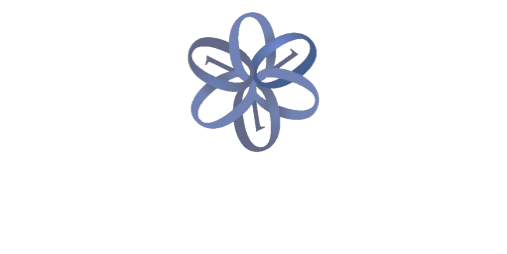
Vocational School of Computer Programming and Innovations



play and LEARN

Documentation

Contents

[Authors 2](#_Toc42896955)

[Summary 2](#_Toc42896956)

[Description of the application 2](#_Toc42896957)

[Table with some of the most important functions 3](#_Toc42896958)

[Diagram of the program 3](#_Toc42896959)

[Milestones in the realization 4](#_Toc42896960)

[Used resources 4](#_Toc42896961)

[Conclusion 4](#_Toc42896962)

# Authors

* Stanislava Andonova - [srandonova18@codingburgas.bg](mailto:srandonova18@codingburgas.bg), 9V class
* Christina Bolashikova - [kvbolashikova18@codingburgas.bg](mailto:kvbolashikova18@codingburgas.bg), 9V class
* Zhenya Yordanova - [ZIYordanova18@codingburgas.bg](mailto:ZIYordanova18@codingburgas.bg), 9G class
* Yavor Karakolev - [YAKarakolev18@codingburgas.bg](mailto:YAKarakolev18@codingburgas.bg), 9B class

# Summary

## Description of the application

Our project is a program developed with C++ on the topic of "strings". At the very beginning of the start-up, a message is displayed on the screen welcoming the user and instructing them that they can start by selecting an option from the main menu below. These options are as follows: “**1. Input a text and get analysis about it**”, “**2. Play games**” and “**3. Exit**”.

When selecting the first option, the user is required to enter a text, and the input stops in the moment a new line is passed. Then there is another menu with seven possible choices, six of which somehow analyze the text or display statistics related to it, and one that allows you to return to the main menu, namely: „1. **Number of words**” - displays how many words there are in total in the text, “**2. Number of sentences**” - displays how many sentences there are in total in the text, “**3. Does it contain a specific word**” - indicates whether a specific word entered by the user is present in the text or not, “**4. How many times is a specific word mentioned**” - shows how many times a specific word entered by the user is present in the text, “**5. How many times each word in the text is used**” - displays how many times each word in the text is used, “**6. Most commonly used words**” - displays the most common words in the text, and “**7. Return back to the main menu**” - returns the user back to the main menu.

If the second option from the main menu, namely - „**2. Play games**”, is selected, a menu with a total of three options - two games and an option to go back are displayed – “**1. Hangman**” - a game almost the same as the game “hangman”, “**2. Riddles**” - displays riddles, the answer to which the user should try to guess, and “**3. Return to the main menu**” - returns the user back to the main menu.

When selecting the last option from the main menu – “**3. Exit**”, the execution of the program ends.

## Table with some of the most important functions

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Purpose | Arguments | Return type |
| readInt | Checks if the input is integer | String | int |
| countEachWordInText | Counts all the words in a text | WORD\*, int | void |
| addWordsInArray | Adds all words from a text in an array | WORD\*, string | int |
| randomInt | Generates a random integer | Int, int | Long int |
| gameHangman | Shows the hangman game | - | bool |
| generateRiddles | Gets a riddle from a file | - | RIDDLE |
| generateRiddleAnswers | Gets the answer for the specific riddle | RIDDLE | string |
| isCorrect | Checks if the answer is correct | String, string | bool |
| playAgain | Asks is another game round is wanted | - | bool |
| mainMenu | Shows the main menu | - | bool |
| textMenu | Shows the text menu | WORD\*, int | void |
| gamesMenu | Shows the games menu | - | void |

## Diagram of the program

## Milestones in the realization

Before we started, we had to make a plan how to make the program. That included:

* Deciding what to do
* Searching for similar ideas
* Programming
* Testing

## Used resources

For the implementation we used Microsoft Visual Studio 2019. For sharing the code between us and working together, we used GitHub. For discussing ideas, roles of the members in the team and other things connected with the organization we used Microsoft Teams and we wrote our plan on a OneNote Document.

## Conclusion

We had only 5 days to make the project. In conclusion, we realized that we have to be well-organized and capable of working under pressure because it is hard to make a user-friendly bug-free program for such small period of time.