Logo

Description automatically generated

ALGORITHMS & DATA STRUCTURE

MINESWEEPER GAME

ĐÀO HUỲNH THIÊN LONG ititiu19026

LA TRÍ NGUYÊN ITITIU19036

ĐỖ ĐÔNG QUÂN ititiu19043

VÕ HUỲNH HUY THỊNH ITITIU19043

INTERNATIONAL UNIVERSITY

VIETNAM NATIONAL UNIVERSITY

Table of Contents

**I. introduction 2**

**II. PROPERTIES OF TETRIS 3**

1. **RULES 3**
2. **FUNCTIONS 3**

**III. IMPLEMENTATION 4**

**IV. DESIGN OF PROJECT 8**

1. **Creation board game, set bomb and hint numbers 8**
2. **Set cell icons 10**
3. **Set hint numbers around bomb cells 10**
4. **Calculate when player choose empty cell 11**
5. **Calculate the statistics of player 13**
6. **Save the player’s statistics 14**
7. **Check each clicked cell 16**
8. **Check neighbours 17**
9. **Check game’s solution 17**
10. **Result 18**

**V. RESULT 19**

**VI. CONCLUSION 19**

# 

# **INTRODUCTION**

- This project was designed based on Algorithms & Data Structure method by Java language. This method has solved some problems occurred in the building process using algorithms and structure method:

* The code is clear, concise and easy to understand.
* The project is a unified logic system of related classes.
* Each class has methods which take on different behaviors.
* The ability of reusing the resources.

- The purpose of this project is to design a basic game that was built on the foundation of data and algorithms structure. Our group decided to modify a game named “MineSweeper – DSA\_Project”, following the principle of this method. Minesweeper is a game which requires the user to open all cells on the board in order that you won’t touch unrevealed bombs under cells. This game will clearly show the course properties, the combinations of classes and the relationships between classes and objects.

- Being inspired by this famous game, our group choose to create a special version of MineSweeper by applying the knowledge that we have learnt from the course ADS. Furthermore, we want to challenge and improve ourselves on building a software that runs smoothly. In the process of creating, we have met many difficulties that force us to practice our soft skills, such as teamwork, problem solving, information finding and selecting. Through this project, we have found ourselves getting better and having a deeper understanding of the algorithms and data theories.

1. **PROPERTIES OF MINESWEEPER**
2. **Rules**

- The board is divided into cells, with mines randomly distributed. To win, you need to open all the cells. The number on a cell shows the number of mines adjacent to it. Using this information, you can determine unrevealed cells that are safe, as well as cells that contain mines. Cells suspected of being mines can be marked with a flag using the right mouse button.

1. **Functions**

- Our minesweeper game will be played according to the basic rules but there will be some more open functions that attract players:

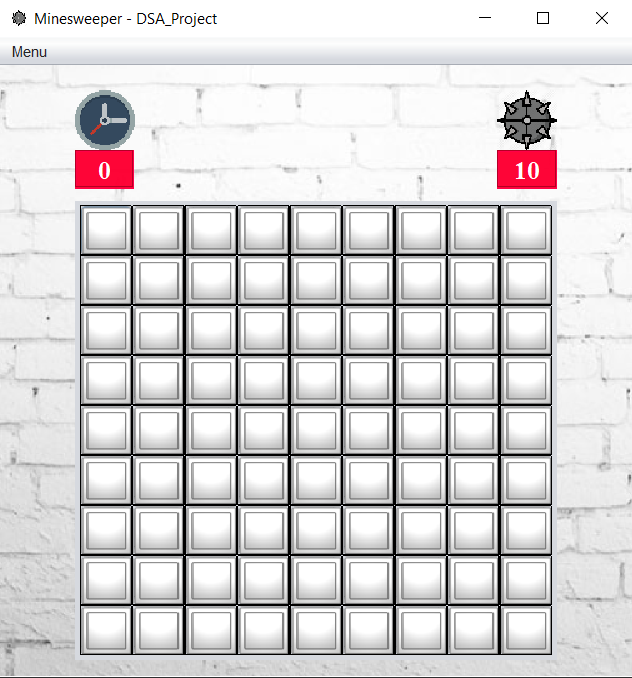
* You can mark the empty box that you think an unrevealed bomb by right-clicking and “Flag” will be displayed.
* If you click on any cell it will have 2 directions displayed:
* You click on the cell with the number, it will show the number of bombs around with the number in that cell.
* Unfortunately, you click on the bomb box, the game will show you 10 red cells which have bombs under and will automatically save the result including time and your performance on the leaderboard.
* If you click an empty cell, the system will display a wide hint until there are cells with numbers for you to guess.
* A special thing in our game project is that the system will total up your performance parameters. If you click on the "Menu" 🡪 "Detail" the system will give a table of data of players who have been saved so far such as:
  + Winning streak
  + Losing streak
  + Number of games played
  + Number of wins
  + Win Percentage
  + Current Streak
    - In this section if you click "Reset" it will delete all your material to the beginning.

1. **IMPLEMENTATIONS**

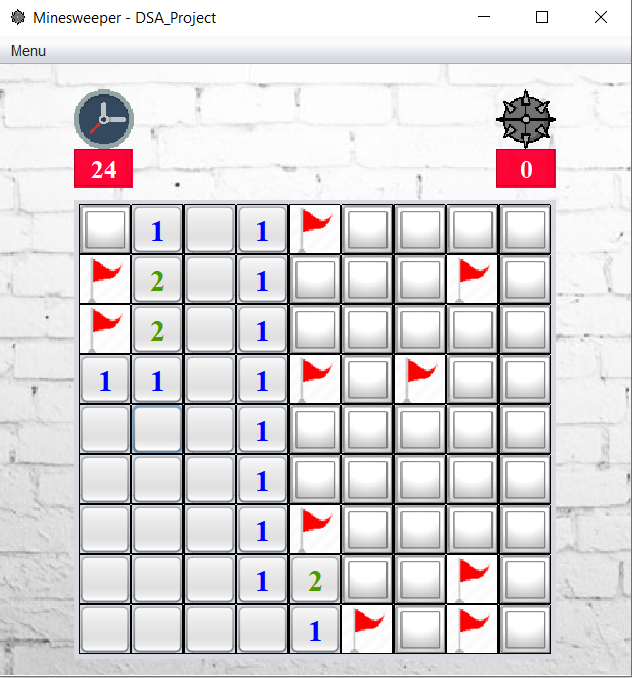
- On the game screen, system will display the number of bombs to guess and will decrease if you mark by flag on the game board.

