# 

# 

# 

# SNAKE GAME

# 

# 

# 

# 

# 

# **TABLE OF CONTENTS**

[**Introduction**](#_ivnj7pgrnwqg) **3**

[Problem Statement](#_2zvdexz5e47) 3

[Project Description](#_n9pd7wtr49qh) 3

[**Requirements**](#_8rsarbwq5lx5) **4**

[Hardware Requirements](#_fal0ukv9ybla) 4

[Software Requirements](#_q01603pfggvd) 4

[Functional requirements](#_vp9aywpx3woa) 4

[**Architecture**](#_wiygzlxy8hlv) **5**

[**Test Plan**](#_w32ch3f4fe3l) **6**

[**Test Cases**](#_7gyrt3vrsl4k) **6**

[**Expected Results**](#_hyjwucsgsnux) **8**

[**Conclusion**](#_kpwou6hlo05k) **8**

# 

# **Introduction**

## **Problem Statement**

To develop a console-based Snake game using C programming.

## **Project Description**

This console-based game is a simple application, similar to the Snake game introduced by Nokia in 1997. The player uses the “w”, “s”, “d” and “a” arrow to move the snake. He/she attempts to eat items by running into them with the head of the snake. Each item eaten makes the snake longer. The player loses the game when the snake runs into the screen border or itself. It gives you an idea of what you can achieve with a relatively simple c program.

