Vivekanand Education Society's
Institute of Technology
Approved by AICTE & Affiliated to University of Mumbai

Artificial Intelligence and Data Science Department

Game Theory/Odd Sem 2023-23/Experiment 1

Name : Subrato Tapaswi	Class/Roll No. : D16AD/60	Grade :

Title of Experiment : Program to implement Prisoner's Dilemma

Objective of Experiment : To understand the concept of nash equilibrium, and perform a program to implement Prisoner's Dilemma, to understand application of nash equilibrium.

Outcome of Experiment : Participants will develop a solid understanding of the fundamental concepts of game theory, specifically the Prisoner's Dilemma. They will grasp the nuances of strategic decision-making, the tension between individual rationality and collective cooperation, will be a deeper understanding of game theory, enhanced programming skills, strategic thinking abilities, and insights into cooperation and decision-making dynamics.

Problem Statement : To write a python code to implement application of nash equilibrium on prisoner's dilemma.

Description / Theory :
Prisoner's Dilemma:
The typical prisoner's dilemma is set up in such a way that both parties choose to protect themselves at the expense of the other participant. As a result, both participants find themselves in a worse state than if they had cooperated with each other in the decision-making process.

The prisoner's dilemma presents a situation where two parties, separated and unable to communicate, must each choose between cooperating with the other or not. The highest reward for each party occurs when both parties choose to cooperate.

- A prisoner's dilemma describes a situation where, according to game theory, two players acting selfishly will ultimately result in a suboptimal choice for both.
- The prisoner's dilemma also shows us that mere cooperation is not always in one's best interests.
- A classic example of the prisoner's dilemma in the real world is encountered when two competitors are battling it out in the marketplace.
- In business, understanding the structure of certain decisions as prisoner's dilemmas can result in more favorable outcomes.
- This setup allows one to balance both competition and cooperation for mutual benefit.

The prisoner's dilemma scenario works as follows: Two suspects have been apprehended for a crime and are now in separate rooms in a police station, with no means of communicating with each other. The prosecutor has separately told them the following:

- If you confess and agree to testify against the other suspect, who does not confess, the charges against you will be dropped and you will go scot-free.
- If you do not confess but the other suspect does, you will be convicted and the prosecution will seek the maximum sentence of three years.
- If both of you confess, you will both be sentenced to two years in prison.
- If neither of you confesses, you will both be charged with misdemeanors and will be sentenced to one year in prison.

Possible outcomes of prisoner's dilemma:

	cooperate	defect
cooperate	(5, 5)	(0, 10)
defect	(10, 0)	(0.5, 0.5)

The dominant strategy here is for each player to defect (i.e., confess) since confessing would minimize the average length of time spent in prison. Here are the possible outcomes:

- If A and B cooperate and stay mum, both get one year in prison—as shown in the cell (a).
- If A confesses but B does not, A goes free and B gets three years—represented in the cell (b).
- If A does not confess but B confesses, A gets three years and B goes free—see cell (c).
- If A and B both confess, both get two years in prison—as the cell (d) shows.

Algorithm/ Pseudo Code / Flowchart (whichever is applicable):

Define the payoff matrix:
- R = Reward payoff
- T = Temptation payoff
- S = Sucker's payoff
- P = Punishment payoff

```
Calculate payoffs based on decisions:
  If Agent 1 and Agent 2 both cooperate:
    Agent 1 payoff = R
    Agent 2 payoff = R
  If Agent 1 betrays and Agent 2 cooperates:
    Agent 1 payoff = T
    Agent 2 payoff = S
  If Agent 1 cooperates and Agent 2 betrays:
    Agent 1 payoff = S
    Agent 2 payoff = T
  If Agent 1 and Agent 2 both betray:
    Agent 1 payoff = P
    Agent 2 payoff = P
```

End simulation

Calculate and display the cumulative payoffs for both agents

Game Theory/Odd Sem 2023-23/Experiment 1

Program :

1. **Programs on Basic programming constructs like branching and looping.**

```
def prisoner_dilemma(player1_choice, player2_choice):
    if player1_choice == "cooperate" and player2_choice == "cooperate":
        return 5, 5
    elif player1_choice == "defect" and player2_choice == "defect":
        return 0.5, 0.5
    elif player1_choice == "cooperate" and player2_choice == "defect":
        return 0, 10
    elif player1_choice == "defect" and player2_choice == "cooperate":
        return 10, 0
    else:
        raise ValueError("Invalid choices")
```

Output Screenshots :

```
Player 1, enter your choice (cooperate/defect): defect
Player 1 payoff(in years): 10
Player 2 payoff(in years): 0
Payoff Matrix:
           Cooperate   Defect
cooperate           (5, 5)   (0, 10)
defect     (10, 0)   (0.5, 0.5)
```

```
Nash Equilibria:
Player 1: [1. 0.], Player 2: [1. 0.]
Player 1: [0. 1.], Player 2: [0. 1.]
```

Results and Discussions :

The prisoner's dilemma shows us that mere cooperation is not always in one's best interests. In fact, when shopping for a big-ticket item such as a car, bargaining is the preferred course of action from the consumers' point of view. Otherwise, the car dealership may adopt a policy of inflexibility in price negotiations, maximizing its profits but resulting in consumers overpaying for their vehicles.