- The time it takes to count the playing time of 1 game is saved in the system.

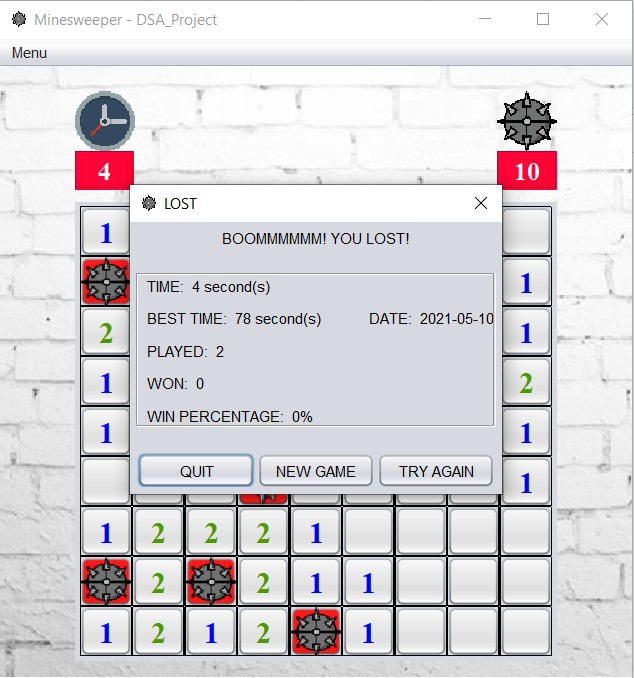
- Board game is a 9x9 two-dimensional array.



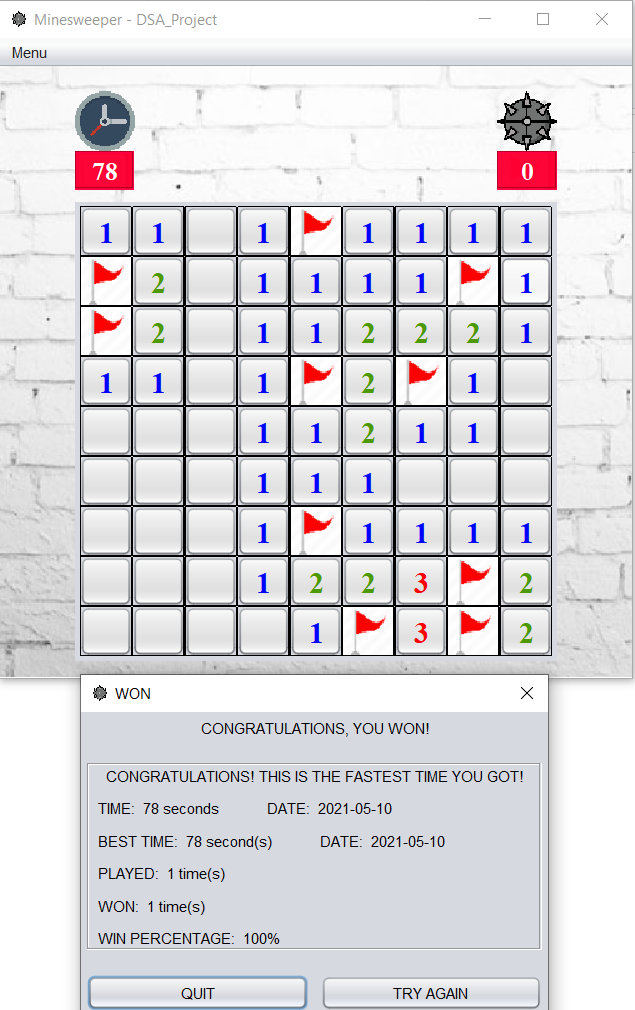
**-** When you want to mark a bomb, right-click on the cell then the system will display a flag on the selected cell and reduce the number of bombs to guess by 1.



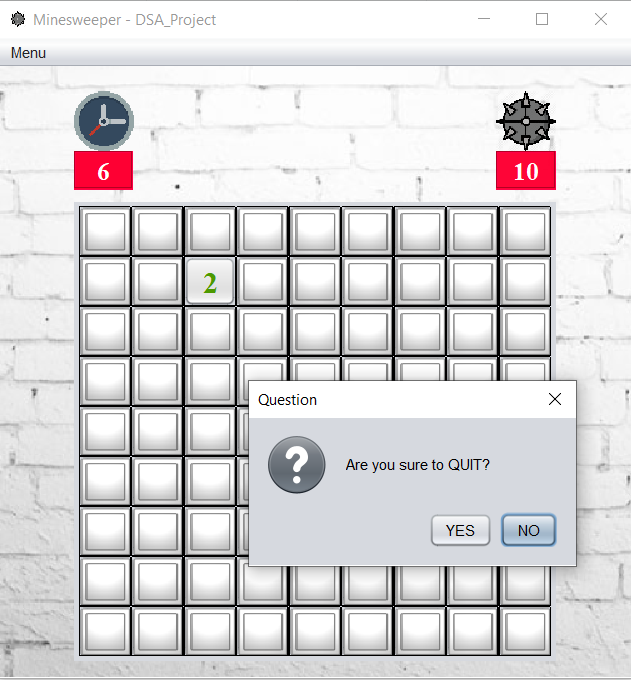
- Unluckily, when you touch an unrevealed bomb, you will lose and the system will display a message board of parameters you have played and ask if you want to play a new game. It will also automatically count the increased number of games played if you click “Try again” and will reset all your previous details when you click “New game”.



