

Ain Shams  
Spring Semester 2023

**Doctor: Tamer Mostafa**

# Game Theory

## Documentation

---

Name: Haya Ahmed Hamed Shaker

ID: 20191703039

Name: Antony Nabil Naguib Youhanna

ID:20191703004

Name: Amira Barakat Ali

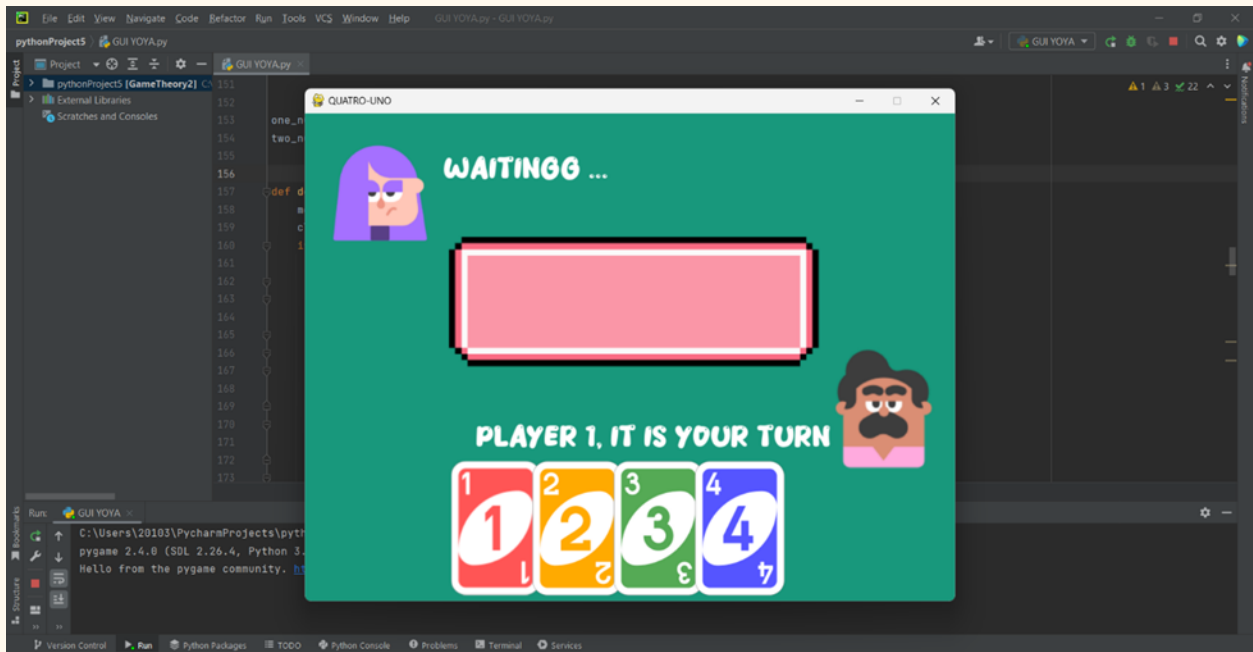
ID: 20191703002

# Game Theory

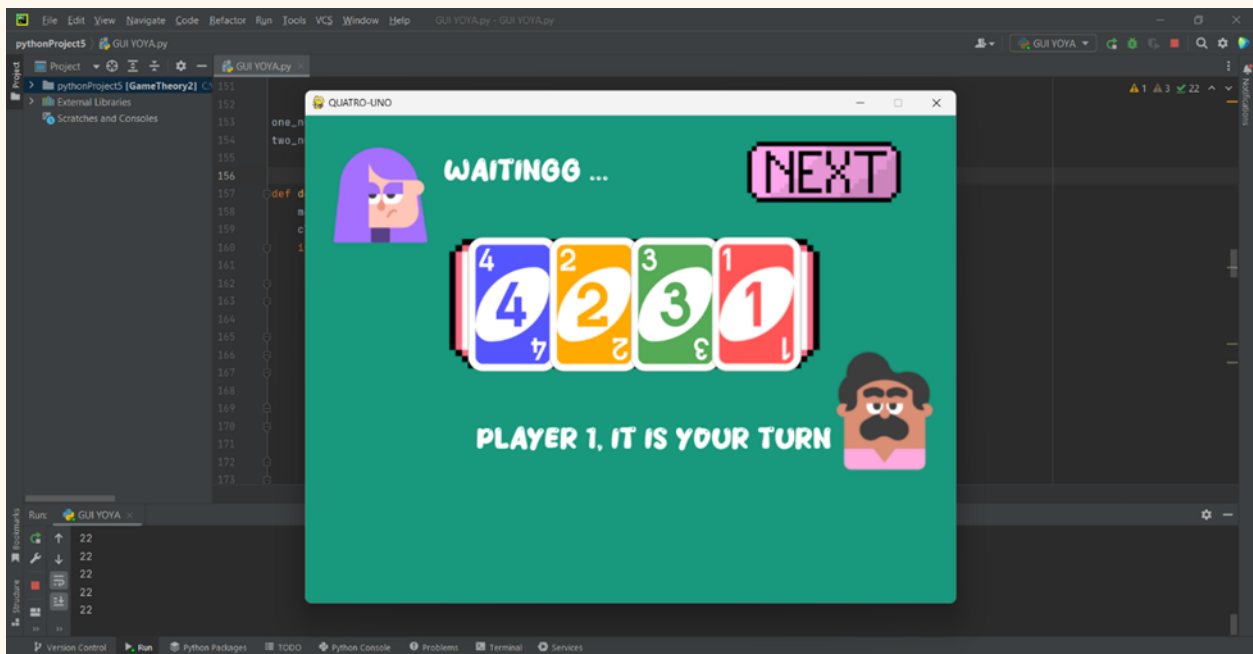
## Task 1:



We decided to use pygame for this project, starting with main menu, it presents two buttons, one for starting the game or exit, and since there are no buttons in pygame, therefore we created a function called “button”, and by clicking on it, it performs an action called in one of the parameters.

