**World Quant University**

**Professor: Ivan Blanco**

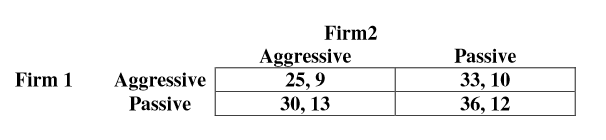
**Alpha Design I**

Nikolas Lippmann Pareschi - [nikolaslippmann@gmail.com](mailto:nikolaslippmann@gmail.com)

**Mini Project: Unit 1**

In Two firms are competing in an oligopolistic industry. Firm 1, the larger of the two firms, is contemplating its capacity strategy, which could be either “aggressive” or “passive”. The aggressive strategy involves a large increase in capacity aimed at increasing the firm’s market share, while the passive strategy involves no change in the firm’s capacity.

Firm 2, the smaller competitor, is also pondering its capacity expansion strategy; it will also choose between an aggressive strategy and a passive strategy. The table below shows the profits associated with each pair of choices.



1. If both firms decide their strategies simultaneously, what is the Nash equilibrium (or equilibria)?

The equilibrium will be Passive for firm 1 and Aggressive for Firm 2.

1. Is there any mixed strategy Nash equilibrium in this game?

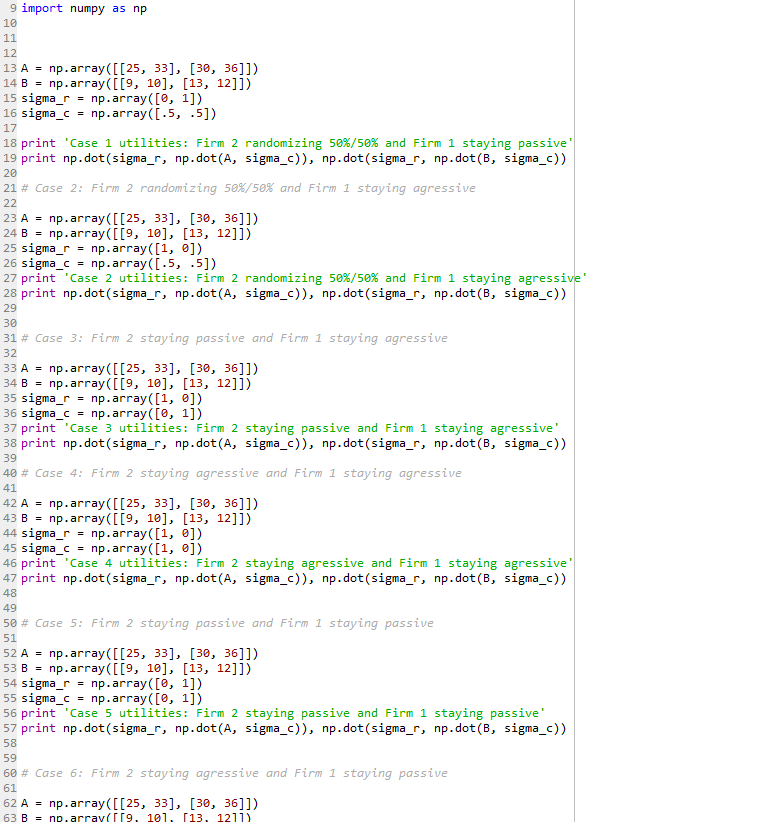
Yes, by theorem we have that a game with a finite set of players and a finite strategy sets has at least one (mixed) Nash Equilibrium. As the mixed strategy can have a weight of 100% in one strategy, the pure strategy is a case (subset) of the mixed strategies space.

1. If yes, what is the mixed strategy Nash equilibrium (or equilibria)?

The mixed strategy is the pure strategy. By definition every pure strategy is a “distorted” mixed strategy with a 100% weight.

1. Also, write a simple Python implementation of Game Theory to illustrate the implementation of Nash Equilibrium.

I wrote a simple program to compute the utilities in some cases. I also considered some mixed strategies to get a feel for the equlibirum.



1. Which Python Packages are best suited for this implementation? Would the Python Libraries on Linear Programming be suitable for this purpose?