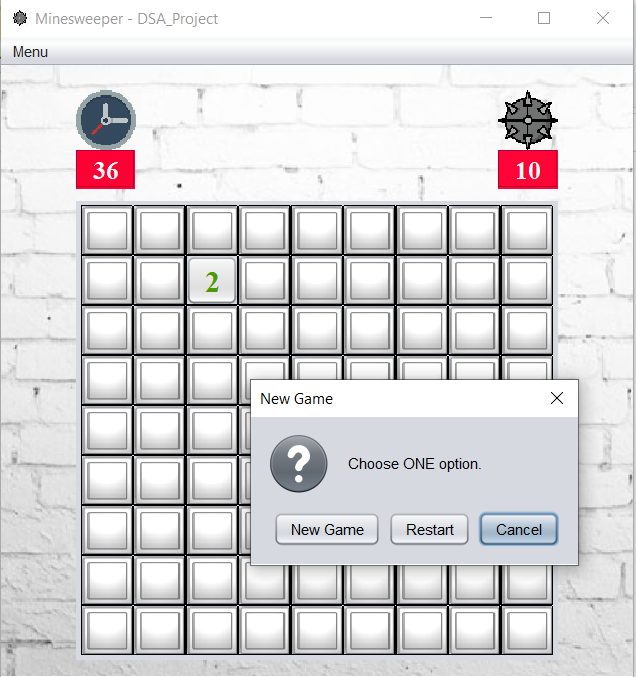
- Congratulations message will be displayed when you win along with invitation asking you to continue playing with "Try again" or turn off the game with "Quit”. In addition, your victory information will include “Best Time”, “Date”, and “Win Percentage”.



- You can exit the game at any time, the system will confirm again and will not save the game information you have played.



- You can have 2 options: "New Game" to play a new game or "Restart" to play again from the beginning if you choose "Menu" --> "New game".



- In “Menu” have another option “Player profile” to see statistics of you have played. System will show you statistics about:

- Winning streak

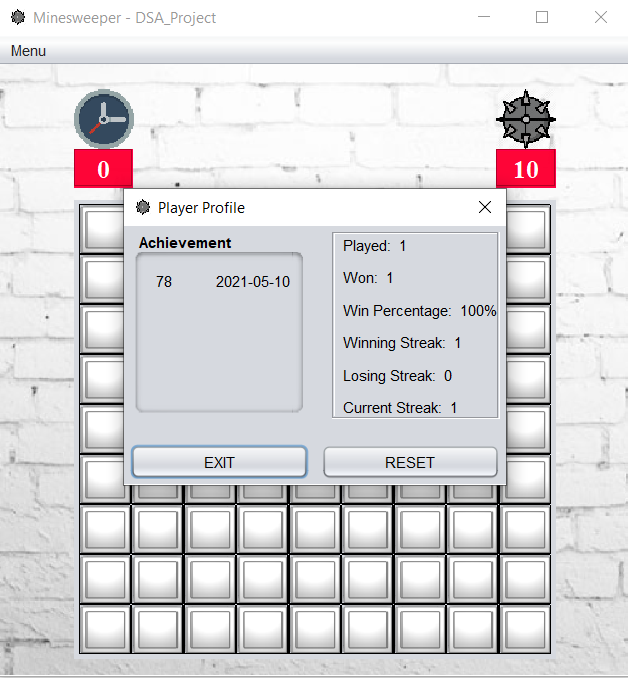
- Losing streak

- Number of games played

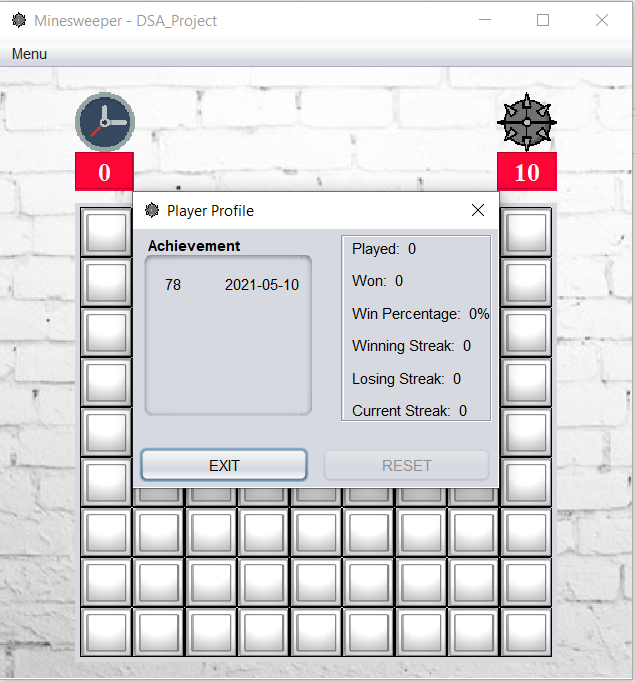
- Number of wins

- Win Percentage

- Current Streak



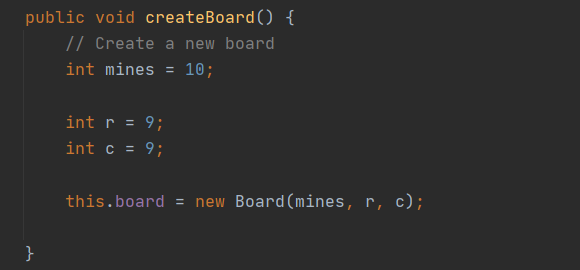
- And special thing that show you information about game you won with time and date. You also can clear all statistics with “Reset” to recount all previous information when clicking “Reset”.

