Refactoring documentation for project Labyrinth-4

Team “Nitrogen”

1. Changes to the project structure:

* Renamed the default source files: **KPK\_proekt** 🡪 **LabyrinthGame**, **LabyrinthProcessor** 🡪 **LabyrinthEngine**, **Top5Scoreboard** 🡪 **Scoreboard**.
* Added static class **LabyrinthInputOutput** to handle all visualization to the console
* Added class diagram
* The method for visualization of the game field is moved to the **LabyrinthInputOutput** class
* The method for visualization of the scoreboard is moved to the **LabyrinthInputOutput** class
* Extracted private methods from the code to improve readability (e.g. **IsAvailable** checks if the next position after the move is a free cell)
* Extracted all code form the class’ methods that handles console output and moved it to the

**LabyrinthInputOutput** class (e.g. the printing of victory message from the **IsFinished** method)

1. Renaming of variables and methods:

* Capitalizing the first letter of **Move** methods,
* Renamed position variables to more suitable names
* Renamed console output methods to more suitable names

1. Additions/changes to the code logic

* Introduced constants for the game field in the **LabyrinthMatrix** class: **CenterX, CenterY, MatrixSize**
* Added constants for the cell types : **EmptyCell, WallCell**
* All methods that previously made checks with “magic numbers” now use the constants instead
* Added separate methods for console output in the **LabyrinthInputOutput** class
* Most output methods now return string value instead of printing directly to the console

1. Defensive programming

* Added checks for validity of the input to all public methods/properties that require input
* Added exceptions for incorrect inputs

1. Improvement of functionality

* Added recursive method **IsCorrect** for checking if a generated matrix is valid (i.e. has at least one possible exit path from the starting position)
* The **Restart** method of the **LabyrinthEngine** class now generates a matrix and checks if it is valid, if not – a new matrix is generated until a valid one is found.

1. Unit test

* Added unit tests for all output methods by retargeting the output to a Console.SetOut stream
* Added unit tests for invalid inputs from the user