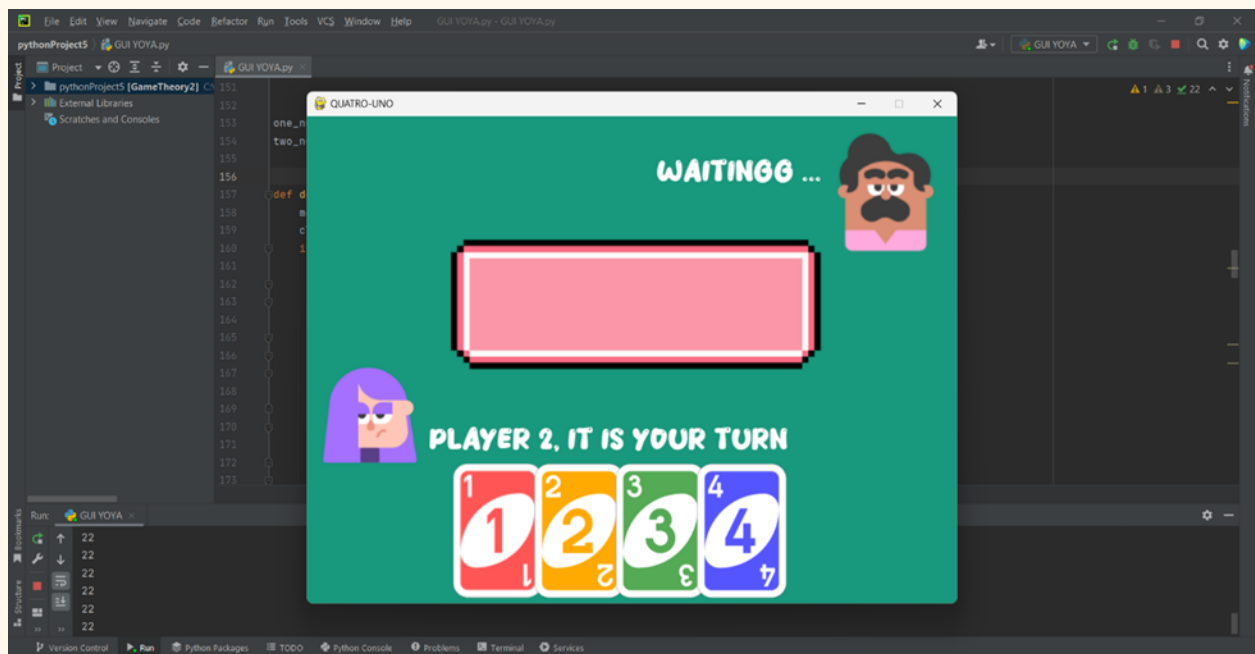


After clicking on “start” it performs the action “player 1’s turn”, and this page allows the first player to choose their card arrangement.



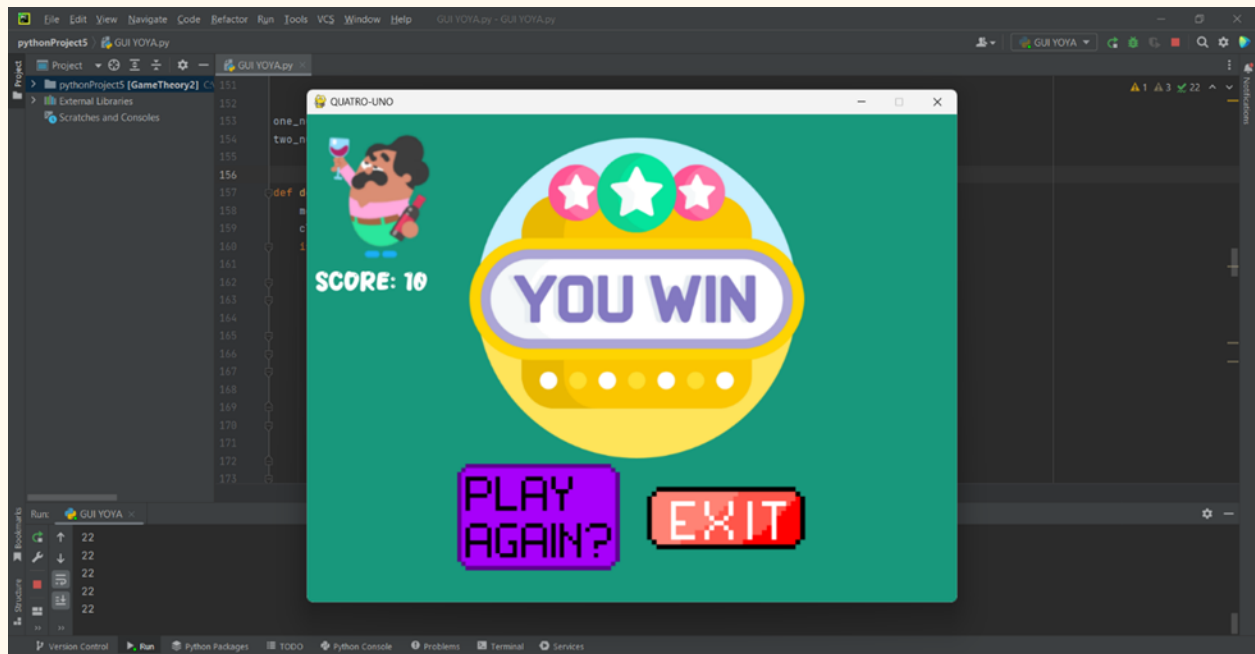
And displays it on the table, by using a list that saves the order of the cards and because we’ll need this list for other purposes later, and by pressing

the “next” button, the function for “player 2’s turn” will be called.

