COMP90041 ProjC Feedback: XX_USERNAME_XX

Program Presentation

Including: layout and style, readability, adherence to coding expectations, general care and appearance.
The full marks for this section of marking are as follows, depending on up to which stage you have completed.

- System Initialization and Exit, +0.5.
- Player Modification, +0.5.
- Player Ranking, +0.5.

Deductions

Some subset of the following lines will be selected by the marker. Deduct 0.5 marks for any two types errors listed below. (point out where the mistake happens if there is deduction)

- Constants not in upper case;
- bad choices for method names;
- bad choices for variable names;
- no commenting;
- inconsistent bracket placement;
- inconsistent indentation;
- lack of whitespace to separate different parts of code (visual appeal);
- lines >100 chars;
- no authorship statement (name, student number, username);
- use of magic numbers;
- other stylistic issue, if major then deduct 0.5 marks for this error alone;

Additions (only makes up for marks lost in this section)

- Overall care and presentation, +0.5;

Other Comments from Marker:

Structure and Approach

Including: decomposition in to methods, declaration of instance variables at the appropriate locations, choice of parameters to methods. The full marks for this section of marking are as follows, depending on up to which stage you have completed.

- System Initialization and Exit, +0.5.
- Player Modification, +0.5.
- Exception Handling, +0.5.
- Game Play, +1.
- Player Ranking, +1.

Deductions

Some subset of the following lines will be selected by the marker. Deduct 0.5 marks for any error listed below. (point out where the mistake happens if there is deduction, Maximum deduction is **3.5**)

- duplicate code segments;
- methods too long or too complex;
- insufficient use of methods;
- overly complex algorithmic approach;
- unnecessary duplication/copying of data;
- method has more than 5 arguments;
- use more than 3 static methods (main method included);
- use more than 4 static variables;
- use more than 4 public instance variables;
- not use try/catch to handle Exceptions;
- only throws **Exception** or **RuntimeException,** should use more specific/detailed exceptions(Both their custom exception or java pre-defined exception are fine.)
- AI player has been implemented without making use of inheritance/polymorphism (use if-else to test player type in game-play), -1.0 marks;
- **not** creating player instance or **not** using player instance to play the game, -1.0 mark;
- other structural issue, if major then deduct 0.5 mark;

Other comments from marker:

Program execution

Including: compilation, execution on test data, output presentation and readability. Programs that do not compile in the test environment will lose all marks in this section. Be sure to `verify` your submission and check the output before you say "finished" to yourself.

The full marks for this section of marking are as follows, depending on up to which stage you have completed.

- System Initialization and Exit, +1.
- Player Modification, +1.
- Game Play, +1.
- Player Ranking, +1.
- Exception Handling, +1

Deductions

Some subset of the following lines will be selected by the marker. (point out where the mistake happens if there is deduction, Maximum deduction is **5**)

- somewhat incorrect Exception Handling output on test1, -1;
- grossly incorrect Exception Handling output on test1, -2;

There are 2 testing points here in test1, 'createplayer' is not a valid command and Incorrect number of arguments supplied to command.
If students get 1 right and 1 wrong, deduct 1 mark, if students get 2 wrong, deduct 2 marks.
Do not deduct marks for game play logic here.

- somewhat incorrect Player Statistics Recording output on test2, -0.5;
- grossly incorrect Player Statistics Recording output on test2, -1;

If the students have 2 players but the data is wrong due to the wrong game play logic in test 1, deduct 0.5 marks.
For other errors deduct 1 marks.

- somewhat incorrect AI Player output on test3, -1;
- grossly incorrect AI Player output on test3, -2;

If the AI can play game in some logic, but in some stage it gives the wrong number for removing stones, deduct for 1 mark. In other words, if the AI can play, even if it just gives a random number, we only deduct 1 mark here.
If the AI cannot player at all, or each time it gives a fix number of stone to remove, deduct 2 marks.

Bonus part

A **screenshot for 1.5 marks** would look like this.

```
Your results seem to be CORRECT.  :)
====================BONUS=====================
Wining ratio if your AI player moves first, dummy rival player
Expected: 100%  Yours: 100%.

Wining ratio if your AI player moves first, oracle rival player
Expected: 100%  Yours: 100%.

Wining ratio if your AI player moves second, dummy rival player
Expected: 100%  Yours: 100%.

This test is for the victory guaranteed strategy for the advanced Nim game
Your solution is correct if it outputs 100 percent winning ratios in all cases
```

You will get 0.5 mark for each of the 100% in this part.

Important Note:

Your total mark for projects won't exceed 40, which means if you get full marks for Project A and Project B, also 11.5 for Project C, you will get 40 for projects in the end.

Total marks:

XX_TOTAL_MARK_XX

Overall comments from marker:

XX_EMPTY_COMMENTS_XX

Assignment Marker: XX_MARKER_XX
If you have any questions regarding your mark, please contact the lecturer.