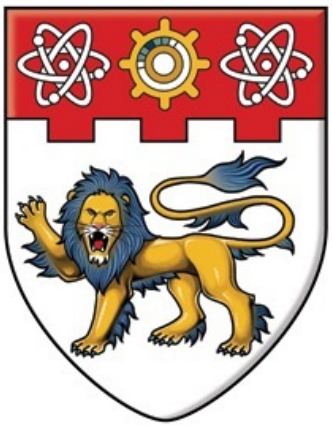

NANYANG TECHNOLOGICAL UNIVERSITY

COURSE CODE	**:**	**CZ4046**
COURSE NAME	**:**	**INTELLIGENT AGENTS**
NAME	**:**	**HTET NAING**
MATRICULATION NO.	**:**	**U1620683D**
COURSEWORK	**:**	**ASSIGNMENT 2**

Table of Contents

1) Agent Design

The design of our agent used in this assignment is inspired by one of the famous strategies stated in the book "The Art of War" by Sun Tzu. It says :

> *"Know your enemy and know yourself and in hundred battles, you will never be in peril."*

Figure 1: Quote extracted from The Art of War by Sun Tzu [1]

The agent will be designed based on the following rules. The order of the rules reflect their priorities:

1) "Do not be the first to defect."
2) "Play nasty for the last remaining few rounds."
3) "If they are nice, I will continue to be nice to them."
4) "If someone betrays me, I am triggered"
5) "By my default nature, I am nice."

1.1) Rule 1: "Do not be the first to defect"

According to Axelrod, a program is known to be "nice" if it starts by cooperating. It is observed that the outcome of the game indeed depends on whether the rule used by the program was nice or not. Although there is a risk that it may receive the sucker's payoff on the first round, this loss of utility is relatively small when compared to potential benefits that will be achieved in later rounds by means of mutual cooperation [2]. Hence, our agent will adopt this rule and start the game by being nice.

1.2) Rule 2: "Play nasty for the last remaining few rounds"

As one match will contain about 100 rounds, a threshold value of 95 is set for our agent to play *Nasty* from the 95th round onwards. This is to maximize the total payoff for our agent. It follows the game-theoretic analysis that demonstrates that "Always defecting" is the dominant strategy. As it is approaching to the end of the match, the duration of staying in mutual business together with the opponents become shorter. Hence, for maximal benefits, our agent is designed

to exploit the system by always defecting regardless of the opponent's behaviors as the match is approaching to the end.

1.3) Rule 3: "If they are nice, I will continue to be nice to them"

This follows the above-mentioned strategy extracted from "The Art of War" in Figure 1. Our agent not only knows its own behavior but also takes into account of the opponent's behaviors. If it is observed that both opponents are cooperative in the previous round, then our agent will continue to cooperate with them. This is because our agent recognizes the importance of mutual cooperation in maximizing its long term benefits.

1.4) Rule 4: "If someone betrays me, I am triggered"

If one of the opponents has ever betrayed our agent by defecting, then our agent will not tolerate such behavior. Our agent's trust on its partners is broken and it is no longer interested in staying in mutual cooperation with any of them anymore. In other words, our agent will adopt the "Trigger" strategy if one of the opponents ever defects.

The Trigger strategy forms a Nash equilibrium both with itself and with Tit-for-Tat (TfT). Therefore, this rule counters the TfT because "TIT-FOR-TAT was able to succeed when it had the opportunity to play against other programs that were inclined to cooperate" [3]. By following the Rule 4, our agent does not always tend to cooperate and in fact, it is easy to defect.

1.5) Rule 5: "By my default nature, I am nice"

Our agent is nice by nature. It means that if all the above conditions are not met, our agent will choose to cooperate by default since it knows that "Social Welfare" is important in maximizing the total payoff.

2) Agent Implementation

The code snippet below implements our agent by following the rules that are described in the Section 1.

```
class Naing_Htet_Player extends Player {
    int selectAction(int n, int[] myHistory, int[] oppHistory1, int[] oppHistory2) {

        // Rule 1: our agent will cooperate in the first round
        if (n == 0)  {
            return 0;
        }

        // Rule 2: our agent will defect in the last few rounds, NastyPlayer mode is turned on
        if (n > 95) {
            return 1;
        }

        // Rule 3: if all players including our agent cooperated in the previous round,
        // then our agent will continue to cooperate
        if (myHistory[n-1] == 0 && oppHistory1[n-1] == 0 && oppHistory2[n-1] == 0) {
            return 0;
        }

        // Rule 4: check opponents history to see if they have defected before
        for (int i = 0; i < n; i++) {
            if (oppHistory1[i] == 1 || oppHistory2[i] == 1) {
                // if either one of them defected before, our agent will always defect
                return 1;
            }
        }
        // Rule 5: Otherwise, by default nature, our agent will always cooperate
        return 0;
    }
}
```

Figure 2: Implementation of the agent used in this assignment

3) Evaluation of Agent Performance against others

The source code given for this assignment is modified to calculate the average score of players for more than one tournament. Each tournament consists of about 100 rounds.