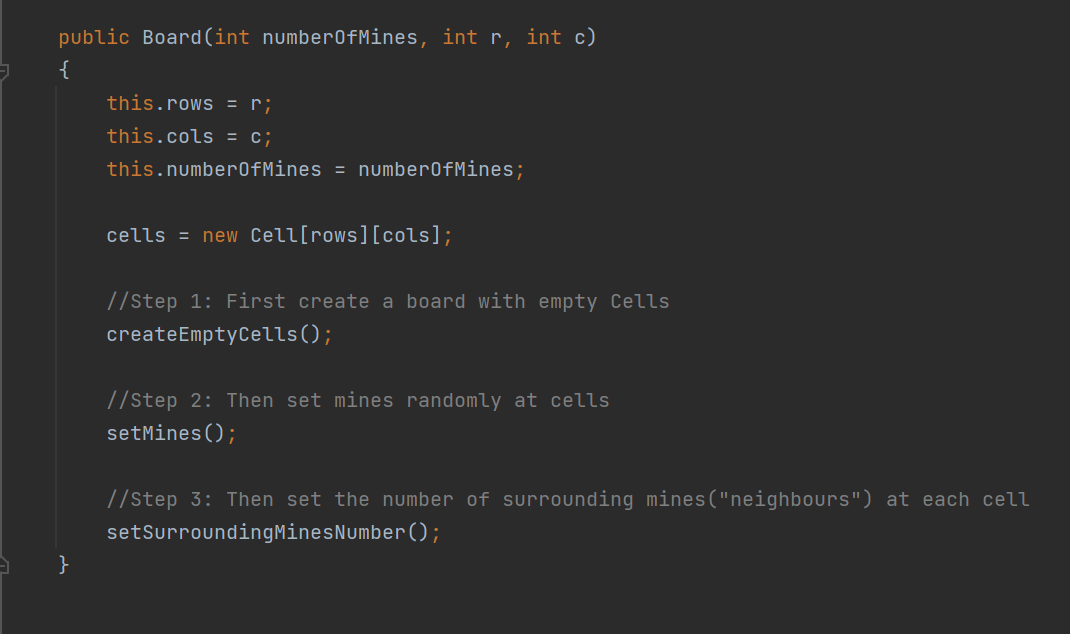


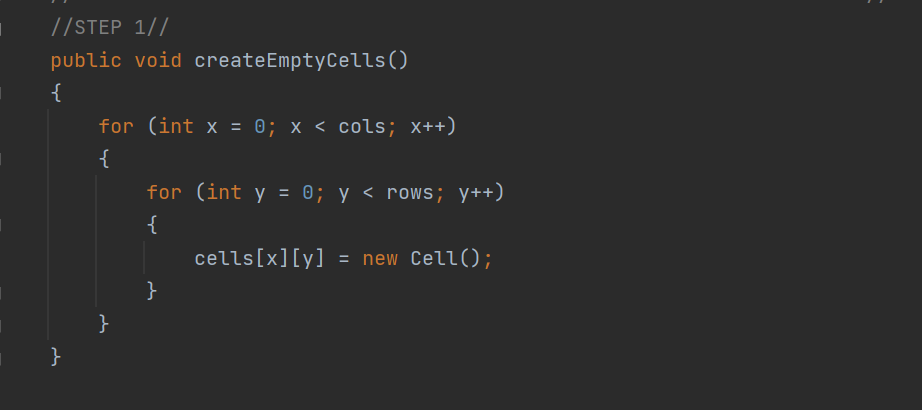
1. **DESIGN OF PROJECT**
2. **Creation board game, set bomb and hint numbers:**

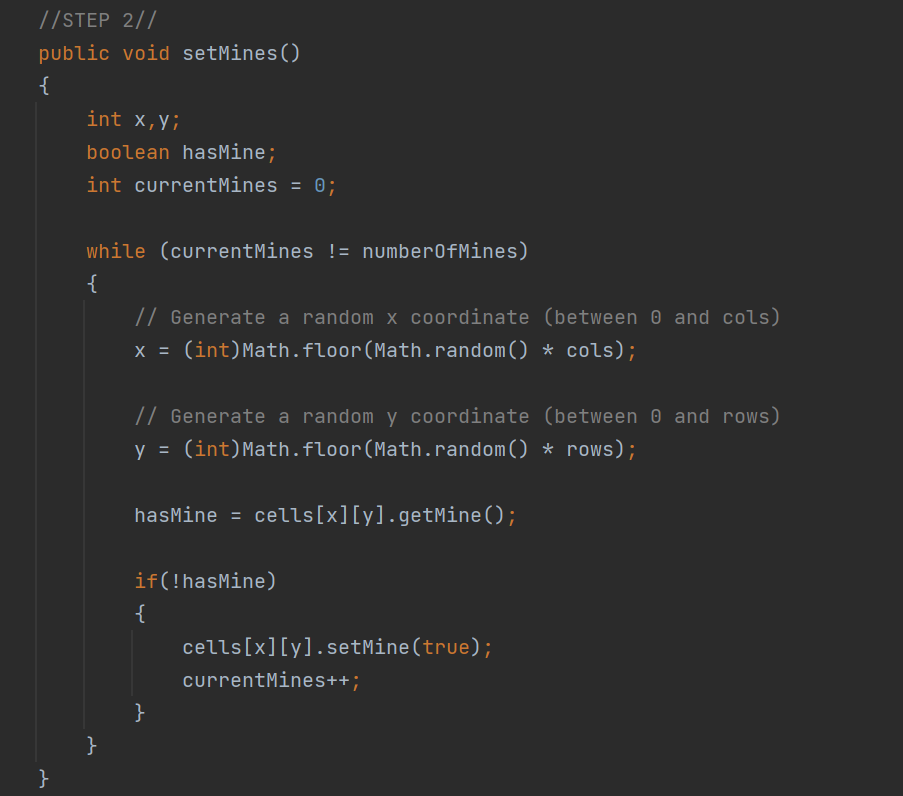
- To create a game board, we use a 9x9 two-dimension array and set 81 empty cells for that array.

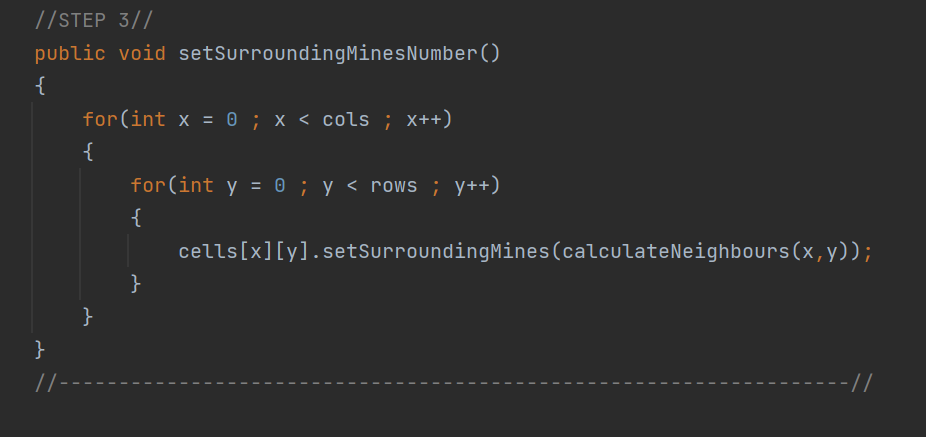
- We put up 10 bombs randomly within the boundaries of the board and set up the number cells around the bombs that were placed.



