

SNAKE-MANIA

C++ PROJECT

Submitted in the partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

CS with specialization in AI and

ML

Submitted by:

Kunal

20BCS6278

Under the Supervision of:

Mr.Gurpreet Singh Panesar



**CHANDIGARH
UNIVERSITY**
Discover. Learn. Empower.

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
APEX INSTITUTE OF TECHNOLOGY**

CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,

PUNJAB

JULY 2021

DECLARATION

I, '**Kunal**', student of '**Bachelor of Engineering in Computer Science with specialization in AI and ML**', session:**2020-2024**, Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Punjab, hereby declare that the work presented in this Project Work entitled '**Snake-Mania**' is the outcome of our own bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Kunal

Candidate UID:20BCS6278

Date: 27/07/2021

Place: Jaipur

ACKNOWLEDGEMENT

I would like to acknowledge all those without whom this project would not have been successful. Firstly, I would wish to thank our Computer Science teacher **Mr. Gurpreet Singh Panesar** , who guided me throughout the project and gave immense support. He made us understand how to successfully complete this project and without him, the project would not have been complete.

This project has been a source to learn and bring our theoretical knowledge to the real-life world. So, I would really acknowledge his help and guidance for this project.

I would also like to thank my parents who have always been there whenever needed.

Once again, thanks to everyone for making this project successful.

Table of Contents

Title Page	i
Declaration of the Student	ii
Acknowledgement	iii
1. INTRODUCTION AND DETAILS	
1.1 Overview of C++	
1.2 Inroduction to Project	
1.3 Hardware Requirements	
1.4 Software Requirements	
1.5 Specification	
2. PROJECT FORMULATION	
3. OBJECTIVES	
4. METHODOLOGY	
4.1 Header Files	
4.2 Flowchart	
5. SOURCE CODE	
5.1 main.cpp	
5.2 GUI.h	
5.3 gotoxy.h	
5.4 color.h	
4. OUTPUT	
5. CONCLUSIONS	
6. REFERENCES	

1 INTRODUCTION AND DETAILS

1.1 C++ Overview

C++ was developed by Bjarne Stroustrup at AT & T Bell Laboratories in Muarry Hill, New Jersey (USA) in 1983. C++ is an OOP i.e. Object Oriented Programming, which allows programmers to develop large and complex applications. The OOP languages existed before C++ were slow and inefficient.

So, Bjarne Stroustrup who was a great admirer and master of C and SIMULA 67, combined the features of both the languages into more powerful language. This combination of features of SIMULA 67 and C resulted in a new language called **C with Classes** by Bjarne in 1979. **C with Classes** lacked some OOP features. Therefore some features and ideas were taken from ALGOL 68 (Algorithmic Language). Thus it resulted in C++. The name C++ is credited to Rick Mascitti who suggested this name and was first used in 1983. It is named C++ not C+ or ++C because C+ has been used as the name of an earlier unrelated language and C++ is more commonly used than ++C. Moreover, it is not named D language because it is an extension of C.

1.2 Introduction to Project

We all are stuck at home at the time of global pandemic, and are unable to enjoy the games in outside world. Here in this project we focused on main making your brain active with a game of Snake . The idea behind the project is to give the user a sense of suspense and joy with a competition The Snake-Mania is a game written in C++ based on the classic Nokia Snake game which has been around since the earliest days of home computing and has re-emerged in recent years on mobile phones.

It isn't the world's greatest game, but it does give you an idea of what you can achieve with a language like C++, and perhaps the basis by which to extend the principles and create more interesting games of your own.

1.3 Hardware Requirements

- Desktop Computer or Laptop
- Keyboard

1.4 Software Requirements

- C++ Compiler (Dev C++ Recommended)
- OS : Windows 7/8/10

1.5 Specifications

- Can run on a low end PC
- No Internet Connection Required
- Game Difficulty increases as score increases
- Easy to play

2 PROJECT FORMULATION

- To create an arcade snake game called “Snake Mania” using C++ which will run on the console window .
- Helps us to implement C++ programming in games
- Powerful tool to help children develop certain life skills
- Improve language and math skills
- Improve hand-eye co-ordination and help

3 OBJECTIVES

- Snake-Mania is a simple console application with very simple graphics. In this project, you can play the popular “Snake Game” just like you played it elsewhere.
- Foods are provided at the several co-ordinates of the screen for the snake to eat. Every time the snake eats the food, its length will be increased by one element along with the score.
- The control of this game are as follows :-
 - ✓ W – UP
 - ✓ S – DOWN
 - ✓ A – LEFT
 - ✓ D – RIGHT