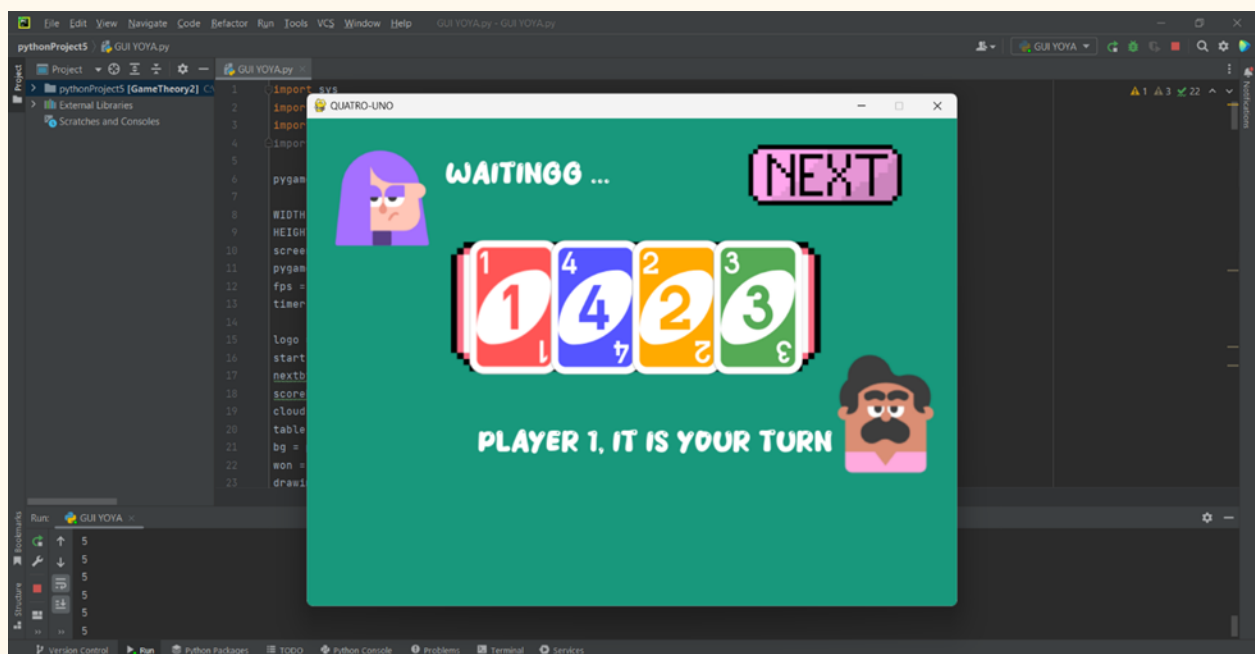


After player 2 chooses their arrangement, it's also saved in another list, and when clicking on “score” button, it calls the function that calculates the winner, set of if conditions that compares between the 2 lists for each

player.



In this case player 1 wins the game, this gives player 1 plus 10 points and it leaves the two players with 2 options, whether to play again or exit the game.






Another case where player 1 and player 2 play the same cards.



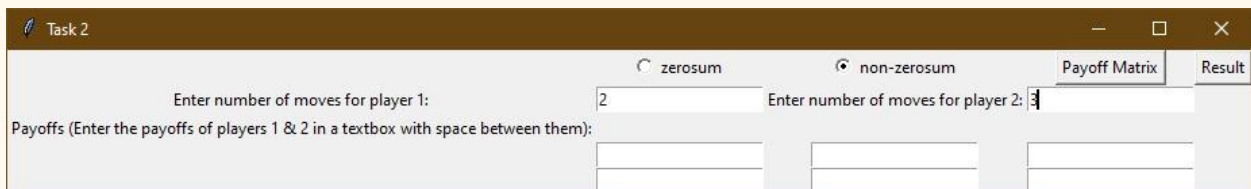
Players 1 and 2 both increase with the same 5 points.

# Task 2:

First, the user should choose the type of the game and number of moves for each player, then click on the payoff matrix button to enter the payoff matrix of the game.



