A mini project report submitted on

**“** **SNAKE AND LADDER ”**

For partial fulfilment of the requirement of the degree of

Bachelor of Technology

In Computer Science & Engineering

By

|  |  |
| --- | --- |
| **STUDENT NAME** | **PRN NO.** |
| Ms. Desai Vaishnavi Sanjay | 2021075979 |
| Ms. Desai Soniya Prakash | 2021074861 |
| Mr. Gavali Omkar Vijay | 2021075920 |

Under the guidance of

Prof. P.G. Sanmane

**Academic Year 2022-23**



**Department of Computer Science & Engineering**

**Sant Gajanan Maharaj College of Engineering , Mahagaon**





**NAAC B++ Accredited & ISO 9001:2015 Certified Institute**

**SANT GAJANAN MAHARAJ COLLEGE OF ENGINEERING,**

**MAHAGAON**

**Site- Chinchewadi, Tal – Gadhinglaj, Dist – Kolhapur 416502**



**CERTIFICATE**

This is to certify that, following students have satisfactorily completed the mini project work entitled, “ **SNAKE AND LADDER ”.** This mini project is being submitted for the partial fulfillment of the award of degree of **Bachelor of Technology** in **Computer Science and Engineering** under Shivaji University, Kolhapur, for year 2022-2023.

.

|  |  |  |
| --- | --- | --- |
| **STUDENT NAME** | **PRN NO.** | **SIGNATURE** |
| Ms. Desai Vaishnavi Sanjay | 2021075979 |  |
| Ms. Desai Soniya Prakash | 2021074861 |  |
| Mr. Gavali Omkar Vijay | 2021075920 |  |

**Prof. P. G. Sanmane Mr. S. G. Swami**

**Guide Head of Department**

**Examiner Dr. S. H. Sawant**

**Principal**

# ACKNOWLEDGEMENT

We would like to express our sincerely gratitude towards respected Hon. Founder Chairman Adv. Annasaheb D. Chavan, trustee & all Board of Directors, our beloved Principal Dr. S. H. Sawant for his encouragement & support.

We are very thankful to. Prof. S. G. Swami (Head of Computer Science & Engg. dept) & respected guide Prof. P. G. Sanmane for her constant encouragement and valuable guidance during the completion of this project & valuable co-operation & guidance during this project work & we also thank our mini project coordinator Prof. P. G. Sanmane.

We take this opportunity to thank the entire Teaching & Non-Teaching members of CSE dept. for their co-operation and their helpfulness during this project work. Last but not the least assistance offered by various friends and colleagues related directly or indirectly to this work are also gratefully acknowledged.

|  |  |
| --- | --- |
| **STUDENT NAME** | **PRN NO.** |
| Ms. Desai Vaishnavi Sanjay | 2021075979 |
| Ms. Desai Soniya Prakash | 2021074861 |
| Mr. Gavali Omkar Vijay | 2021075920 |

# ABSTRACT

This is a simple board game which is very easy to understand and play. All the playing rules are the same just like we play in real time Snakes and Ladders. This is a simple 2D multi player game. This game has been build using c language. After starting the game, a Snakes and Ladders board appears, other rules are the same. First, the player has to roll the dice. The main thing in this simple game is that the 1st player just has to press “ 1 ” to roll the dice and the 2nd player press “ 2 ” to roll the dice. The player has to keep on rolling until there’s a possible pawn to move. All the game movements are to be performed manually by the player. A simple board is provided for easy game play. The game play design is so simple that user won’t find it difficult to use and understand.

## INDEX

## Sr No Name Page No

1. Problem Statement 6
2. Introduction 7
3. Software and hardware requirements 8
4. Flowchart 9
5. Block diagram 10
6. Working / Implementation 11
7. Input-output scenario 12
8. Conclusion 16
9. Future work 17
10. Reference: 18

**PROBLEM STATEMENT**

Given a snake and ladder board, find the minimum number of dice throws required to reach the destination or last cell from the source or 1st cell. Basically, the player has total control over the outcome of the dice throw and wants to find out the minimum number of throws required to reach the last cell. If the player reaches a cell which is the base of a ladder, the player has to climb up that ladder and if reaches a cell is the mouth of the snake, and has to go down to the tail of the snake without a dice throw.

## INTRODUCTION

Remember the childhood snake and ladder game which we used to play? Well, in this article we will be discussing the same game and its implementation in programming.

Given a snake and ladder board, find the minimum number of dice throws required to reach the destination or last cell from source or 1st cell. This is done by considering that the player can determine which number appears in the dice being biased. The player rolls the dice and if reaches a base of a ladder then he can move up the ladder to a different position/cell and if he reaches a snake then it brings him down away from the destination cell/position. This problem can be solved using a Breadth-First Search (BFS). So, let us first see the prerequisites briefly. In order to run the project, we must have installed [Dev C++](https://sourceforge.net/projects/orwelldevcpp/) or VS Code  on our PC.

