlayer.cpp -fPIC -shared -o aiplayer.so
cd ..
#if you want the Human play first, use
python3 game.py Human CPP 1
#if you want the AI play first, use:
python3 game.py Human CPP 2
```

2. If your AI is implemented by Java:

```
cd java
javac AIPlayer.java
mkdir com
mv AIPlayer.class com
jar cvf AIPlayer.jar com
cd ..
python3 game.py Human JAVA 1
```

3. If your AI is implemented by Python:

```
python3 game.py Human PYTHON 1
```

```
#if you want the JAVA AI play with PYTHON AI
python3 game.py JAVA PYTHON 1
#if you want the PYTHON AI play with PYTHON AI
python3 game.py PYTHON PYTHON 1
```

How to implement your AI algorithm

Take the python version for example, you need to modify the **get_move()** function in AIPlayer.py.

```
def get_move(self, state, player):
    # implement your algorithm here
    games = self.empty_cells(state)
    random_move=games[0]
```

```
      #the end of your algorithm
      return random_move
```

The example returns a random move, and you should modify it using your own algorithm. If your AI fails to return a move in a long time (e.g. 10 seconds per turn), your opponent will win the game.

You can define your own variables and functions in the AIPlayer Class, but you cannot delete existing variables or functions in AIPlayer Class. Meanwhile, you cannot change the function parameters of get_move() fuction.

Note: When implement your AI, you **MUST** not change game.py or gui.py.

Scoring Scheme and Requirements

Total (20 scores) = Performance of your AI + codes + report

1. **Requirement for the codes:**
   - At the beginning of your file, add the following comments to your code:

     ```
     ////////////////////////////////
     // CS4386 Semester B, 2021-2022
     // Assignment 1
     // Name: [Your name]
     // Student ID: [Your student ID]
     ////////////////////////////////
     ```

   - Rename the zip file that contains the related files of your **AI codes** as **CS4386_2122B_[studentID]_[name].zip**, where [studentID] is your student ID and [name] is your name. You need to submit this zip file to Canvas.

2. **Requirement for the report:**
   - You should write a report to explain your AI.
   - At the beginning of the report, include the following information:
     - The heading "CS4386 Assignment 1 (Semester B, 2021-2022)"
     - Your name
     - Your student ID
   - You should then describe your algorithm as clearly as possible. Feel free to use examples and add screenshots or other figures if it can help better illustrate your method.
   - **If you adopt some part of your code from somewhere, you must fully acknowledge this and provide a reference to where you obtain the code. You must declare how much of your submitted code is obtained from someone/somewhere else and how much is indeed written by you.**

- At the end of your report, include the related references from where you have gathered useful information in working on your assignment
- Convert your report to a PDF file with the file name **CS4386_2122B_[studentID].pdf**, where [studentID] is your student ID.
- You need to submit this report together with your source code to Canvas.

3. **Submission**

   Submit both **CS4386_2122B_[studentID]_[name].zip** and **CS4386_2122B_[studentID].pdf** to canvas.

Q&A

We have created a discussion on CANVAS. You can ask questions there and the TAs will check it everyday and answer your questions as soon as possible.

˅ **Discussions**			*Ordered by Recent Activity*
	Discussion of Assignment1		✅ 🔖 ⋮
⠿	1 Section		
		Available until Apr 2 at 23:59pm	

Deadline

The DDL for this assignment is **19th March**.