

**1.Project explanation:**

This report considers a console version of the popular game “Frogger”. The idea of the game is to help a little frog to cross a busy road and navigate it to a calmer and safer place – the wonderful home swamp!

The player guides the frog using the following key arrows: "↑" for up, "↓" - down, "←" - left, and "→" - right, and each pressing of the respective key of direction causes the frog to hop once in that direction.

The game starts at the bottom of the screen. In general the screen is divided in two: the lower and the upper halves of it represent a heavy traffic road with various motor vehicles: cars, trucks, buses, which move at different speeds. The frog should pass through that traffic chaos in attempt to reach the upper part of the screen. In the middle of the play area is the "Safe Zone" – a place where the little frog can have some rest before continuing its dangerous journey. The frog has only three lives! If it gets hit by a vehicle, it loses one life.

Throughout the game the player receives an encouraging message each time he succeeds in directing a frog get safe and sound to the top of the screen. And the more and better he plays, the more encouraging and funnier messages he gets! But when the frog reaches the top of the screen, the game difficulty changes and it becomes harder for it to cross the traffic next time.

At the upper part of the console along with the encouraging messages, the player gets as well a notification about the number of frogs he managed to save from the traffic hell; information is also given about the level he is at, the number of lives left and the score he gains during the game.

**Brief technical explanation of the most important game implementations:**

The program is written in OOP manner. There are three classes containing objects used for the frog, tunnels and bridges. The Frog class is modified with ISerializable interface with purpose of storing the name and the score of each player in a binary file. There exist a namespace containing various types of cars used in the game. All car classes inherit the methods of the Enemy class and override its drawing method. There are three classes used for GUI purposes. One of the them converts a png picture to ASCII art and another draws the menu with the picture. The third GUI class checks if the player has scored enough points for the high score table. It also prints Game Over or Game Complete depending on the outcome of the game. Finally, we put everything together in the Engine class.

**List of the member's usernames from the student system:**

1. Nelfurion ( Радостин Ангелов )
2. DareDev1l ( Теодор Ханев )
3. nikolai.mishev.5 ( Николай Мишев )
4. ivanvasilev\_90 ( Иван Василев )
5. B3f0r ( Красимир Стоянов )
6. konstantin\_malinov ( Константин Малинов )
7. LSvilenov ( Любомир Свиленов )
8. dobromira.boycheva ( Добромира Бойчева )
9. si\_555 ( Силвия Иванова )

**Project repository:**

**https:// github.com/KrasimirStoyanov/C-Sharp-2-Group-Project.git**