No. of Tournaments Played: 1	
Player Name	Total Scores
Naing_Htet_Player	**162.71387**
TolerantPlayer	160.97064
T4TPlayer	156.3113
NicePlayer	150.20436
NastyPlayer	142.06693
RandomPlayer	125.10323
FreakyPlayer	123.49353

No. of Tournaments Played: 5	
Player Name	Total Scores
Naing_Htet_Player	**171.784326**
T4TPlayer	167.349
TolerantPlayer	163.63618
NicePlayer	153.81274
FreakyPlayer	147.44822
NastyPlayer	137.74018
RandomPlayer	136.93472

No. of Tournaments Played: 10	
Player Name	Total Scores
Naing_Htet_Player	**167.68469**
TolerantPlayer	165.02788
T4TPlayer	162.01987
NicePlayer	152.92528
FreakyPlayer	144.5278
NastyPlayer	140.01362
RandomPlayer	135.56077

No. of Tournaments Played: 50	
Player Name	Total Scores
Naing_Htet_Player	**167.87414**
TolerantPlayer	163.01684
T4TPlayer	162.39293
NicePlayer	149.41562
FreakyPlayer	144.783086
NastyPlayer	140.43376
RandomPlayer	133.96014

No. of Tournaments Played: 100	
Player Name	Total Scores
Naing_Htet_Player	**168.14744**
TolerantPlayer	165.1346
T4TPlayer	162.57287
NicePlayer	151.61362
FreakyPlayer	143.44397
NastyPlayer	140.93894
RandomPlayer	133.33974

No. of Tournaments Played: 500	
Player Name	Total Scores
Naing_Htet_Player	**168.0062**
TolerantPlayer	163.68325
T4TPlayer	162.60442
NicePlayer	150.9609
FreakyPlayer	145.7182
NastyPlayer	140.39522
RandomPlayer	134.01652

Figure 3.1: Player performance rankings based on their total scores

In the above figure, it can be clearly seen that our agent, *Naing_Htet_Player*, outperforms other players in all cases. Here, it is important to note that, whenever only a single tournament is played, the above performance ranking for a single tournament may not always hold to be true. However, for approximately eighty percent of the time, our agent will always perform better than the rest in a single tournament. When the number of tournaments played is increased to 5, *T4TPlayer* performs better than *TolerantPlayer*. There are also some changes in the ranking among other players as highlighted by the green and red arrows.

When the number of tournaments is set to 10, *TolerantPlayer* overtakes *T4TPlayer* again in the ranking table. However, it could not still perform any better than our agent. Again, it is also important to note that the ranking may still vary among the top three players, *Naing_Htet_Player*, *T4TPlayer* and *TolerantPlayer.* Nevertheless, our agent, *Naing_Htet_Player,* will take the 1^{st} position for ninety percent of the time.

Our agent, *Naing_Htet_Player,* will always stay at the 1^{st} position when the number of tournaments is set to 50, 100 or 500. It can be concluded that the more number of tournaments are played, the higher the chance that our agent outperforms the rest consistently. The bar chart below will illustrate performance of the players visually.

