Report on game project using C : Snake

Submitted by :

Fairooz Nawar

201914033

CSE - 19

Course Code : CSE - 105

Submitted to :

**Lt Cdr Arnab Pal**

**Overview of the project :**

Snake game is a popular game in specially old mobile phones. It can be easily developed using C program. To build this project we require some very basic knowledge and understanding of c language and its syntax. Mainly concept of loop should be very clear. Also 2D array and different functions are built to develop the game.

In this game, we have used ‘A’,’D’,’S’ and ’W’ keys for controlling the direction of the snake. Here we use a frog as a food for snake and provide it in different coordinates of the game field. Every time the snake eats the food, its length will by increased by one element along with the score. If the snake touches its body then the game overs.

**How the project is different from other similar projects :**

* We have used ‘D’, ‘A’, ‘W’, ’S’ keys to direct the snake respectively to right, left, up and down instead of arrow keys.
* We have used special characters to make the snake and frog. We have also used different characters to make the border of the play field than other snake games.



* We have used a beep sound to notify the player when the game is over.

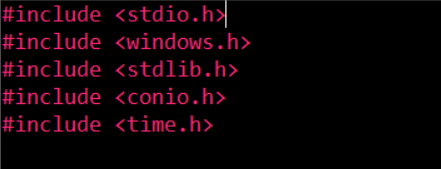
**Features of the project :**

* Creating field of play
* Snake body
* Game loop (Continuity of the game)
* Printing frogs randomly in different places of the screen
* Movements of the snake in different directions
* Continuous movement of the snake
* Handling the length of snake
* Updating tail after eating a frog
* Fixing other errors
* Game over conditions
* Scores and Highscore

**How the components interact with each other :**

We use a number of different functions to make connections between the components. We also use many built-in functions. For this we need some header files which are built-in C language.

They are - stdio.h, windows.h, stdlib.h, conio.h, time.h .



The used funtions are -

void snakeInitialization()

void print()

void ResetScreenPosition()

Random()

int getch\_noblock()

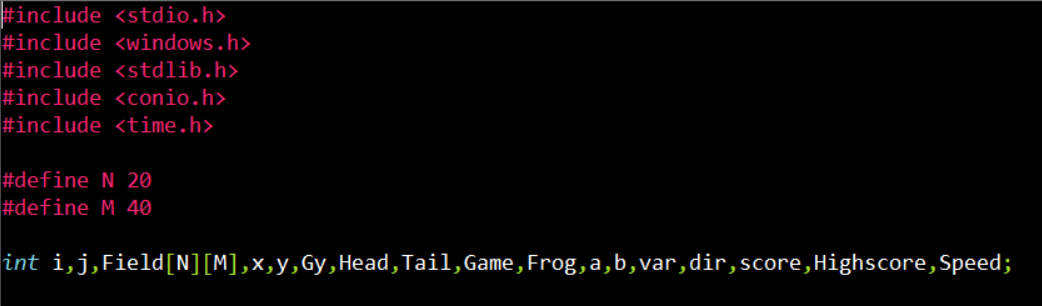
void movement()

TailRemove()

void GameOver()