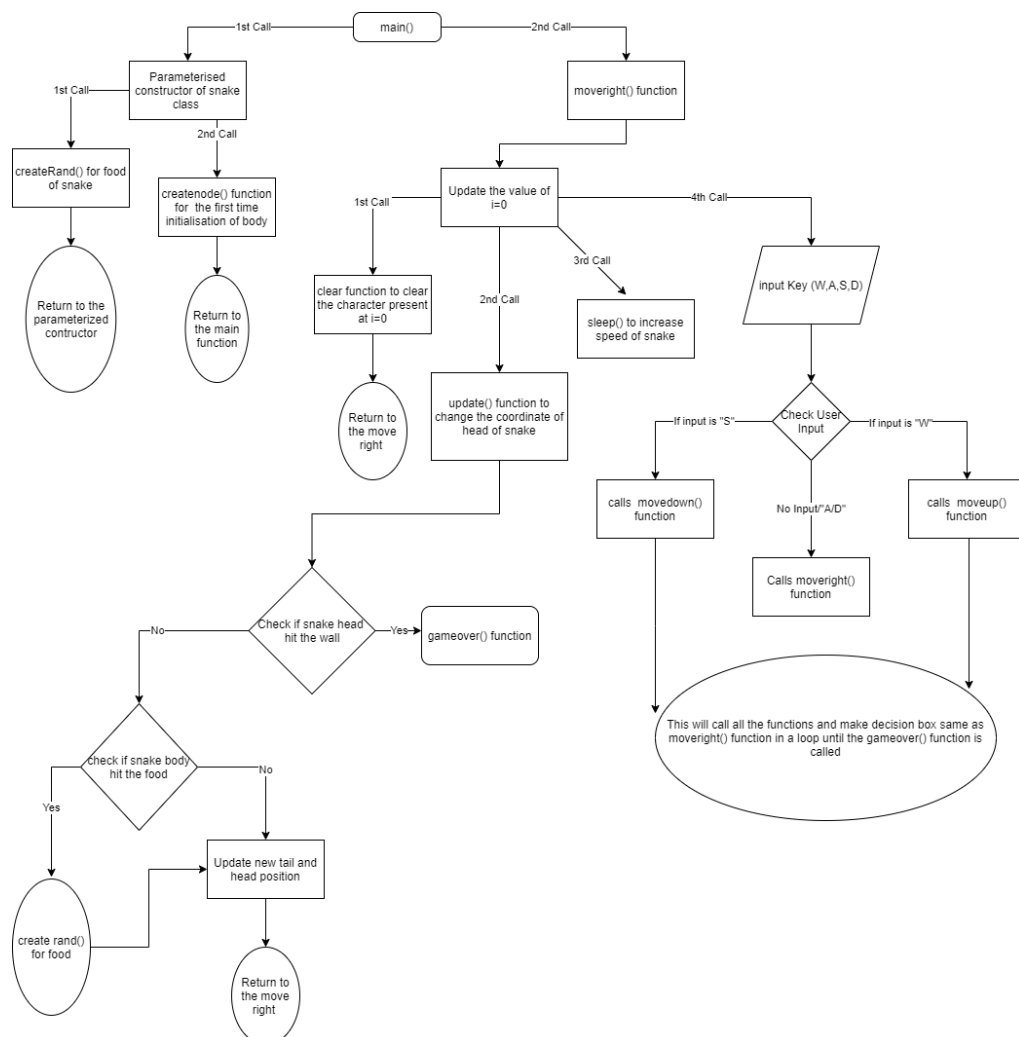
4 METHODOLOGY

4.1 Header Files

We've divided the project in four parts :

- GUI.h – For the background of our game which include play area borders and scoreboard portion .
- gotoxy.h – For moving the cursor to specific co-ordinates of the console screen .
- color.h – For coloring the background console , snake and scoreboard .
- Main.cpp – For the main logic of program and loops .

4.2 Flow Diagram



5 SOURCE CODE

5.1 main.cpp

```
#include<stdio.h>
#include "GUI.h"
#include<conio.h>
#include "color.h"
#include "gotoxy.h"
#include<stdlib.h>
int posx=0;
int posy=0;
int sizes=6;
int prizex=0;
int prizey=0;
int score=0;
int lastx=0;
int lasty=0;
int time=120;
class node
{
public :
    int place;
    int nodex;
    int nodey;
    void createRand()
    {
        time=time-4;
        sizes++;
        score++;
        prizex=rand()%67;
        if(prizex<10)
            prizex=13;
        prizey=rand()%24;
        if(prizey<4)
            prizey=14;
```



```

    gotoxy(70, 8);
    printf("Score = %d", score);
    gotoxy(70, 9);
    printf("Snake = %d", sizes);
    gotoxy(prizex, prizey);
    printf("%c", 489);
}
void GameOver()
{
    while(true)
    {
        gotoxy(20, 20);
        printf("\t\tGameOver, Your Score = %d", score);
        gotoxy(20, 21);
        printf("\t\tPress Any Key to Exit");
        getch();
        exit(EXIT_FAILURE);
    }
}
int update(int x, int y)
{
    if(x>67||x<10||y>22||y<3)
        GameOver();
    else if(x==prizex&&y==prizey)
        createRand();
    gotoxy(lastx, lasty);
    printf("o");
    gotoxy(x, y);
    nodex=x;
    nodey=y;
    posx=x;
    posy=y;
    printf("@");
    lastx=x;
    lasty=y;
}
void CreateNode(int x, int y, int i)
{
    place=i;