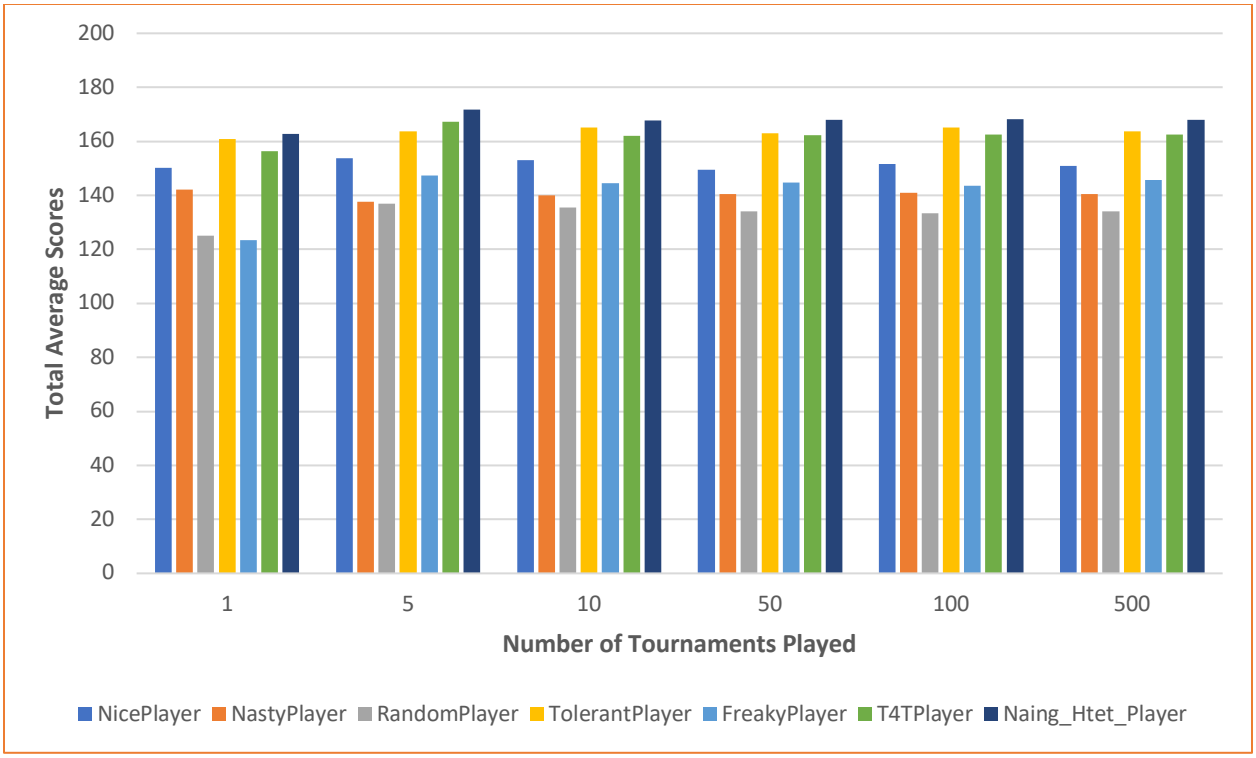

Figure 3.2: Performance evaluation of our agent against other players

3) Analysis of Agent Performance against others

3.1) Naing_Htet_Player versus Nice Player

Our agent performs always better than Nice Player no matter how many number of tournaments are played. This may be due to:

Rule 3, *"If they are nice, I will continue to be nice to them"*.

3.2) Naing_Htet_Player versus Nasty Player

Our agent always achieves betters scores than Nasty Player in all kind of tournaments. This may be due to:

Rule 4, *"If someone betrays me, I am triggered"*.

3.3) Naing_Htet_Player versus Random Player

Our agent will always beat Random Player regardless of how many number of tournaments are played. This statement is true for all cases. This may be due to:

Rule 3, *"If they are nice, I will continue to be nice to them"* and

Rule 4, *"If someone betrays me, I am triggered".*

3.4) Naing_Htet_Player versus Freaky Player

Since Freaky Player is either Nasty or Nice player, the explanation given in the above Section 3.1 and 3.2 is applicable.

3.5) Naing_Htet_Player versus Tolerant Player

Tolerant Player is one of the top competitors for our agent, along with T4T Player. Tolerant Player sometimes beat our agent when the number of tournaments played are low (i.e. 1, 5 or 10 tournaments). However, our agent will always trump Tolerant Player when more number of tournaments (for instance, 50, 100, 500, etc.) are played as it can be observed in the Figure 3.2. This may be due to:

Rule 2: "Play nasty for the last remaining few rounds".

When the number of tournament played is low, the impact of Rule 2 seem to be insignificant. However, as more and more number of tournaments are played, it has become more effective. Consequently, our agent outperforms Tolerant Player when the number of tournaments is set to 50, 100 and 500 in this assignment.

3.6) Naing_Htet_Player versus T4T Player

T4T Player is another top competitor in this assignment. T4T Player sometimes beat our agent and Tolerant Player when the number of tournaments played are low (i.e. 1, 5 or 10 tournaments). Our agent will always beat T4T Player when more number of tournaments (for instance, 50, 100, 500, etc.) are played as it can be seen in the Figure 3.2. It may be due to the same reason that is stated in Section 3.5 for Tolerant Player.

Conclusion

To conclude, the design of our agent in this assignment thrives against other popular strategies such as Tolerant Player and Tit-for-Tat Player because the composition of the rules for designing our agent is simple and balanced. It also resembles the four rules for success for the iterated prisoner's dilemma characterized by Axelrod [4].

Our agent is *not envious of other players*. It will always cooperate if the two opponents show sincerity in their willingness to cooperate. Our agent is nice because it *avoids being the first to defect* by always cooperating in the first round. Our agent will *reciprocate cooperation and defection* strategically. However, when it has been betrayed, our agent is triggered and will not trust its opponents again by playing Trigger Strategy. Ultimately, as shown in the Figure 2, our agent is *not trying to be too clever* since the implementation is simple and intuitive while achieving the best possible results.

References

[1] S. Tzu, "The Art of War with an introduction by Tom Butler-Bowdon", Capstone Publishing, 2010.

[2] M. Wooldridge, "Multiagent Interactions" in *An Introduction to Multiagent Systems*, John Wiley & Sons Ltd., 2002, pp. 114-122.

[3] M. Wooldridge, " Multiagent Interactions" in *An introduction to Multiagent Systems*, John Wiley & Sons Ltd. , 2002, pp. 114-122.

[4] M. Wooldridge, " Multiagent Interactions" in *An introduction to Multiagent Systems*, John Wiley & Sons Ltd. , 2002, pp. 114-122.