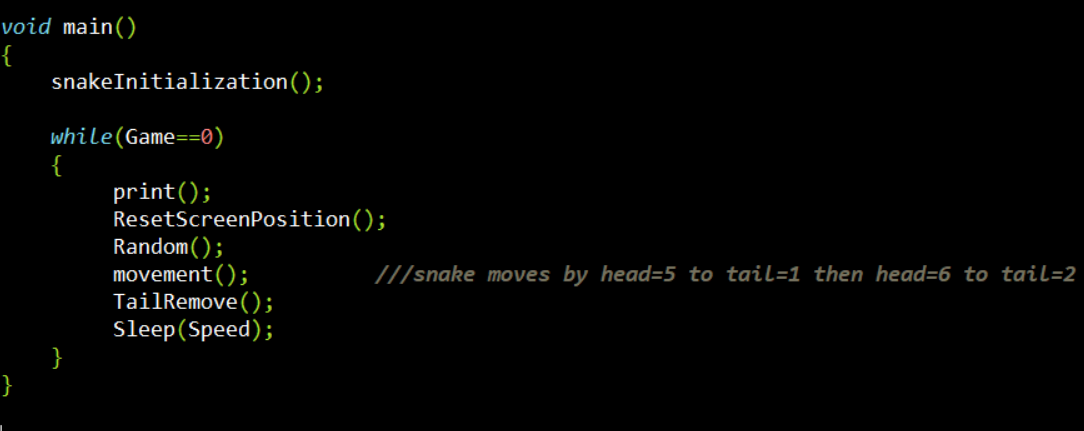
All these functions are called in the main function except GameOver() and getch\_noblock(). GameOver()is called in the movement() function and getch\_noblock() is called in the movement() and the GameOver() function.

**Code for important component with explanation**

1. 

At first we have included all the header files to access the library functions. Then we have defined the length and width of the field of play which actually refers to the row and column of the 2D array field we have made in our game. Here number of column is double of number of row because in the console the length of a character is double of its width. Then we have declared all the global variables we need to implement the game.

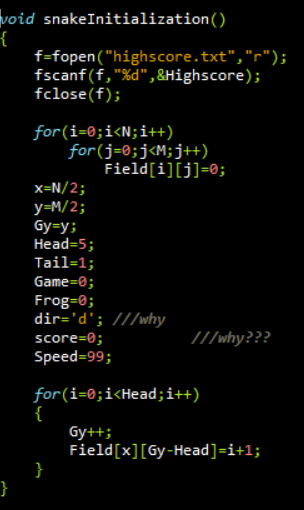
2.



In the main function, we have called all the functions. snakeInitialization() has called only once when the game starts at the beginning.

Then other functions are called in every moment of the game until the game ends. We have done this using a while loop. As well as the value of the variable game is 0, the while loop exists.

3. snakeInitialization()

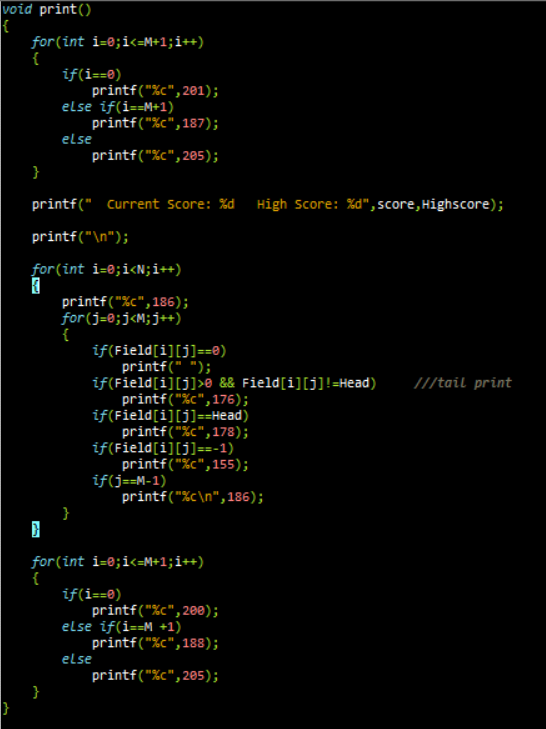


This function sets all the default values for our snake so that the snake holds some default properties whenever the game starts. In this function firstly we have set all the field of play of the array we created to 0, because in the beginning the field is empty. We have done this using a nested for loop.

Then we have initialized all the value for the snake and frog. We have set the coordinate of our snake at the center of the field using the variables x and y. And we initiated the value of Head to 5 and Tail to 1. Other points between head and tail are the body of the snake. Here at first there will be no frog on the screen so the value of Frog is also 0. Game has initialized to 0 because it means the game is on. Score also initialized to 0 at the beginning. And the dir is ‘d’ because we want when the game starts, our snake will be moving to the right direction.

