CS 357: Algorithmic Game Theory	**Spring 2022**
Assignment 8 (due 04/20/2022)	
Instructor: Shikha Singh	*Solution Template*

Instructions. This is a single-person assignment. Points will be awarded for clarity, correctness and completeness. The assignment is due **at 11 pm EST** on the due date.

Note that this assignment is intentionally shorter than a typical assignment and due a day early so that you can spend the remainder of the time preparing your project proposal (due Friday, Apr 22 by 5 pm).

Topics and Readings. Sequential and Repeated Games: PS Ch 4.

Extensive-form Games

Problem 1. (Short-answer question) Consider the extensive-form game in Figure 1.

(a) Identify all (pure) Nash equilibria of the game. Which of the Nash equilibria is implausible and is rather sustained by an *empty threat*?

(b) Identify the subgame-perfect equilibrium of the game.

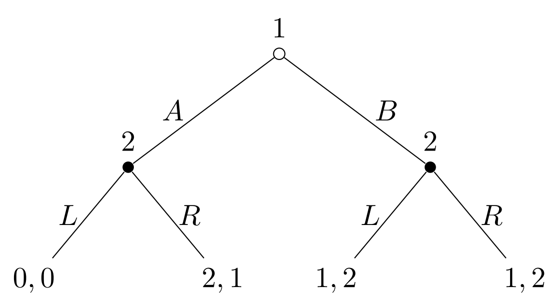


Figure 1: Extensive-form game for Problem 1

Problem 2. (Alternate Bargaining Game) In lecture, we considered the one step bargaining game where player 1 proposes a split and player 2 must accept or reject. We can imagine an alternating bargaining game where player 2 can either accept or make a counter-offer to player 1. We consider such a game in this problem.

Suppose we have two players and they are trying to divide a \$1 between them through a two-round bargaining game. We assume both players want to maximize the amount of money they get but they both dislike delay, thus for every round of bargaining their utility goes down by a factor $\delta \in (0, 1)$. In the first round, player 1 proposes a split $(x, 1 - x)$, where $x \in [0, 1]$, which player 2 can accept or counter. If player 2 accepts, the game ends and the players each get their respective share. Otherwise, player 2 proposes a counter split $(y, 1 - y)$, where $y \in [0, 1]$. Finally, player 1 can either accept or reject. If player 1 accepts, the game ends and they each get their respective share (but each discounted by δ), if player 1 rejects, the game ends and both players get 0.

Formalize this as an extensive-form game (by sketching out the game tree)[1] and apply backward induction to find the unique subgame-perfect equilibrium of the game.

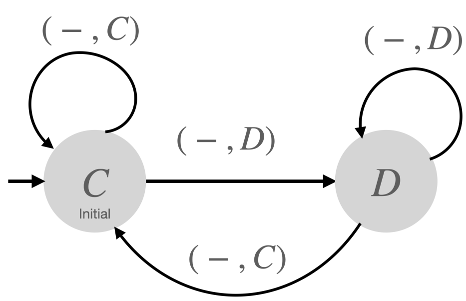
Problem 3. (Tit-for-Tat and Repeated Prisoner's Dilemma)

(a) Recall the payoff matrix of Prisoner's Dilemma from lecture.

	C	D
C	$2, 2$	$-1, 3$
D	$3, -1$	$0, 0$

Recall the Tit-for-Tat strategy: at stage 1, cooperate; at stage i, do whatever the other player did in stage $i - 1$. Prove that the Tit-for-Tat strategy never wins a head-to-head match: no matter what strategy Beth uses, if Aamir uses Tit-for-Tat, then Aamir's total payoff is at most that of Beth's.

(b) Show that the Tit-for-Tat strategy represented as the automaton below is not a symmetric subgame-perfect Nash equilibrium for all $0 < \delta < 1$ (except for the degenerate case when $\delta = 1/3$).



More on Iterated Prisoner's Dilemma. Check out this simulation of various strategies in iterated PD and how they evolve (especially when there may be errors in executing a strategy): Evolution of Trust[2]. Python's Axelrod library provides tools for running your own prisoner's dilemma simulations.

[1] The game tree for this game has an unbounded branching factor (for all possible "splits") so can be hard to draw with Tikz; attaching a hand drawn sketch is sufficient.

[2] Credit: Thanks Eva Borton for sharing this!