# **Requirements**

## ***Hardware Requirements***

* Hard disk : 32 GB
* RAM : 128 MB

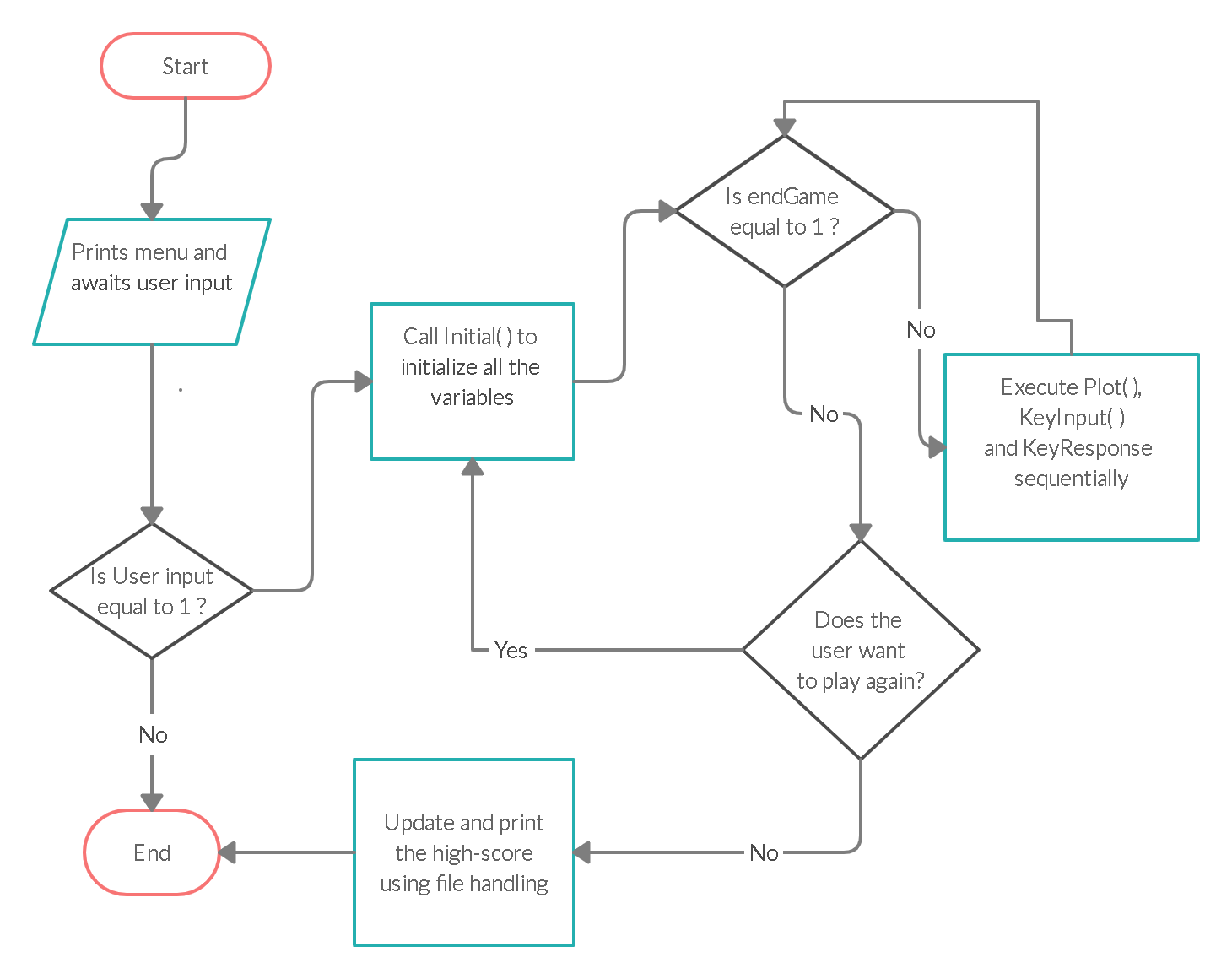
## ***Software Requirements***

* Operating System : Ubuntu
* IDE : Codeblocks
* Programming Language : C
* Compiler : GCC

## ***Functional requirements***

* Main menu
* Console-based gameplay
* The snake must appear to move around the screen
* The snake must turn in response to user input
* The snake will increase in length if it eats food
* The snake will die if it runs over itself
* The snake will die if it runs into the walls
* The snake never stops moving
* The snake should respond to keyboard inputs
* Option to quit during gameplay

# **Architecture**



# 

# **Test Plan**

|  |  |  |
| --- | --- | --- |
| SL NO | Module | Description |
| 1 | Display Menu | The main menu is displayed |
| 2 | Start gameplay | Gameplay starts when the player taps on the spacebar. |
| 3 | Display snake head | The snake is visible on-screen. |
| 4 | Display food | The food is visible on-screen. |
| 5 | Response to keyboard inputs | The snake moves according to the user’s input. |
| 6 | Snake’s tail growth | The snake grows its tail when it eats an item. |
| 7 | Score display | The player’s score is displayed when he loses the game. |
| 8 | Quit game | The user can quit the game by tapping on ‘X’. |

# 

# **Test Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| SL NO | Module | Expected Result | Actual result |
| 1 | Display Main menu | The main menu must be displayed when the player starts playing the game. | The main menu is displayed on the screen. |
| 2 | Start gameplay | The gameplay should start when the user taps on the spacebar. | The player proceeds to the play screen. |
| 3 | Display snake head | The snake must be visible to the player. | The player sees the snake on-screen. |
| 4 | Display food | The target food must be visible to the player and it should be at a point on the screen, separate from the snake’s body. | The player sees the food on-screen. |
| 5 | Response to keyboard inputs | The snake must move according to the user’s input. | The snake moves left, right, top or bottom when the user taps on ‘a’, ‘d’, ‘w’ or ‘s’ respectively. |
| 6 | Snake’s tail growth | The snake’s tail should grow whenever it eats a food item. | The snake’s tail grows by 1 unit, each time it eats a food item. |
| 7 | Score display | The player’s score must be displayed at the end of the game. | The score is displayed on the screen according to the length of the snake. |
| 8 | Quit game | The user must be able to exit the game . | The game exits when the user taps on ‘x’. |

# **Expected Results**

* The user must be able to play the game without facing any glitch.
* The snake must move according to the player’s inputs and grow according to the logic of the game.
* The application must pass all the test cases mentioned above.
* The user must be able to quit the game when he wants to.

# **Conclusion**

Creating a game is a truly rewarding creative process that requires skills in game design, graphic design, and programming. Most of the time these talents don’t exist all together in one person. This project is a simple one. But its principles can be extended to more complex ones.

# 