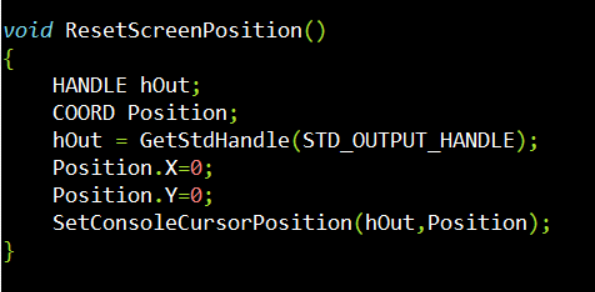
We have set the body of the snake using a for loop. For this we used another variable Gy which has initialized as y, and we incremented this by 1 inside the loop. Thus the body of snake has created.

4. print()



In this function we have printed everything that will be visible on our screen. Firstly, we have printed a perfect square to create the field of play using some special characters. The ASCII values of these characters are 201, 205, 187, 186, 200, 188. Then we have printed the head of the snake using the character of ASCII value 178, and the other parts of the snake using the character of ASCII value 176. When the frog will be displayed on the screen, the value of Frog will be -1 and it will displayed by using the character of ASCII value 155.

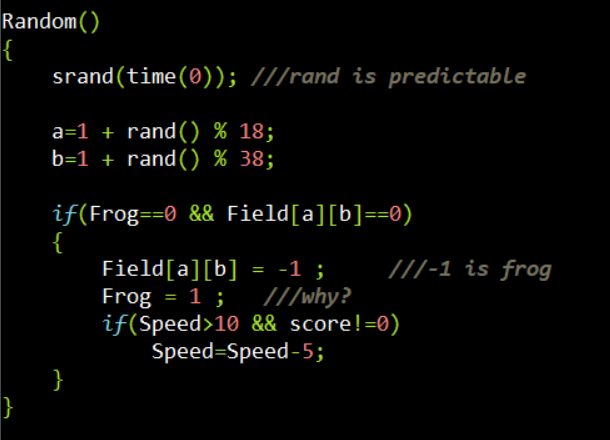
5. ResetScreenPosition()



This function is for setting the cursor position each time to the top of the screen so that the game looks like the frame is not moving and it is continuously on the same position. We needed to include some header files like windows.h and stdlib.h. Inside this function, we have called another two built-in functions -

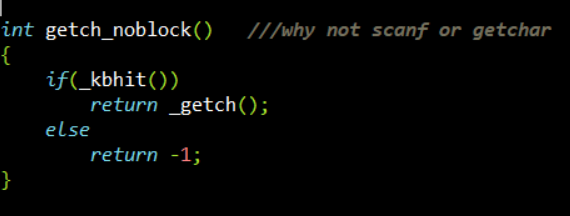
GetStdHandle() and SetConsoleCursorPosition()

