Maze Game Project

Text

Description automatically generatedTeam : beyond the wall

November 2021

Contents

[1. Project descroption 1](#_Toc87113148)

[2. Participants in the project 1](#_Toc87113149)

[3. Summary of the project 2](#_Toc87113150)

[3.1 STAGES OF REALIZATION 2](#_Toc87113151)

[3.2 DIFFICULTIES 2](#_Toc87113152)

[3.3 Game description 2](#_Toc87113153)

[4. TABLE WITH FUNCTIONS 5](#_Toc87113154)

[5. Block scheme 6](#_Toc87113155)

[6. Future Ideas 6](#_Toc87113156)

# 1. Project description

Our task was to make a game using the language C++. The game had to be made using dynamic memory. The game had to contain a maze which the player had to escape from using the arrow keys.

# 2. Participants in the project

* Niya Runcheva 10A - Front-end Developer - NSRuncheva19@codingburgas.bg
* Lyubomir Bozukov 10B - QA - LVBozukov19@codingburgas.bg
* Tereza Opanska 10V - Back-end Developer - TNOpanska19@codingburgas.bg
* Miriam Georgieva 10G - Scrum Trainer - MKGeorgieva19@codingburgas.bg

# 3. Summary of the project

## 3.1 STAGES OF REALIZATION

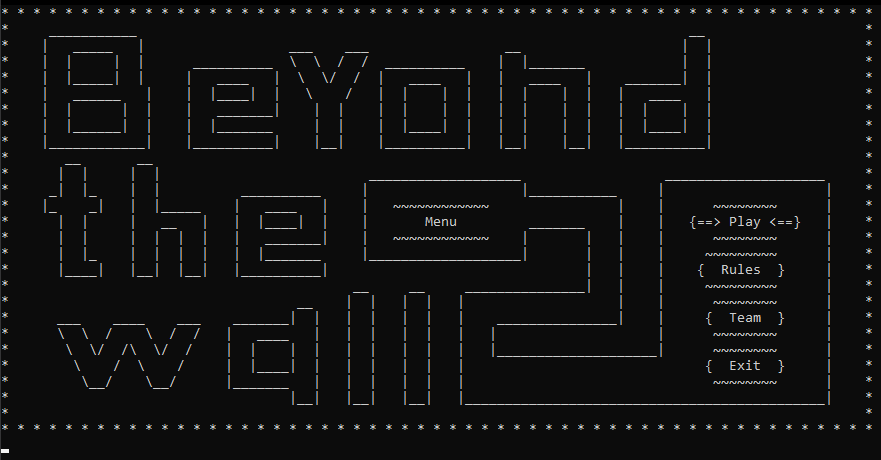
1. The first stage we had to go through was making our team according to the given team making criteria. After forming our team, which took us roughly around a week, we held our first meeting. There we choose our team’s name and gave each other different roles and their according tasks.
2. The second stage was working on our given tasks. This being the most difficult stage was also the most time-consuming. Fortunately, we did not have many problems and in the end, we had a finished product.
3. The third and final stage was finalizing our GitHub repository, creating the presentation and documentation, making our README.md file and getting ready to present our work.

## 3.2 DIFFICULTIES

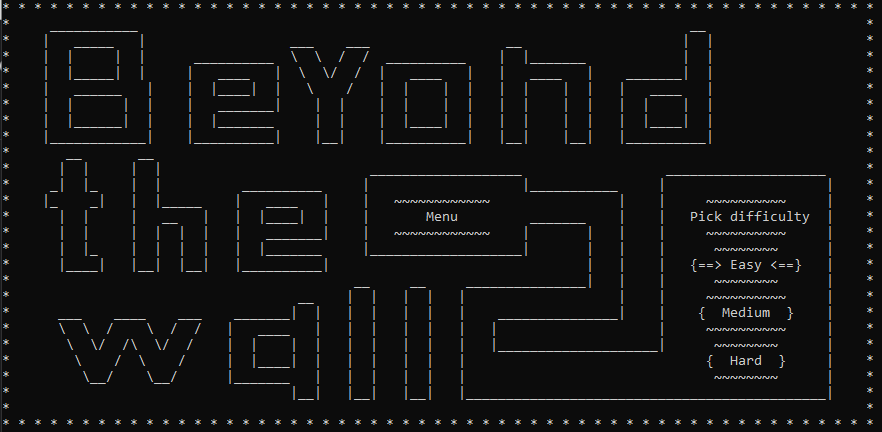
Although we did not have many difficulties some of them were making the program work using arrow keys and making the actual mazes you have to escape look appealing for the users. We also had one of the most common problems in group projects, which was finding a right time for the whole team to have a meeting.