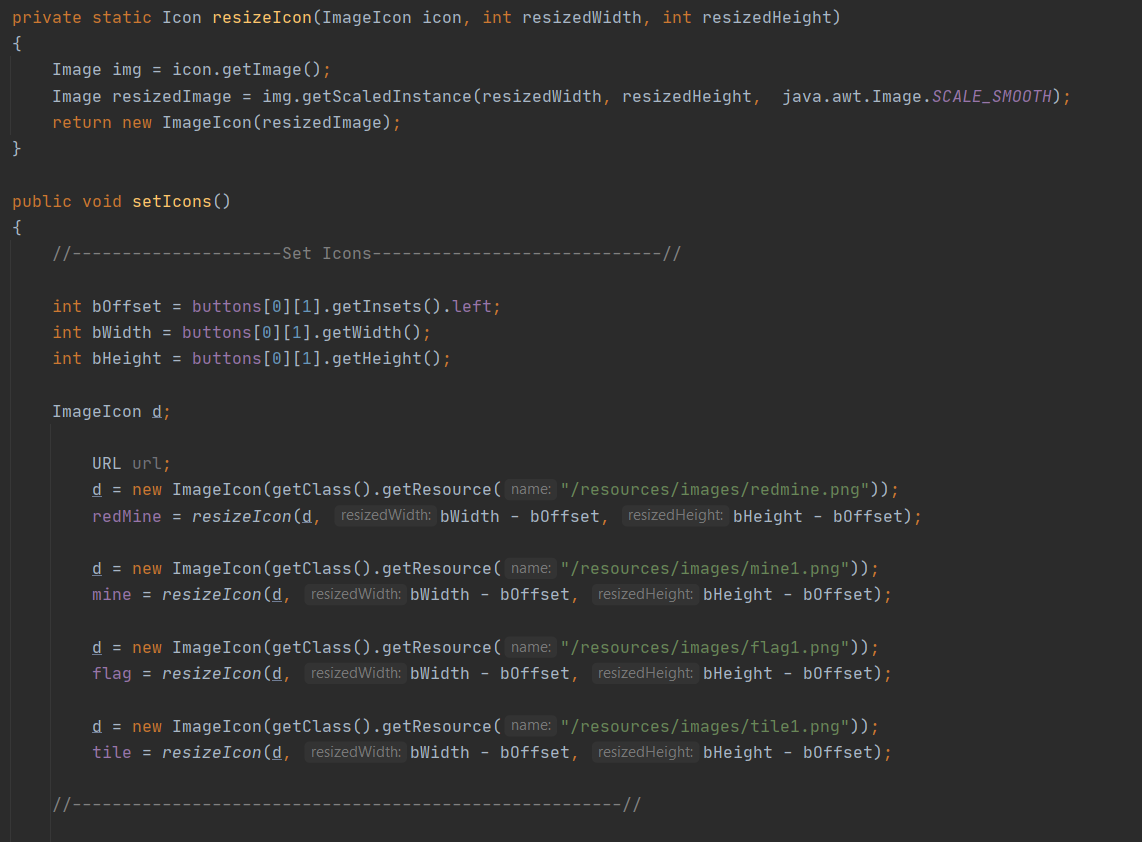








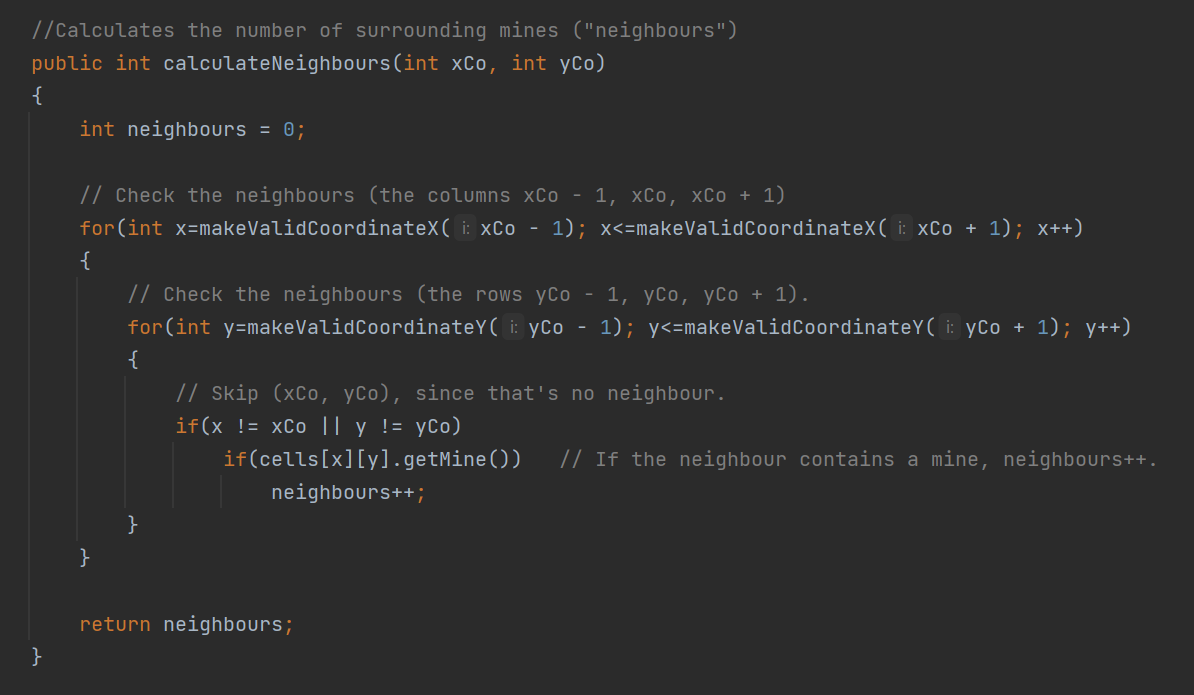
1. **Set cell icons**



1. **Set hint numbers around bomb cells**

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

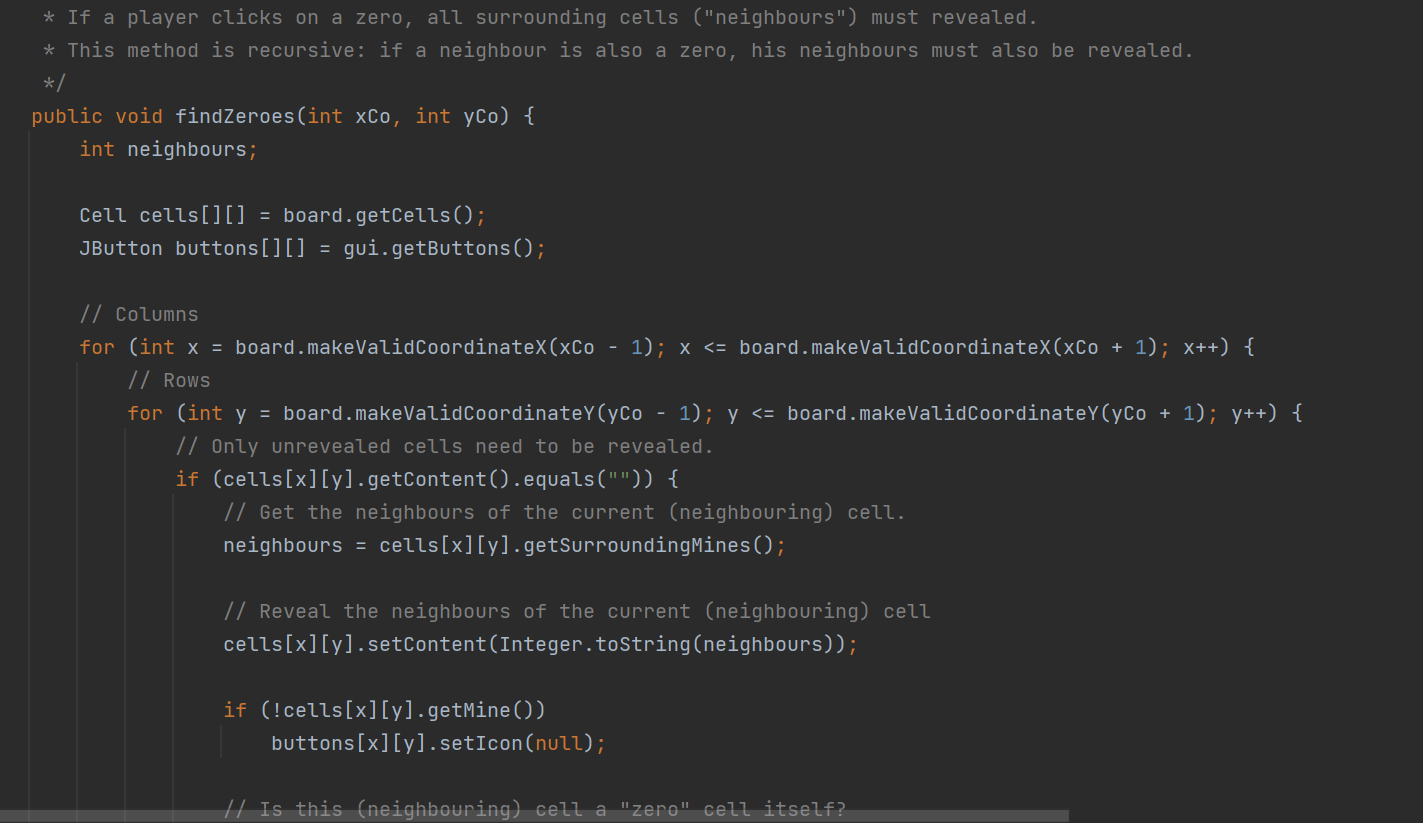
- To set the number cells, the system considers from the cells with bombs (blue cells) and 8 surrounding cells (red cells), when there is a bomb the red cells will automatically count from 1, if the red cells and bombs will not count but count the surrounding cells, the number cells will increase if more bombs are detected than the number of bombs checked.