```

```

        nodex=x;
        nodey=y;
        gotoxy(nodex, nodey);
        printf("%c", 489);
    }
void Clear()
{
    gotoxy(nodex, nodey);
    printf(" ");
}
};
class snake
{
public:
    node n[100];
    int i=0;
snake(int x, int y, int s)
{
    n[0].createRand();
    sizes=s;
    posX=x+5;
    posY=y;
    for(int i=0;i<sizes;i++)
        n[i].CreateNode(x+i, y, i);
}
void move_right()
{
    if(i==sizes)
        i=0;
    n[i].Clear();
    n[i].update(++posx, posY);
    i++;
    Sleep(time);
    if(GetAsyncKeyState('W'))
        move_up();
    else if(GetAsyncKeyState('S'))
        move_down();
    else if(GetAsyncKeyState('D')){ }
    else if(GetAsyncKeyState('A')){ }
}

```

```

    move_right();
}
void move_left()
{
    if(i==sizes)
        i=0;
    n[i].Clear();
    n[i].update(--posx, posy);
    i++;
    Sleep(time);
    if(GetAsyncKeyState('W'))
        move_up();
    else if(GetAsyncKeyState('S'))
        move_down();
    else if(GetAsyncKeyState('D')){ }
    else if(GetAsyncKeyState('A')){ }
    move_left();
}
void move_up()
{
    if(i==sizes)
        i=0;
    n[i].Clear();
    n[i].update(posx, --posy);
    i++;
    Sleep(time);
    if(GetAsyncKeyState('D'))
        move_right();
    else if(GetAsyncKeyState('A'))
        move_left();
    else if(GetAsyncKeyState('W')){ }
    else if(GetAsyncKeyState('S')){ }
    move_up();
}
void move_down()
{
    if(i==sizes)
        i=0;
    n[i].Clear();

```

```

        n[i].update(posx, ++posy);
        i++;
        Sleep(time);
        if(GetAsyncKeyState('D'))
            move_right();
        else if(GetAsyncKeyState('A'))
            move_left();
        else if(GetAsyncKeyState('W')){ }
        else if(GetAsyncKeyState('S')){ }
        move_down();
    }
};

```

```

int main()
{

```

```

    SetColor(15);
    printf("-----\n");
    printf("-----Snake-Mania-----\n");
    hidecursor();
    GetFrame();
    gotoxy(70, 19);
    printf("Developers : ");
    gotoxy(69, 20);
    printf("Kunal");
    gotoxy(69, 21);
    printf("KunalGaurav");
    gotoxy(69, 22);
    printf("HimanshuGidwani");
    SetColor(10);
    snake s(10, 10, 7);
    s.move_right();
    return 0;
}

```

5.2 GUI.h

```
1  #ifndef GUI_H_INCLUDED
2  #define GUI_H_INCLUDED
3  #include "windows.h"
4  #include<stdio.h>
5
6  void GetFrame()
7  {
8      int i;
9      printf("\t_____");
10     for(i=0;i<20;i++)
11     printf("\n\t||_____||");
12     printf("\n\t||_____||");
13 }
14 #endif // GUI_H_INCLUDED
```

5.3 gotoxy.h

```
1  #include<windows.h>
2  #include<stdio.h>
3
4  COORD coord={0,0};
5
6  void gotoxy(int x,int y)
7  {
8      coord.X=x;
9      coord.Y=y;
10     SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),coord);
11 }
12 void hidecursor()
13 {
14     HANDLE consoleHandle = GetStdHandle(STD_OUTPUT_HANDLE);
15     CONSOLE_CURSOR_INFO info;
16     info.dwSize = 100;
17     info.bVisible = FALSE;
18     SetConsoleCursorInfo(consoleHandle, &info);
19 }
```

5.4 color.h

```
1 #include <windows.h>           //header file for windows
2 #include <stdio.h>             //C standard library
3
4 void SetColor(int ForgC)
5 {
6     WORD wColor;
7     //We will need this handle to get the current background attribute
8     HANDLE hStdOut = GetStdHandle(STD_OUTPUT_HANDLE);
9     CONSOLE_SCREEN_BUFFER_INFO csbi;
10
11     //We use csbi for the wAttributes word.
12     if(GetConsoleScreenBufferInfo(hStdOut, &csbi))
13     {
14         //Mask out all but the background attribute, and add in the foreground color
15         wColor = (csbi.wAttributes & 0xF0) | (ForgC & 0x0F);
16         SetConsoleTextAttribute(hStdOut, wColor);
17     }
18     return;
19 }
```