6. Random()



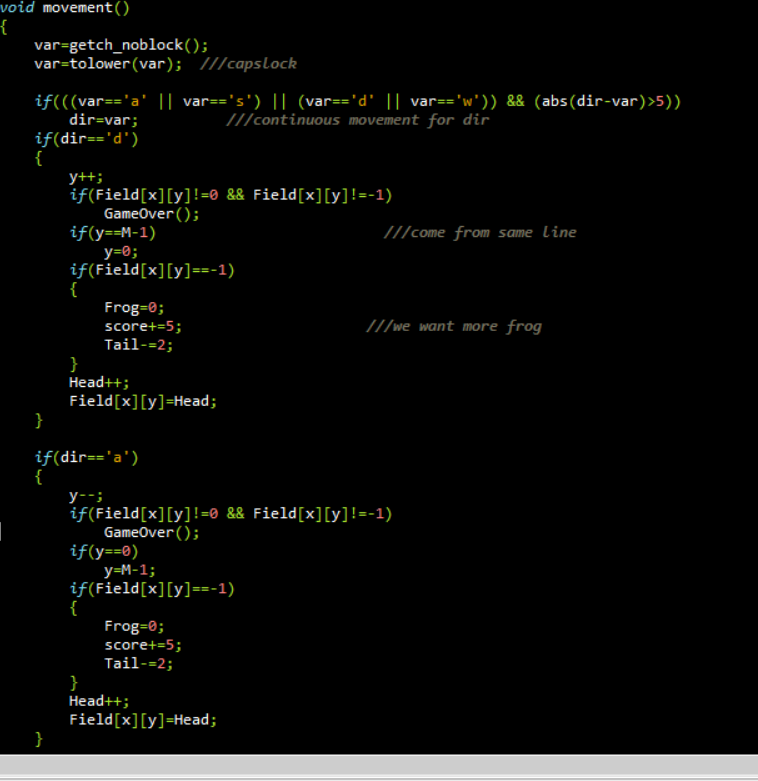
This function has used to generate a frog on the different co-ordinate of the screen randomly. For printing the frog, the variable is Frog. a and b are two variables to hold the random position of the frog. We want our frog to be in the random position and a single frog at a time. So we made this function to do this. Again we have increased the speed of the snake everytime when it eats up a frog in this function.

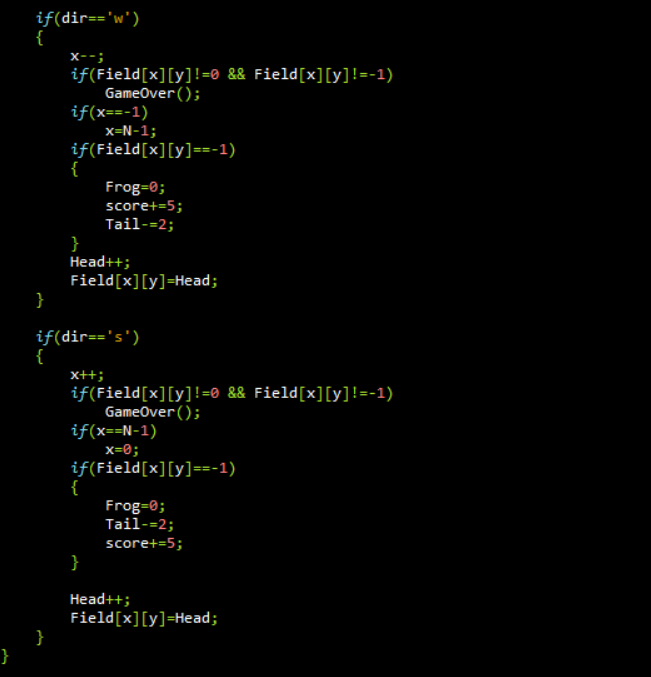
7. getch\_noblock()



When the player press a key in the keyboard, this function receives it and return the integer value of the character. Else it returns -1.

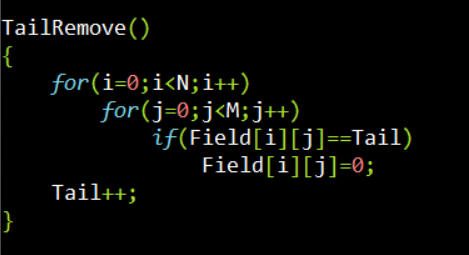
8. Movement()





