**SPACE EVADER**

**COMPUTER SCIENCE PROJECT**

**INDEX**

1. Introduction
2. UDFs used
3. Concepts involved
4. Outputs
5. Bibliography

**INTRODUCTION**

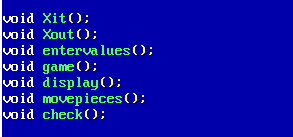
Space Evader is an arcade game in which the aim of the game is reach the portal (0) without being captured by the enemy (\*). The game consists of 10 levels, each level more daunting than the previous one. A player needs to control his character using ‘a’, ‘s’, ‘d’ and ‘w’ keys. The movement of this keys are as follows: -

* ‘a’ key: - Move one step to the left
* ‘s’ key: - Move one step downward
* ‘w’ key: - Move one step upward
* ‘d’ key: - Move one step to the right



**UDFS USED**

1. Xit()- It is a void type UDF, which has been used in the program to enter the details of the player and his performance in a text file.
2. Xout()- It is a void type UDF, which has been used in the program to display the details of any previous player and his performance by using the player’s name.
3. game()- It is a void type UDF, which has been used in the program to initialize certain variables essential to the function of the program and to create an infinite for loop which calls the following functions - entervalues( ), display( ), movepieces( ) and check( ).
4. entervalues()- It is a void type UDF, which has been used in the program to give values to the matrix which acts as the board or platform on which the game is played.
5. display()- It is a void type UDF, which has been used in the program to give values to the matrix which acts as the board or platform on which the game is played.
6. movepieces()- It is a void type UDF, which has been used in the program to give values to the matrix which acts as the board or platform on which the game is played.
7. check()- It is a void type UDF, which has been used in the program to give values to the matrix which acts as the board or platform on which the game is played.



**CONCEPTS INVOLVED**

The program has been made with the help of basic C++ concepts learnt by the students in Grade 11 and 12:

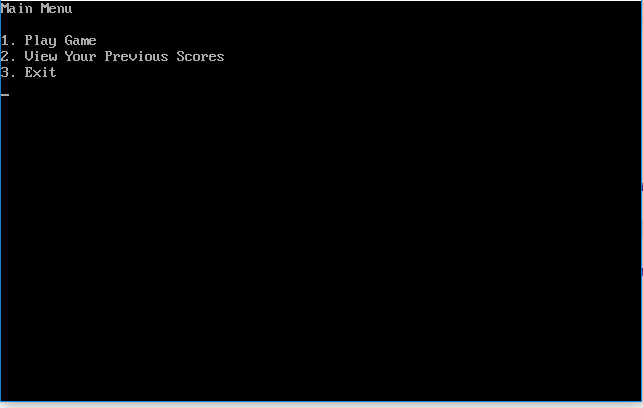
* Class
* File Handling
* Switch case
* For and while loops
* Random function
* Exit function
* 2D array operations
* Function overloading

The project is based on a vast 2D array in which the positions of an element can be changed through input and random functions. This is used to navigate a character through the array to a desired location. This project consists of a menu driven program which comprises of a game. Apart from this, concepts of class, function overloading and file handling have been used in order to calculate, store and display data such as scores, moves taken and points (score/level).

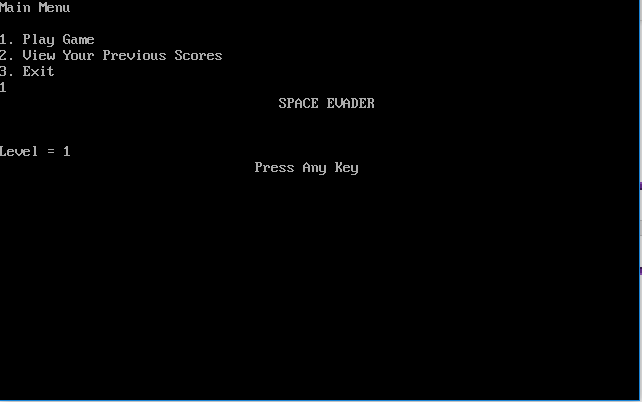
**OUTPUTS**

When the program is run at first, the program displays a main menu consisting of 3 options-

1. Play game (to play the game)
2. View your previous score (to view the previous score)
3. Exit (to exit from the program)



On selecting option 1, the game proceeds to the level 1 of the game. The user needs to press any key to start playing the game.