## 3.3 Game description

The first thing you see when running our game is the main menu. On it you can find the options: Play, Rules, Team and Exit.



If the Play option is chosen you get sent to a second menu where you can choose the maze difficulty.



Depending on the chosen difficulty you get sent to a maze which should look something like this:



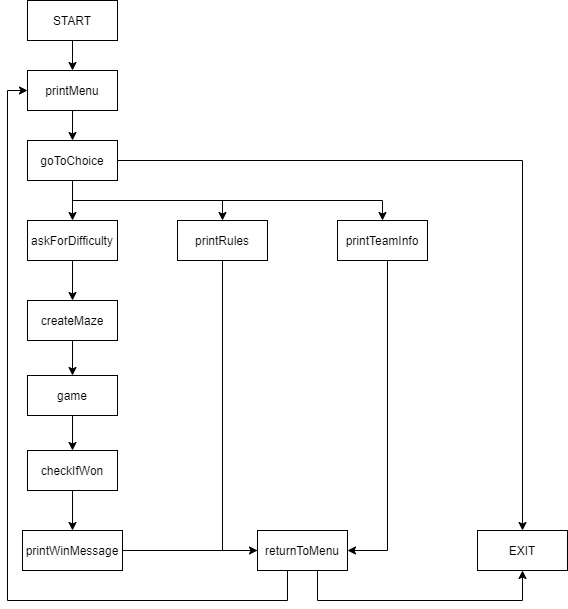
When completing the maze, you get asked if you want to get sent back to the main menu or fully exit the program. By pressing “Enter” you return and by pressing “ESC” you exit the game.

Back at the main menu the three options left are “Rules” which shows you a page explaining the rules of the game, “Team” which provides you with information about who we are and “Exit” which exits the program.

# 4. TABLE WITH FUNCTIONS

|  |  |  |
| --- | --- | --- |
| Function Name | Function Type | Function Description |
| printMenu() | void | Prints out main menu. |
| goToChoice(int option) | int | Sends the user to their selected action. |
| askForDifficulty() | void | Asks the user for maze difficulty. |
| printRules() | void | Prints out the game rules. |
| printTeamInformation() | void | Prints out the information about the team. |
| createMaze(unsigned difficulty) | char\*\* | Creates the maze according to the selected difficulty. |
| getMazeRows(int difficulty) | int | Returns the number of maze rows according to the difficulty. |
| getMazeCols(int difficulty) | int | Returns the number of maze columns according to the difficulty. |
| game(char\*\* maze, int rows, int cols) | void | The main game functionalities get performed in this function. |
| displayMaze(char\*\* maze, int rows, int cols) | void | Displays the maze. |
| makeMove(char\*\* maze) | char\*\* | Changes the player’s position depending on which keys they press. |
| checkIfWon(char\*\* maze) | bool | Checks if the player has escaped the maze. |
| printWinMessage() | void | Prints out a message congratulating the player after wining. |
| resetGame(char\*\* maze, int rows) | void | Resets values to all global variables. |
| returnToMenu() | void | Asks the player if they want to return to the main menu. |

# 5. Block scheme



# 6. Future Ideas

For the future when we have more knowledge in the area of programming there are a few things that we would like to change. The first one would have to be the creation of the maze. For now, we only have 3 mazes which you can escape that do not change. If we have the chance, we would like a feature in the future that would generate random mazes, so that players get a better experience from our game. Another additional change that we would make is adding colors to the design and potentially making it better.