6 OUTPUT



Figure 1 Play Menu

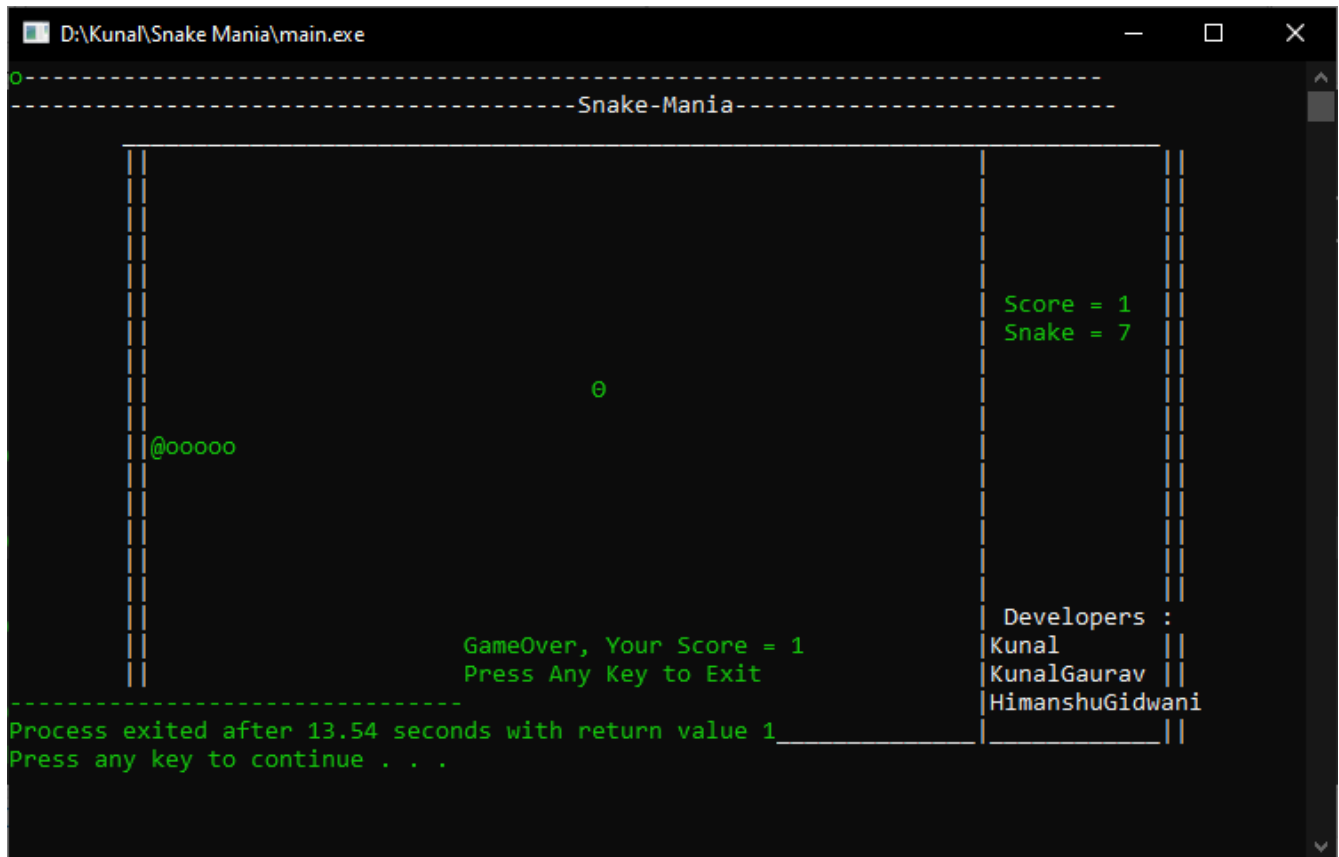


Figure 2 Game Over Menu

7 CONCLUSION

- Snake-Mania based on Classic Nokia Snake game can run on any low end PC having a C++ compiler
- No Internet Connections Required
- By creating this game we got to know how C++ language can be used to create big games like PUBG PC , CS:GO

8 REFERENCES

- <https://www.geeksforgeeks.org/c-classes-and-objects/>
- <https://data-flair.training/blogs/header-files-in-c-cpp/>
- https://www.tutorialspoint.com/cplusplus/cpp_loop_types.htm
- <https://www.codespeedy.com/play-music-in-cpp/>
- <https://devdocs.io/cpp/>
- **Lecture Notes On Object Oriented Programming Using C++** Dr. Subasish Mohapatra
- **Introduction to C++ Programming** by Hans Petter Langtangen

-----END-----