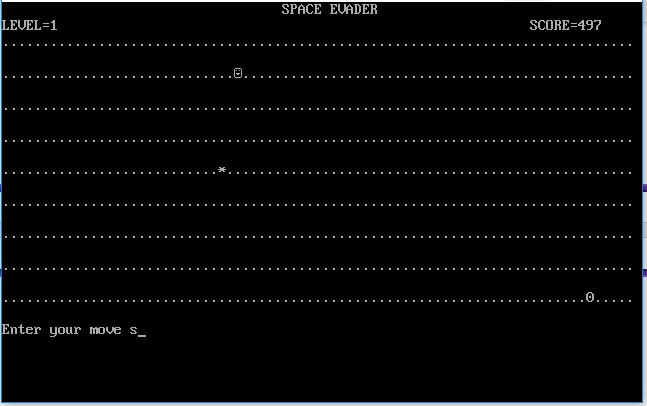
When the game begins, the user must move his character {(char)1} using the a, s, w, d keys in an aim to reach the portal (0) without being captured by the enemy (\*).

The ‘a’ key is to move the character one spot to the left.





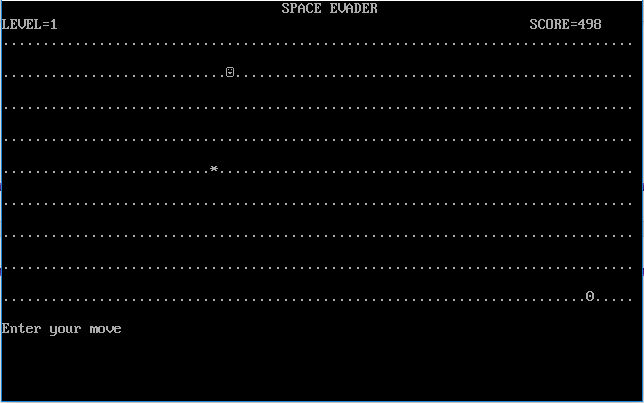
The ‘s’ key is to move the character one spot to the bottom.



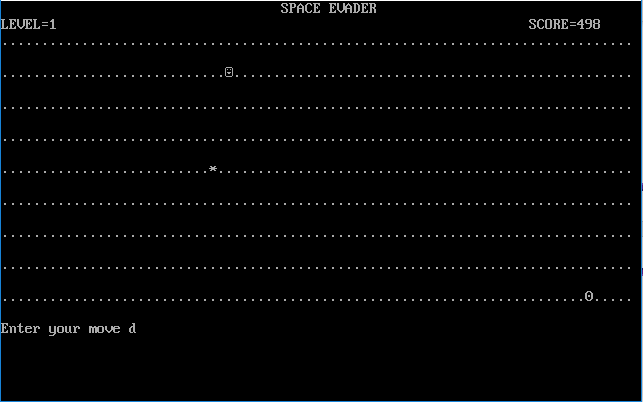


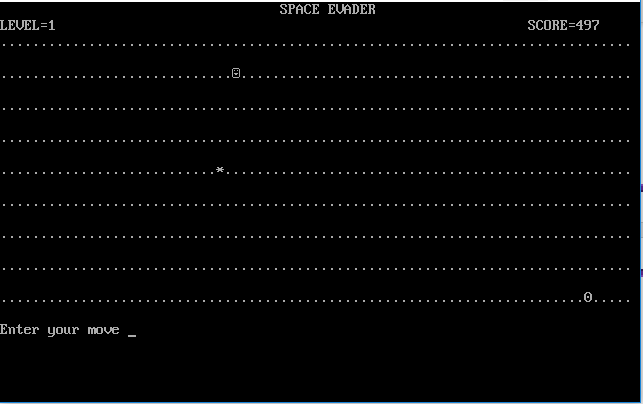
The ‘w’ key is to move the character one spot to the top.



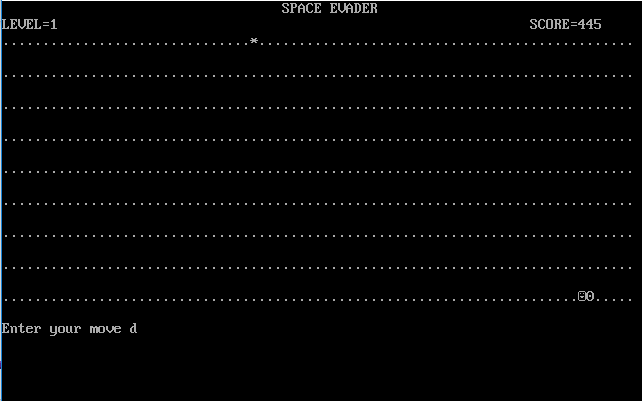


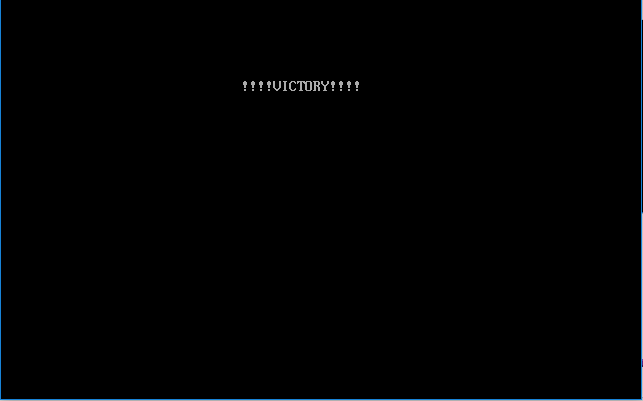
The ‘d’ key is to move the character one spot to the right.





