

|  |
| --- |
| SNAKE GAME IN C |
|  |
| January 1  Institute of Engineering and Management, Kolkata  Authored by:  1. Rohan Das 2. Soham Maitra 3. Saheli Sau 4. Rounak Ganguly |



# Abstract

# For this project, we chose to create a Snake Game which we used to play in old mobiles.

# The purpose of the project was to gather knowledge about different header files in C Language, their different functions and how to use them.

# Using standard functions we were able to develop and implement most of the features, though it is not exactly perfect. We’ll discuss the drawbacks of the application in this report itself.

# Introduction

# Properly structured, group projects can reinforce skills that are relevant to both group and individual work, including the ability to:

# 1. Break complex tasks into parts and steps.

# 2. Plan and manage time.

# 3. Refine understanding through discussion and explanation.

# This report aims to provide a detailed look at the resulting application.

# 2.1) Background

# At the beginning of the project we had no prior experience of game development in C Language. So a huge amount of time was invested investigating, understanding and testing smaller bits of functionality, as well as looking at the alternating implementations to figure out what we thought worked best.

# 2.2) Project Brief

# The main goal of this project was to develop a simple game to play and strengthen our C Language skills.

# Application Details

# We’ll completely describe each and every line of this project.

# Firstly, we’ll discuss about some functions which are not generally used in programming, they are:

# kbhit()

# rand()

# system(“cls”)

# First lets discuss about the kbhit() function, the full name of the function is keyboard hit. It is present in conio.h header file and used to determine if a key has been pressed or not. If a key has been pressed it returns a non-zero value otherwise it returns 0.

# Now lets discuss about the rand() function. The name clearly suggests that it is used to generate random numbers in the range [0, RAND\_MAX).

# rand()%100 -> It generates a Random Number between 0 to 99

# rand()%100 + 1 -> It generates a Random Number between 1 to 100.

# system(“cls”) -> It clears the Console Screen.

# Firstly, we’ve initialized these header files.

# 

# Secondly, we’ve initialized these global variables.

# 

# Now, we’ll discuss about the various functions that we created.

# The first function is frame(). This functions is takes nothing, returns nothing type. The source code is given here:

# 

# The first line of the function i.e, Line No. 15 clears the console screen. The first if statement in Line No. 21, prints the border. Now let’s look into the else block of Line No. 23. The first if statement in the block prints the head of the snake. Then the else if statement prints the Food of the snake. Now the exciting part is the else block at Line 28. It is the condition that when the snake eats food its length increases. There is an integer variable ch which is used to prevent the right side of the frame from breaking.

# Now we’ll define the setup() function. Here the x and y denotes the head of the snake. Initially, we’re setting the head of the snake at the middle of the frame i.e, at (height/2,width/2). Now, in Line 54 we’re generating random food within the border of the frame. Here x\_food and y\_food denotes the x and y coordinate of the food respectively. If x\_food == 0; it coincides with the border of the frame. In that case we’re generating again the coordinate of the food. Same in the case of y\_food.

# 

# Now we’ll discuss about the input() function. This function contains only one if block. If we press any key from the keyboard, then the if block is executed. Then we’re going inside the swich() block. Here, we can press only ‘w’ or ‘s’ or ‘d’ or ‘a’ or ‘x’. Let’s discuss about the keys.

# w-> Snake will move upwards.

# s-> Snake will move downwards.

# d-> Snake will move to the right.

# a-> Snake will move to the left.

# x-> Stop the Game.

# 

# Now we’ll discuss about the main function of the game i.e, the logic() function. Here the prevX, prevY, countTail variables are used to increase the length of the snake after eating food.

# Next, the switch block with argument flag is used to move the snake’s head according to the users input.

# After that, the if block represents that if the head of the snake hits the frame the game is over.

# Then the for loop represents that if the head of the snake hits the body of the snake the game is also over.

# Lastly, the if block generates another random food after the current food is being eaten. It also increases the score by 10.

# 

# 

# Now we’ll discuss about the main function. In the first line in the main function, system(“color 3c”) turns the background color of the console screen to blue. Then we mention the instruction of the games. After that, in while loop we call the functions frame(), input(), logic() function one after another. When the game is over the control screen is closed after displaying the score.

# 

# Difficulties Faced in Making the Project:

# The main problem of making the game is that we faced the blinking of the console screen. Though we try to overcome it with the for loop nested in a for loop in main function, it’s not very efficient. We also used delay function instead of that, but its same as before.

# Conclusion

# It was an exciting project to work on and there is a lot I learnt from it, above and beyond its original scope. I was able to study and train myself on development in an environment that was almost completely foreign to me, which I believe I have been reasonably successful with.

# Acknowledgement:

We would like to express our special thanks of gratitude to my teacher Moumita Ma'am as well as our principal Swagatam Sir who gave us the golden opportunity to do this wonderful project on the topic, which also helped us in doing a lot of research and we came to know about so many new things we are really thankful to them.