Proejct 1: The "Target Game"

a Text-based Application Supporting Command Line Arguments

CISC 3120 Section TR2
Design and Implementation of Software Applications I

The project is to help students who have already had experience in programming in `C++` to achieve the following learning objectives,

- to write an application in Java consisting of *multiple* classes;
- to be able to use *instance variables*, *instance methods*, and simple *constructors*;
- to be able to use *flow control* structures in writing instance methods;
- to be able to design a *text-based* application;
- to be able to use *command line arguments* to control a behavior of an application;
- to be able to locate the Java API and third-party Java library documentation and use the documentation to support the development;
- and to be able to function in a team with support of a set of modern software development and productivity tools, such as, `git`, `Github` and the `Eclipse` IDE¿ and the `Eclipse` IDE.

Project Description

You are to develop and implement a text-based game application, called the "Target Game". The game is a single player game. First, the game sets a target on a two-dimensional space *randomly*, and asks the player to guess where the target is. The player wins the game when the player's guess is correct.

The instructor developed and provides an initial, but *incomplete* rendition of this game. The application is in a `Maven` project. You can locate the project at the `TargetGame` folder in the `sampleprograms` repository on `Github`. For your convenience, the URL to the project on `Github` is as follows,

`https://github.com/CISC3120/sampleprograms/tree/master/TargetGame`

Project Requirement

This is a team project. You and your team collectively must complete the following requirements by making revision to the start-up code.

If you and your team complete *all* of the requirements in this section satisfactorily, the instructor considers the quality of the project as "A". Be aware whether you as an individual will receive the "A" grade for the project depending also on a project peer evaluation to be conducted anonymously after the submission of the project.

Input Validation

The player inputs a guess by providing two integral numbers as the coordinate of the target. However, the start-up code will yield an error when invalid inputs are given, and the error is illustrated in the following example output:

```
Enter your target position (x, y):
18 26
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 26
 at edu.cuny.brooklyn.cisc3120.project.game.TargetGameBoard.getCell(TargetGameBoard.java:30)
 at edu.cuny.brooklyn.cisc3120.project.game.TargetGame.play(TargetGame.java:37)
 at edu.cuny.brooklyn.cisc3120.project.game.TargetGameLauncher.main(TargetGameLauncher.java:9)
```

You must add the logic in the application so that the application will discard the invalid input and ask the player to enter an input again.

Maximum Guess

The Game now permits the player to guesses any number of times before the player guesses correctly. You will set up a maximum guess threshold in the application. When the player exceeds the threshold, the game displays "`You lost`".

Game Level

To make the game fun, you will introduce a "difficulty level" in the game. The game at present displays the target on the screen, which aids the player to produce a correct guess. We shall call this "Level 1". To make the game more difficult, you will set up a threshold for maximum wrong guesses. When the player exceeds the threshold, the target becomes *invisible*. We shall call this "Level 2".

In this start-up code, the instructor provides an unfinished `GameBoardCell` class. You may use this class to revise your code to hide or show the target.

Command Line Arguments

Your enhanced game (the finished project) must support a few command line arguments. It is expected that you can run your game similar to the following,

```
TargetGameLauncher --window-width 80 --window-height 25 --level 2
```

where the first two command line arguments control the size of screen that the game uses and the third the difficulty level of the game. You must use the Apache Commons CLI library to support the command line arguments.

The official documentation of the Apache Commons CLI library is at

`https://commons.apache.org/proper/commons-cli/`

The instructor provides an example application at

`https://github.com/CISC3120/sampleprograms/tree/master/CmdLineArgsDemo`

Bonus Project Requirement

If you and your team complete any one of the following two *bonus* requirements *in addition to* the requirements above, the instructor considers the quality of the project as "A+".

Game Board Area and Game Statistics

You will introduce the concept of "Game Board Area" in the design of your game. The instructor has included an empty `GameBoardArea` class in the start-up code. You shall incorporate this class in the design of your game. An instance of the `GameBoardArea` class represents a rectangle region of the board. The game has three instances of `GameBoardArea`, one is for the area where the target is located, one for player's input, and one for game statistics. The game statistics can include the number of guesses. The locations of these three areas on the screen can be determined from the command line, such as,

```
TargetGameLauncher --TargetArea top-left --StatisticsArea top-right --InputArea bottom
```

and

```
TargetGameLauncher --TargetArea top-right --StatisticsArea top-left --InputArea bottom
```

Third Party Library: Lanterna

The display in the start-up code is done essentially via `System.out` in Java. You can improve it via a third-party text user interface library. You shall revise the game using the Lanterna library. You can find the library at

`https://github.com/mabe02/lanterna`

Project Invitation

Each team shall elects project coordinator for this project. The coordinator has the following responsibility,

- to accept the assignment invitation via the Github Classroom;
- to clone the team project repository;
- to copy and add the start-up project to your own project repository
- to commit and push the project, and to inform team members that the project set-up is ready

The team members shall accept the invitation and clone the project repository. The collaboration and project development continues.

The project assignment invitation is at

`https://classroom.github.com/g/6t0ALjnK`

Submission

Submit your project by pushing your project to Github by 11:59PM, Wednesday, September 27.