A player wins a level only when he manages to safely bring the character to the portal.





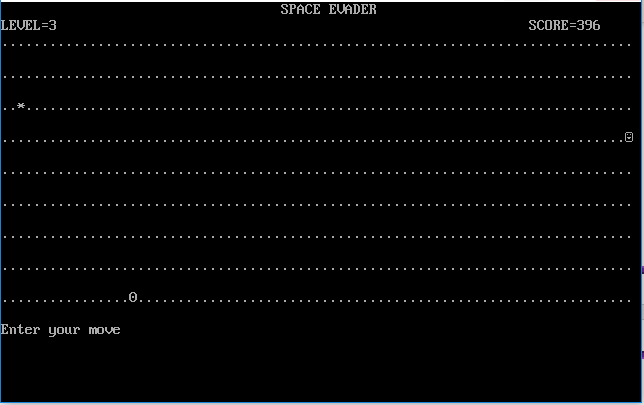
However, a player will lose the game if captured by the enemy

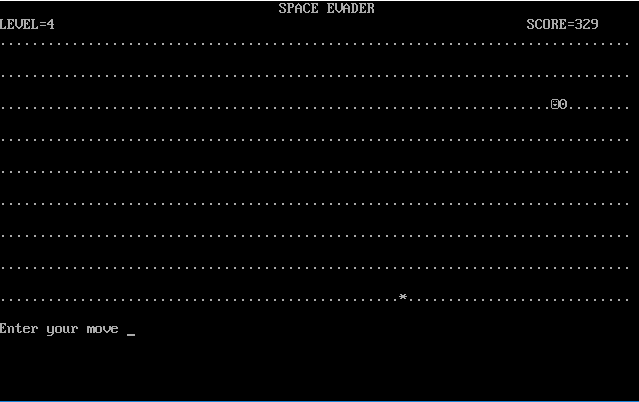


The game consists of 10 levels in total









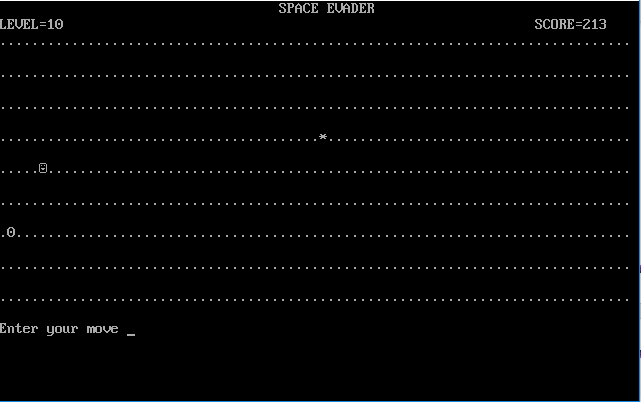




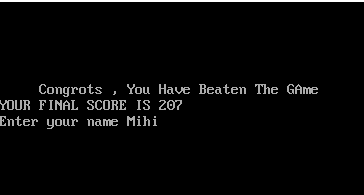




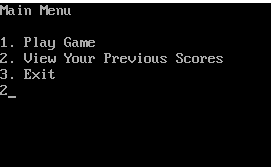




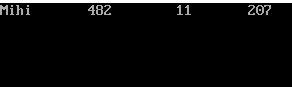
On either quitting, losing or completing the game (winning level 10), the program displays the final score and asks the user to input a 4-character name.



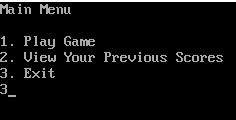
The game then returns back to the main menu.

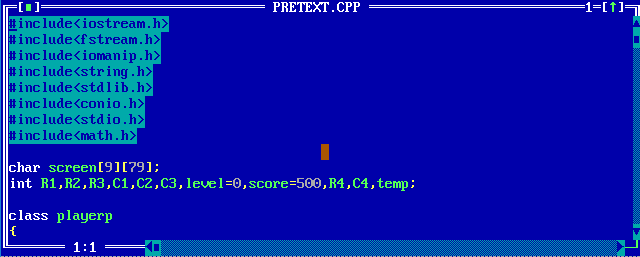


On selecting option 2 from the main menu, the program shows the number of moves taken by the user, the points and score achieved by him/her in the previous attempt.



On selecting option 3 from the main menu, the program exits the game.





**Bibliography**

**Computer Science with C++** by Sumita Arora.

[www.wikipedia.org](http://www.wikipedia.org)

[www.google.com](http://www.google.com)

[www.cbse.nic.in](http://www.cbse.nic.in)