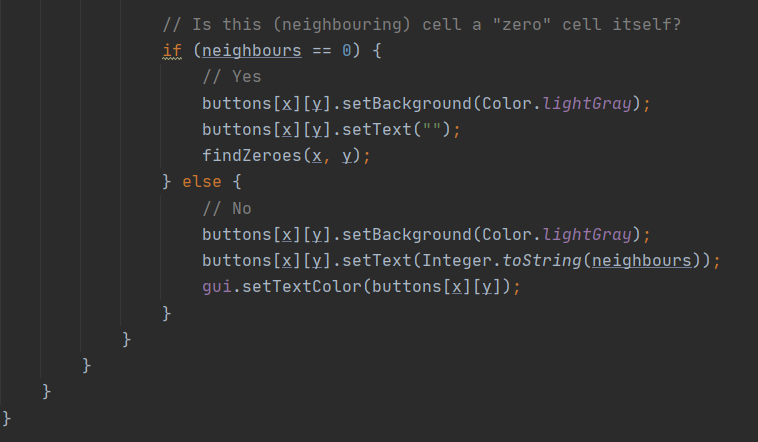


1. **Calculate when player choose empty cell**

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

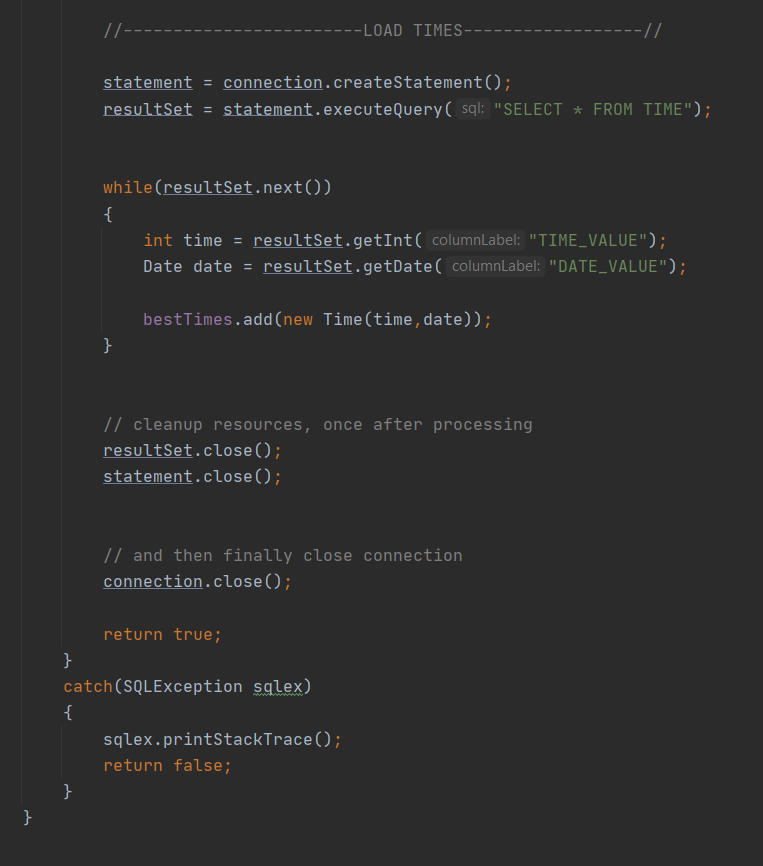
- When the player selects an empty cell (for example, a blue cell), the system checks for 4 cells on the edges of the empty cell (red cell). If these red cells is blank cells, the system will continue to consider the rules with that cell, and if the red cell is not the empty cell, it will be considered if the cell is a number cell, it will stop.