Actually, I installed a package called nashpy and it is very interesting. In zero-sum games it can even compute the equilibrium for mixed strategies. I tried in our game but it didn’t work because it is not a zero-sum game. I cut off the code regarding the library because it only works in Python 3. Linear Programming would be useful to find optimal vectors to maximize utilities, as this problem is a linear one.

1. Research the Gambit software (open source) and show how it can be leveraged for the above implementation. (To download Gambit: <http://gambit.sourceforge.net/>)

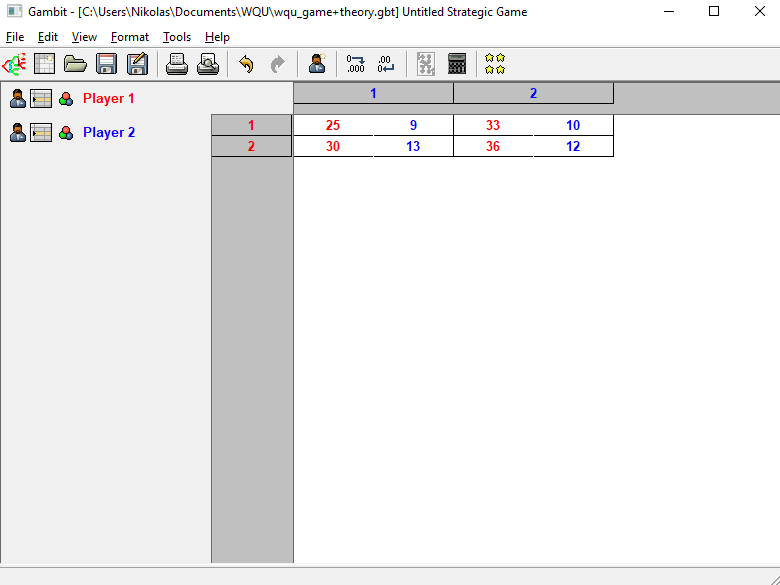
In this video we can see exactly how to use the Gambit software to solve our problem.

[https://www.youtube.com/watch?v=-1WZenmfi1I](https://www.youtube.com/watch?v=-1WZenmfi1I%20)

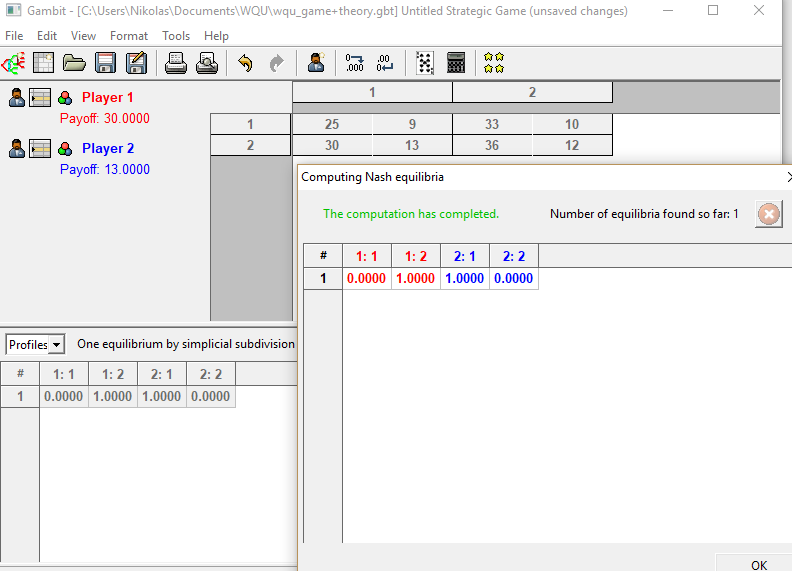
The steps are:

1. File -> New -> Strategic Game
2. Input the strategies for player 1 and 2
3. Go to tools, equlibirum.
4. Compute one nash Equlibirum

Execution:



Nash equilibrium:



PS: I was not able to download gambit software from the site above, instead, I was able to download from: <https://sourceforge.net/projects/gambit/files/>