HW 5: Stackelberg Security Game Tournament
DUE: Wed Nov 9

You may (optionally) work in a group of two on this assignment.

In this assignment you will design agents to play a simple version of a security game. A Stackelberg Game is a special type of game named for German economist Heinrich Freiherr von Stackelberg and involves two player games with one leader agent and one follower agent. Consider the Divide and Choose game where the cutter (leader) decides on how to split a cake in two and the chooser (follower) decides on which piece is his. The cutter makes a choice knowing that the chooser will capitalize on any advantage and therefore tries to equalizes the pieces.

This concept can be applied to a game of security. There are two agents a defender (leader) and an attacker (follower). The defender's action is to protect various targets by allocating resources however there are always more targets than there are resources. The follower can see these targets and knows how they have been covered then decides on which target to attack. Consider the following game:

Defender \ Attacker	1	2
1	1 , 2	0 , 5
2	1 , 4	9 , 1

Assume the Defender has 1 resource to allocate and for simplicity let us say he decides to apply the resource fully to one target instead of dividing it among the two targets. If the agent decides to protect Target 1 then a rational Attacker will respond by attacking Target 2 where the outcome is (0,5). Should the defender decide to fully cover Target 2 then a rational attacker would decide to attack Target 1 for the outcome (1,4). If the defender assumes he is playing against a rational attacker then to maximize his own utility between the outcomes (0,5) and (1,4) he will prefer outcome (1,4) and would therefore cover Target 2. This backwards induction process for determining the leader's best response strategy to a follower who hasn't made a move yet is a good way to find the Subgame Perfect Nash Equilibrium of a game and gives insight into how to find Stackelberg equilibrium. However, if the attacker is not rational and instead is trying to punish the defender he might decide to simply attack Target 1 because the maximum possible defender payoff is 1 instead of a possible 9 if he attacked Target 2.

Now consider the same game but this time the Defender only has 0.5 total resources. Notice that we have been very careful to refer to terms like coverage and resources. This is to differentiate from a mixed strategy which are probabilities for each action that sum to 1. Whereas the a fully covered target has a value of 1, an uncovered target a value of 0, and any other level of coverage so long as the total coverage does not exceed the total number of resources. This also makes it hard to represent utility in security games using a normal form matrix. That is why we usually represent it using the following notation where θ means defender and ψ is attacker.

Target	U_θ^u	U_θ^c	U_ψ^u	U_ψ^c
1	0	1	4	2
2	1	9	5	1

s. t. $U_\theta^u(t) < U_\theta^c(t),\ U_\psi^u(t) > U_\psi^c(t), \sum c \leq m, U(t) = c(t)\big(U^c(t)\big) + \big(1 - c(t)\big)\big(U^u(t)\big)$

Assignment:

- Develop two (2) different defender and attacker strategies using the tournament code we provide.
- Write a brief report documenting, explaining, and analyzing your agents and their performance.
- Submit one defender and one attacker agent (they will be in a **single** class file) which will be pitted against your classmates in a round-robin style tournament. **Make sure your name is in a comment in your submitted code**.
- Bonuses will be given to the defender agent with the highest overall utility, as well as the defender agent with the minimax regret. We will also give a small bonus for attacker agents that achieve the maximum possible utility, or minimize the defender payoff (maximizing punishment).

All of the code, documentation, and other resources you will need are in the following gibhub repository:

https://github.com/osveliz/SG