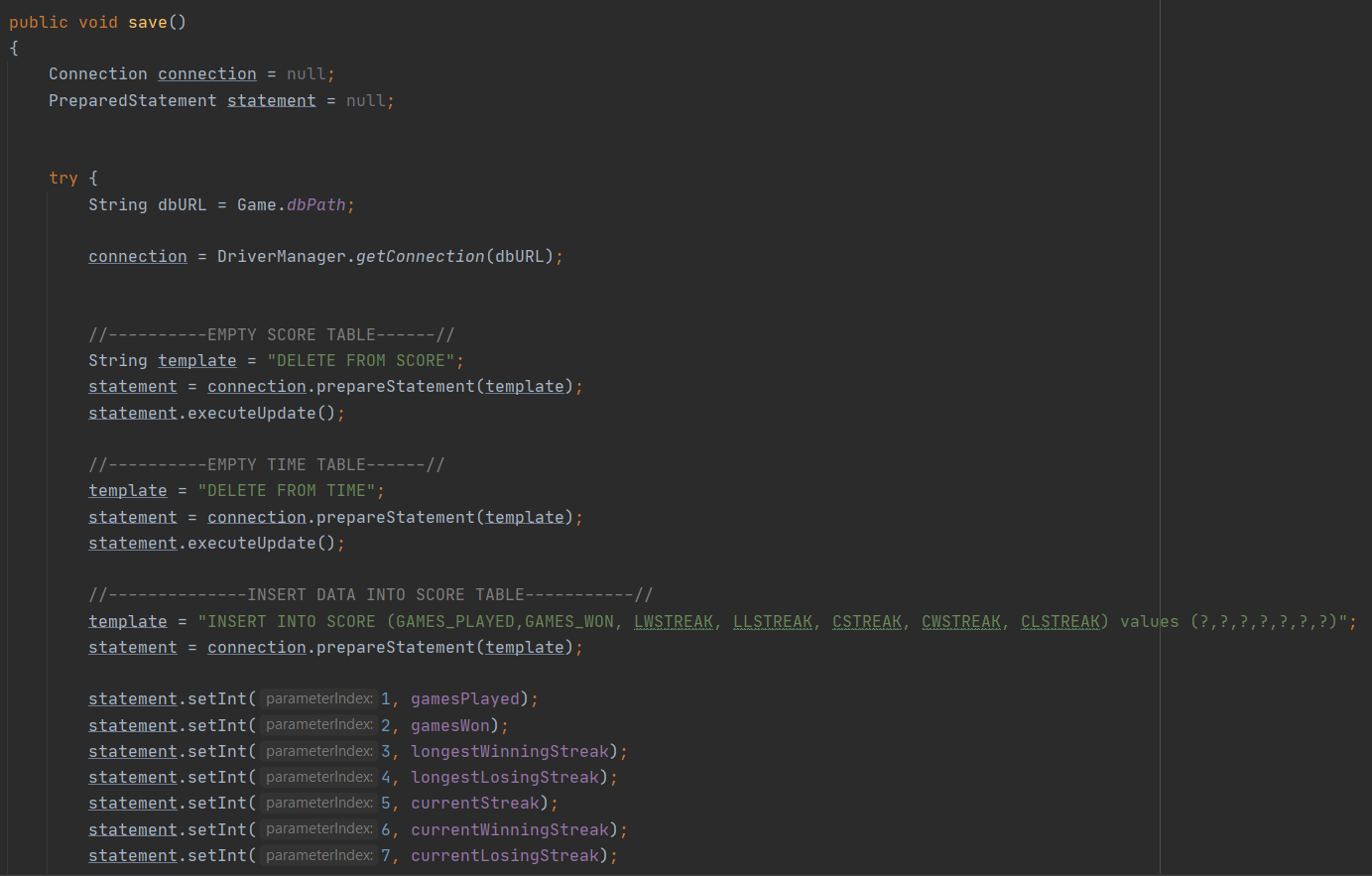


1. **Calculate the statistics of player**



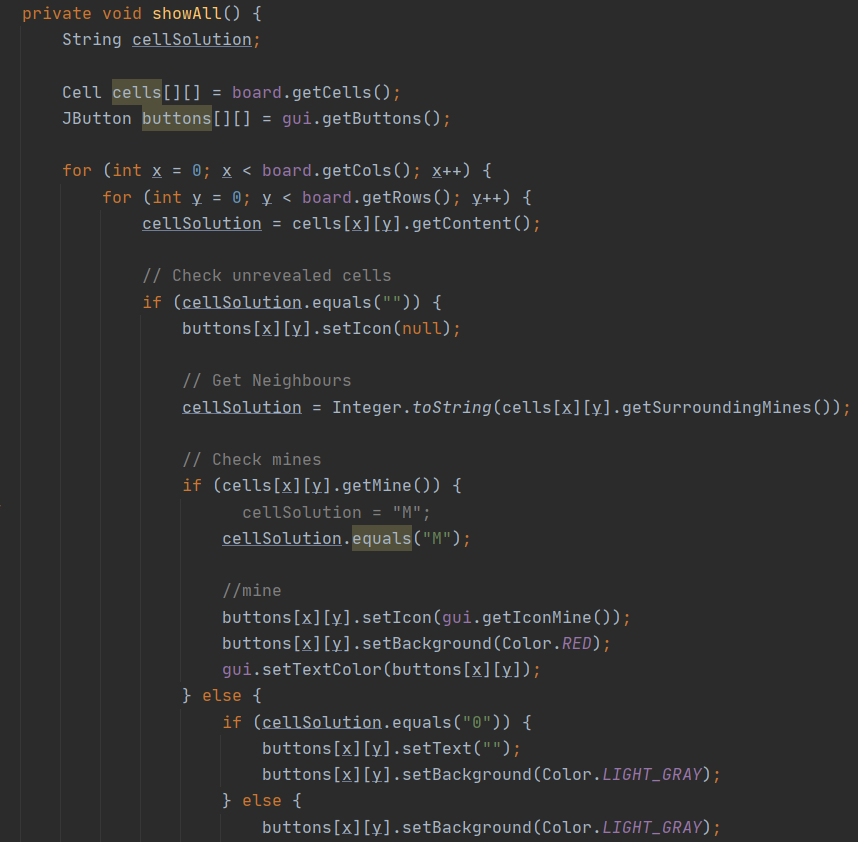


1. **Save the player’s statistics**



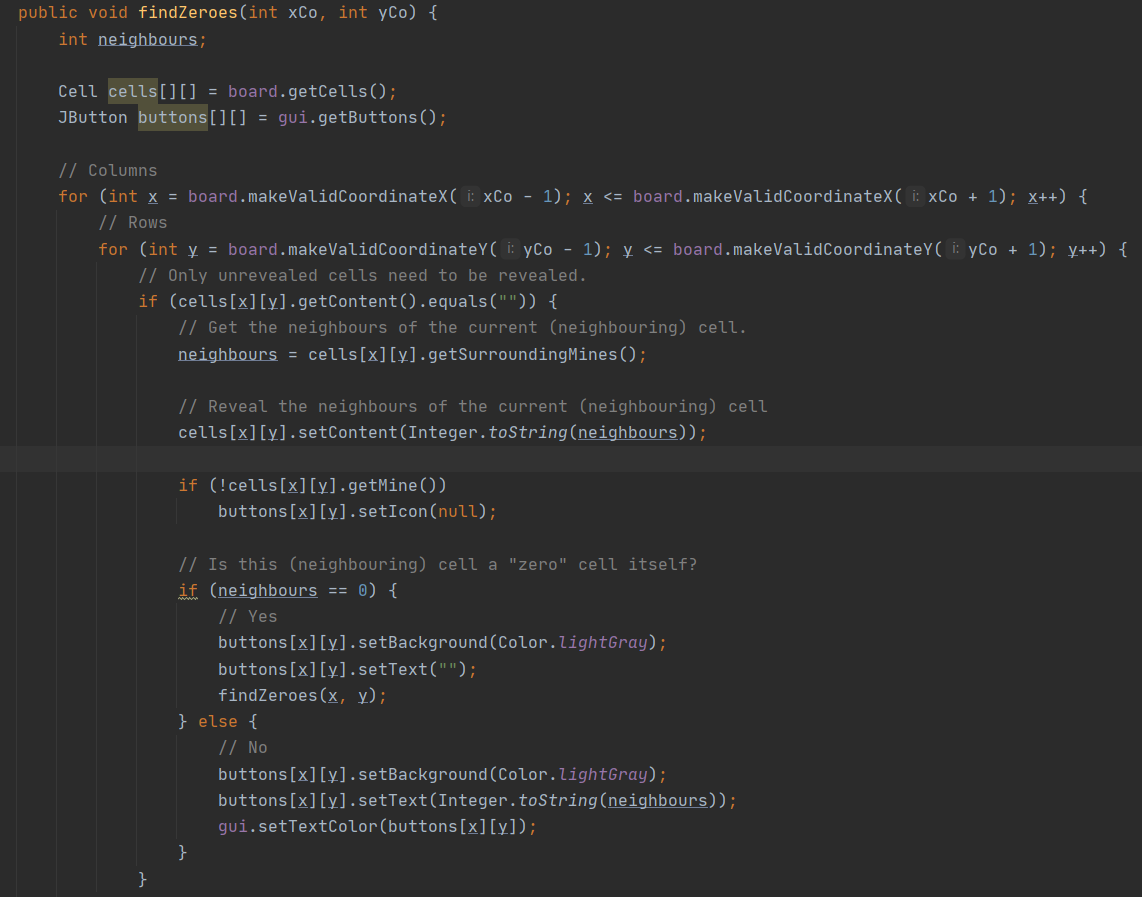


1. **Check each clicked cell**





1. **Check neighbours**



1. **Check game’s solution**



1. **Result**



1. **RESULT**

- Our project has followed completely the principle of Algorithms & Data Structure knowledge and the rule of MineSweeper. Classes and objects in the system are connected relatively and logically. We also success in manipulating the display to conect the user with program. This game can be control by mouse and keyboard.

- Morever, **some new features** out of the basic rule are added into our game such as the background while playing, some button to pause, stop, refresh in order to make the user interface more attractive.

1. **CONCLUSION**

Through this project, we had a chance to study and understand deeper the application of Algorithms & Data Structure in building a program. We also show the systematical and logical relationships between the classes. However, due to the short time of the course, we cannot finish it as its best. It can be assured that our project will be completed in the future.