CSC8501 Coursework 2 – 2017
The Prisoner's Gang Dilemma
Due 27th October 2017 at 10am

Set by Graham.Morgan@ncl.ac.uk

Specification (what you need to do): You will extend the computer program from coursework 1 to handle situations where two rival gangs are questioned.

Clarifying the subject matter (the problem and suggested approach)
Rival gangs of prisoners:

- Two rival gangs exist (Purple Hand Gang and Magenta Hand Gang)
- Each gang has five members and all five members are questioned together
- Sentencing outcomes:
    - All members of each gang stay silent: 2 years for each prisoner
    - All members of the Purple Hand Gang betray all members of the Magenta Hand Gang and all members of the Magenta Hand Gang betray all members of the Purple Hand gang: 4 years for each prisoner
    - All members of the Purple Hand Gang stay silent whereas all members of the Magenta Hand Gang betray the Purple Hand Gang: Purple Hand Gang members receive 5 years and all members of Magenta Hand Gang receive 0 years
    - All members of the Magenta Hand Gang stay silent whereas all members of the Purple Hand Gang betray the Magenta Hand Gang: Magenta Hand Gang members receive 5 years and all members of Purple Hand Gang receive 0 years
    - If there is not total agreement within either or both gangs (i.e., 3 choose betray while 2 choose silence) then the sentence for both gangs are as follows:
        1. If the disagreement is equivalent when comparing gangs then 2 years for all prisoners
            - For example, Purple has 2 betrays and 3 silence and Magenta has 2 betrays and 3 silence
        2. If more choose betray in one gang than the other gang then all members of the gang with most betrays receive 2.5 years whereas the members of the gang with the least betrays receive 3 years
        3. If one gang is not in total agreement while the other is then the rules related to disagreement **are still applied to both gangs**
            - The gang with most betrays gets 2.5 years, and the other gets 3 years
    - A Spy may exist within a gang (randomly distributed across game iterations) and always chooses the minority option. If the spy is discovered then the gang the spy is in receives 0 or 2 years in prison while the opposing team receives 5 years in prison (irrelevant of prior rules governing prison sentencing). If a spy is discovered on opposing sides of the same game iteration then all gang members (and the spy) receive 6 years

- *Tournament*:
    - A game lasts for 200 iterations. Each prisoner may adopt a different strategy (even if they are in the same gang). A tournament requires all strategies to be used at least once in a game. A tournament determines the most successful strategy combination for a gang
- *Your program:*
    - You will utilise the ten strategies created from your previous coursework
    - Your program should read in ten strategy files (one for each prisoner)
        - A combination of strategies is allocated for each gang (we are testing out strategy combinations)
        - Multiple prisoners with the same strategy in the same game is allowed (there is no restriction on this)
        - Strategy files should be interpreted and 200 iterations of the game played with these strategy combinations
        - After each iteration all variables relied upon by the strategies in play are updated (so the strategies can make use of them)
        - After 200 iterations your program will display all the variable values (for each gang) and identify which strategy combination has won and the scores achieved
    - Your program should provide a tournament to ensure a number of strategy combinations are tested against each other in a full game
        - After a tournament the overall winning strategy combination is displayed complete with cumulative score (which combination of strategies was the best overall?)
    - Your program should be able to read in valid strategy files from any of your colleagues in the class and combinations of such strategies
    - You should reuse the same prisoner strategy interpretation language from coursework 1
    - You should produce results for when a spy is present in either team 0%, 5%, 10%, 15% and 20% of the time (randomly distributed across the games)
    - A spy is discovered by an extension of the Monty Hall Paradox:
        - In each game a gang leader is asked to choose another gang member (at random or using a strategy)
        - After this choice another gang member (not the leader, not the chosen member) is revealed as not the spy
        - The gang leader then decides to change their choice or stay with the original choice
            - If the gang leader does not change their choice and they discover the spy then this gang receives 0 years
            - If the gang leader changes their choice and discovers the spy this gang receives 2 year

Questions:

(1) What is the best gang based combination of strategies when no spy is present?

(2) Given the best strategy derived from (1), when a spy is present and is it better for the gang leader to change their choice?

Extensions to the Prisoner strategy interpretation language
LASTOUTCOME (gives the outcome of the last game) has been extended to include the additional 2 possible values: **A** (I was in a gang with a mixed response where we had most betrays); **B** (I was in a gang with a mixed response where we had the least betrays); **C** (I was in a gang with a mixed response where both gangs made cumulative equivalent decisions).
ALLOUTCOMES_A (how many times has A been the outcome previously for me during the current game?)
ALLOUTCOMES_B (how many times has B been the outcome previously for me during the current game?)
ALLOUTCOMES_C (how many times has C been the outcome previously for me during the current game?)
W, X, Y and Z can be associated to a gang rather than an individual prisoner.

Deliverables (what we want to see submitted):
- C++ source code authored by the student
- Executable file containing solution
- Output file that shows statistics for a tournament including the highlighting of the most successful strategy
- Output file that shows statistics for a tournament including the highlighting of how the best strategy performs given the possibility of a spy (0%, 5%, 10%, 15%, 20% of the time)
- You can submit the same program to coursework 1 and coursework 2 is it is easier (i.e., you have developed a whole solution that can demonstrate coursework 1 and coursework 2).

Demonstration (your chance to explain and show your solution):
On Friday 27th October from 10am onwards students will demonstrate their solutions.

Learning Outcomes (what we expect you to demonstrate in a general way)
- To be able to design and create programs, To be able to identify appropriate techniques for analysing the efficiency of programs, To be able to realise inappropriate usage of programming languages, To be able to manage memory, To be able to create and use data structures, To be able to use condition statements, loops and functions, To be able to utilise concurrency when appropriate, To be able to create programs that handle run-time errors, To be able to use appropriate techniques for debugging and analysing existing algorithms, To be able to design programs using a well-known methodology

Marking Scheme (what is worth what):
Marks are out of 25 and are awarded as follows:
- 10 Marks for achieving correct output
- 5 answers to the 2 main questions
- 5 Marks for advanced features (e.g., templates, threading)
- 5 Marks for adherence to the 7 rules of programming (see lecture 1)