After clicking on the payoff matrix button, the number of textboxes will be shown according to the number of moves entered for each player, if the game is non-zero sum, the user should enter the payoffs of every player in each textbox separating them with a space, otherwise, the user should only type the payoffs of the first player only, then the user should click on the results button.



The results will then appear showing the minmax move for each player with the guaranteed payoffs, the best response for each player against the move of the other, and the moves causing Nash equilibrium if exists.

**Task 2**

Enter number of moves for player 1:

Payoffs (Enter the payoffs of players 1 & 2 in a textbox with space between them):

	<input type="radio"/> zerosum	<input checked="" type="radio"/> non-zerosum	Payoff Matrix	Result
Minmax payoff for player 1:	<input type="text" value="7"/>			
Minmax payoff for player 2:	<input type="text" value="6"/>			
Player 2 best response for player 1 move	<input type="text" value="1"/>			
Player 2 best response for player 1 move	<input type="text" value="2"/>			
Player 1 best response for player 2 move	<input type="text" value="1"/>			
Player 1 best response for player 2 move	<input type="text" value="2"/>			
Player 1 best response for player 2 move	<input type="text" value="3"/>			
Nash equilibrium occurs when player 1 plays move	<input type="text" value="2"/>			

Enter number of moves for player 2:

when he/she plays move  
when he/she plays move

with payoff	6
with payoff	12
with payoff	7
with payoff	9
with payoff	11

and player 2 plays move



# Task 3:

The user plays against 5 opponents 25 rounds each by choosing to cooperate or defect.

The screenshot shows a window titled "Repeated Prisoner Dilemma". At the top, there is a section for selecting an opponent, with a dropdown menu currently set to "1". Below this is a payoff matrix table. The table has two columns for the opponent's choice ("Cooperate" and "Defect") and two rows for the user's choice ("Cooperate" and "Defect"). The payoffs are: (Cooperate, Cooperate) = 20, 20; (Cooperate, Defect) = 0, 30; (Defect, Cooperate) = 30, 0; (Defect, Defect) = 10, 10. Below the matrix, there are input fields for "Your Payoff" and "Opponent's Payoff". At the bottom, there are input fields for "Round" (set to 0), "Average:", and "Total Average:" for both the user and the opponent.

	Cooperate	Defect
Cooperate	20, 20	0, 30
Defect	30, 0	10, 10

Round: 0

Average:

Total Average:

For example, the user chose to cooperate, and the opponent (computer) chose to cooperate too.

Repeated Prisoner Dilemma

		Opponent	
		Cooperate	Defect
You	Cooperate	20, 20	0, 30
	Defect	30, 0	10, 10

	Your Payoff	Opponent's Payoff
Round	20	20
2		
Average:		
Total Average:		

Here the user chose to defect while the opponent chose to cooperate.

Repeated Prisoner Dilemma

		Opponent	
		Cooperate	Defect
You	Cooperate	20, 20	0, 30
	Defect	30, 0	10, 10

	Your Payoff	Opponent's Payoff
Round	30	0
3		
Average:		
Total Average:		

After playing 25 rounds with an opponent, the average payoffs are calculated and shown each for the player and the opponent.

The screenshot shows a software window titled "Repeated Prisoner Dilemma". It contains a payoff matrix and a summary table.

**Payoff Matrix:**

		Opponent	
		Cooperate	Defect
You	Cooperate	20, 20	0, 30
	Defect	30, 0	10, 10

**Summary Table:**

	Your Payoff	Opponent's Payoff
Round	20	20
Average:	14.8	22
Total Average:		

After playing with the 5 opponents 25 rounds each, average payoffs are calculated and shown for the player and all the opponents.

Repeated Prisoner Dilemma

Opponent

5

Cooperate

Defect

Cooperate

20, 20

0, 30

You

Defect

30, 0

10, 10

Your Payoff

Opponent's Payoff

Round

30

0

Average:

22.8

3.6

Total Average:

16.08

12.96