Background

An interesting field of study is understanding how cooperation can exist or arise in societies composed of individuals who are generally selfish in their motivations. Can selfish motivations and cooperation co-exist? Under what conditions? How can we understand this? This was the problem the political scientist Robert Axelrod faced in the 1970s. However, he was also familiar with an increasingly widespread technology called a "computer" which he thought might be able to help answer these questions.

His idea was to model competition and cooperation in society using the combination of Game Theory and computers. He reached out to his friends in the emerging community surrounding computing (we'd probably call them hackers today) and asked them to play a game. The rules were simple: you'd write a program to play a simple game of cooperation and competition in a tournament. Your competitors would be all of the other strategies submitted. His intuition was that repeated interaction might lead to the emergence of stable, cooperative behaviour - after all, if we interact repeatedly, we might learn how to be "friends".

The idea caught on! It was clever, interesting, and kind of fun to think about. Dozens of people took part: economists, political scientists, computer programmers, mathematicians, military experts, and philosophers. They wrote all kinds of strategies - complicated, simple, elegant, and exhaustive. Axelrod collected them all, and ran the tournament. The result was surprising: the winner, by a wide margin, was incredibly simple. It was called TIT-FOR-TAT (or COPYCAT) and played nice initially, then simply copied the other player forever afterwards.

In this activity, we will study this theory, and explore how cooperation might emerge - even in situations where it theoretically "should not" exist.

Worksheet

Consider the following Prisoner's Dilemma depicted in Figure 1.

		Player 2: C	Player 2: D
Player 1	C	3,3	0,5
	D	5,0	1,1

Figure 1: The Prisoner's Dilemma

As part of this activity, we will be studying the classic experiment created by Robert Axelrod. As explained, in the experiment dozens of leading scientists, hobbyists, and military experts were asked to design computer programs to play a repeated prisoner's dilemma against one another. The programs were scored based on the number of points that were accumulated.

The rules of this repeated game are simple:

- The tournament is a round-robin, in which each strategy plays every other strategy exactly once (a *match*) in the game depicted in Figure 1.
- Each match consists of the two programs playing 100 repetitions of the Prisoner's dilemma against one another.
- Each match is scored based on the total number of points earned during the match.
- The tournament overall is scored by the total number of points earned across all matches by a strategy.

The goal was simple: get the most points you can. Design the best possible strategy. Out-think, out-strategize, and out-play the other players to maximize your total tournament points.

A good interactive example which illustrates many of these ideas was created by developer Nicky Case and can be found online at:

`http://ncase.me/trust/`

One of the most interesting things about this experiment was that the SPNE strategy did not perform the best. The reasons for this are complicated, and the game above might give you some intuition. The questions on this assignment will walk you through the logic, from a strategic point of view.

Project AXLRD

This isn't just a thought exercise. We're going to do it for real!

- You (and all your classmates) will work in small groups to design a strategy to play in our version of Axelrod's tournament. Your job is to design a strategy to win!
- Fortunately, we have a new tool to help you do this: Project AXLRD. This is a web-based interface which allows you to build strategies using "blocks" (kind of like Scratch or Lego MindStorms). You can find Project AXLRD here:
- There is a tutorial to help you understand the interface. You can also save, import, and export different strategies to test our how your strategy performs.

  - Remember: keep your strategy top secret! Don't add any names or identifying information to your strategies!

- There are also a number of tests to help you ensure your "strategy" is actually a strategy, and you haven't missed any cases.
- Using Project AXLRD, design a strategy to win the tournament! After you're done, save your final strategy (as a .STRAT file) and submit it along with this assignment, so we can see who wins!

With your group, complete the following worksheet, writing down your thoughts and answers as you go. This will help you to design your strategy, as well as understand the ideas developed in class.

Questions

1. (5 points) Consider the repeated version of the game in Figure 1, as carried out in the experiment. What is the Subgame Perfect Nash Equilibrium of this game? Explain.
2. (5 points) Using the AXLRD interfrace, build a strategy which implements the SPNE strategy (call it SPNESTRAT), and add it to a tournament. Take a screenshot of your strategy, and insert it into the answer to this question.
3. (5 points) Now, do the same but for the strategy COPYCAT. Which one is more complicated?
4. (5 points) Before you run the tournament, predict which strategy will do the best. Why? Run a tournament where your two strategies play against one another and see! Were you right?
5. (5 points) Suppose Player 1 believes that Player 2 is playing COPYCAT. Using ALXRD, design a change to COPYCAT which does better for Player 1. What are the payoffs? Explain carefully. Call the strategy you have found BESTRESPONSE1.
6. (5 points) Repeat (2) but for BESTRESPONSE1. Predict the results, then run two tournaments where BESTRESPONSE1 plays against (i) COPYCAT and (ii) SPNESTRAT. What are the payoffs? Is this what you predicted?
7. (5 points) Now, run ONE tournament with all three strategies. Which strategy wins? Can you explain why?
8. (5 points) Duplicate COPY-CAT several times, and repeat the previous question. Do your answers change? Explain
9. (5 points) Think carefully about how different strategies perform against different sets of opponents. What does this say about beliefs? Can this explain why you might not want to play the subgame-perfect strategy?

Tournament Time!

So, you've had a chance to practice against a fictional opponent. Now, it's time to do it for real. Your strategy will play against strategies submitted by *all of your classmates*. Design a strategy to win the whole tournament, taking into account what you learned (above).

Create your strategy using AXLRD, and take a picture. Save it as a .STRAT file, for submission later.

10. (10 points (bonus)) In the strategy that you designed above, explain how it is *supposed* to work, as clearly as you can, and justify why you designed your strategy in the way you did? What did you think about, test, or consider?

11. (10 points (bonus)) Test your strategies against some hypothetical opponents. How does it perform? Which opponents did you choose (and why)? Do you need to make any adjustments?

12. (20 points (bonus)) To provide incentives, we will award you extra bonus points, depending on how well you do! If your strategy earns a score of x and the best strategy earns y, you will get a bonus of $20 \cdot \frac{x}{y}$ points! So design your strategy well!