**AI Assignment 2 Report**

***Prepared by: Ibrahim Khalaff 3495***

***Youssef Sherif***

***Mohamed Tarek***

**Objective:**

To create a simpler version of risk game with GUI using only 2 players: either a human agent against a computer agent or two computer agents against each other in a simulation.

In this report, we will describe the function of each class and file we wrote in order to create this game.

**Color.py**

In this class, we establish that any territory in Grey means that neither player has taken nor attacked this territory. Otherwise, whether it’s Blue or Red, it means that one of the players has marked this territory as their own.

**Territory.py**

In this class, we establish each territory’s initial settings. Among those are its color (initially Grey before the game starts), its coordinates (x, y), its troops and finally, its neighbors.

**Board.py**

In this class, we import the previous two classes to help create the board. In this game, we use Egypt and USA as our main lands to conquer (as opposed to the original game, where it’s for the globe). We create files for both countries to mark their territories with their coordinates to set their limits. The files also include each territory’s neighbors. We then open them in the class methods (init\_egypt) and (init\_usa) to create the board and start the game. We use the method (update) to mark the colors of the territories according to where each player places their armies on the board. (set\_starting\_armies) is for the player to place their armies on the board. Each player starts with 20 armies.

**Agent.py**

This is the parent class for all AI and non-AI agents. Imports Board and Color. Each player starts with 20 armies. The method (receive\_armies) is for every player to receive new armies with each turn. The number of armies received depends on the number of territories the players have. We divide them by 3 and the result is the number of armies to receive. If it’s less than 3, then the player receives 3 armies by default.

**HumanAgent.py**

Imports Color and Agent. Employs methods (place\_territories) to help the agent place his armies on all the territories they want to start with and (attack) in order to take more territories with attacking their neighbors.

**PassiveAgent.py**

Imports Agent and Color. Employs methods (place\_territories) same as human agent, (make\_decision) to see where it places its armies on the board and (find\_territory\_with\_fewest\_armies) to attack these territories.

**AggressiveAgent.py**

Imports Agent and Color. Employs the same methods as the other two agents. Along with (find\_weakest\_neighbour\_to) to find the weakest neighbor to attack and the method (find\_territory\_with\_most\_armies) which returns the territory number with the most armies.

**AgentFactory.py**

This class creates agents for the game.

**Game.py**

Employs the class methods (start\_simulation\_mode) where two AI agents play against each other, (start\_playing\_mode) where a human agent goes against an AI agent and finally, (alternate\_turn) to interchange turns between Blue and Red.