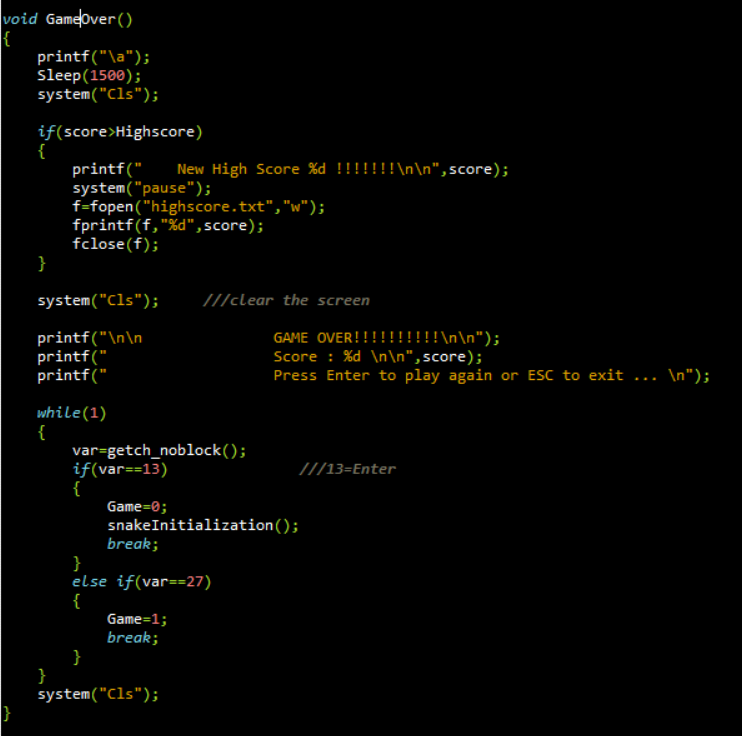
This is one of the most important function of the game. In this function, we have controlled the direction of the snake to different coordinates using ASCII values of the keys ‘D’, ‘A’, ‘W’, ‘S’. We used the tolower function to make the input to the lower case letter as a result we will not face any difficulty if the caps lock is on or not. We called the Gameover() function in this section followed by certain conditions. We also handled the score, the continuity of the snake and the increment of the length of the snake after eating a frog in this function.

9. TailRemove()



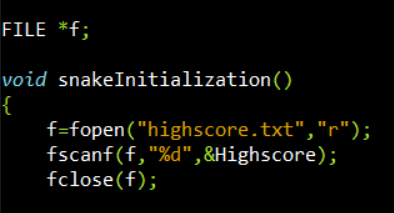
This function has used to maintain the length of the snake otherwise only the head of the snake moves forward, not the tail. So that the length doesn’t remain constant. This function does this work.

10. GameOver()



This function controls the result of the game. The score and the highscore controls in it and also printed on the console. A player can also decided if he starts a new game or not. By pressing Enter key, he can play again and by pressing ESC , he can exit the game. When the game will be over, there will be a beep sound to notify the player.

10. File



We have also created a text file named "highscore.txt" to hold the record of highscore that makes the game more interesting.

**Difficulties we have faced when implementing the code :**

We have faced some problem to set the cursor position each time on the top of the screen as we were unaware about how to do this. So we took help from the internet to solve it. We also faced some problem to implement the game over conditions and we got some errors in our code. Later we fixed it with the help of youtube.

**Resources :**

♣Youtube: Snake game in C tutorial.([https://www.youtube.com/watch?v=stxRI8OUW\_Y&list=PL3pKgZpYZLjmNpbeZrcL8pBSq3YPb-8G6&index=14](https://www.youtube.com/watch?v=stxRI8OUW_Y&list=PL3pKgZpYZLjmNpbeZrcL8pBSq3YPb-8G6&index=14&fbclid=IwAR3FS3L5a61xWmH1Gf26HCce4U_rpocbDSIBXwB3hHbwTcZcIqSpc1EfVn0))  
♣ Google: For Codes Stack: ([https://codereview.stackexchange.com/questions/42602/snake-game-in-c](https://codereview.stackexchange.com/questions/42602/snake-game-in-c?fbclid=IwAR0dNko253T_sQkOm_clil9W2jU8pMcL-OsnE9QPt6pxXnreE2h7KjMRVZ0))