Players roll a die and navigate the board. Landing on a ladder advances a player to a square further up the board, while landing on a snake means they have to go back to a previous square. The aim of the game is to reach the final square. The game is a race that's based on sheer luck, and is popular with children.

## SOFTWARE AND HARDWARE REQIREMENT

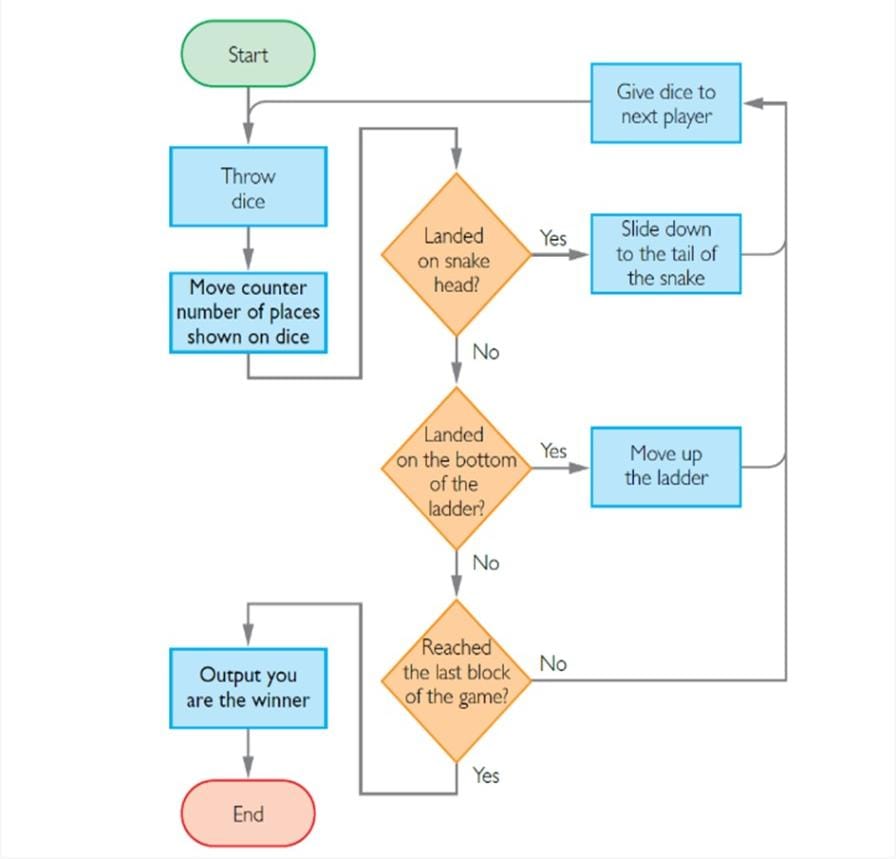
**SOFTWARE REQUIREMENTS**

Language used : C Language  
 Software : Any c Compiler.  
 Operating system : windows 10

**HARDWARE REQUIREMENTS**

Hard disk : Not necessary.  
RAM : 2GB(min)  
Processor : i3  
Memory : 128 GB SSD(min)

## FLOWCHART



## BLOCK DIAGRAM

Win Game

1. Roll Dice
2. Display

Updated Board

Exit

Start

1)Play Game

1. Display Board
2. Enter Names of the player

## IMPLEMENTATION

* Implementing a basic version of the Snake and Ladder game:
* Set up the board:
* Determine the size of the board, usually a square grid.
* Assign numbers to each cell, starting from 1 and going up to the final cell.
* Identify the cells where snakes and ladders are placed and note their start and end points.
* Set up the players:
* Decide on the number of players participating in the game.
* Assign a token or symbol to each player.
* Create a function to simulate dice rolls:
* This function should generate a random number between 1 and 6 (assuming a standard six-sided die).
* Implement the game loop:
* Initialize the current player and set their initial position on the board.
* Enter a loop where each player takes turns until a player reaches the final cell or a winning condition is met.
* In each iteration of the loop:
* Roll the dice for the current player using the dice roll function.
* Move the player's token based on the dice roll.
* Check if the new position is a snake or ladder start point.
* If it is, move the player to the corresponding end point.
* Check if the player has reached the final cell.
* If yes, declare the player as the winner and end the game.
* Move to the next player.
* Display the board
* Visualize the current positions of all players on the board after each turn.
* You can use ASCII art or a graphical interface to represent the board and the player tokens.

## INPUT-OUTPUT SCENARIO

Given a snake and ladder board, find the minimum number of dice throws required to reach destination or last cell from the source or 1st cell. Basically, the player has total control over the outcome of the dice throw and wants to find out the minimum number of throws required to reach the last cell.

If the player reaches a cell which is the base of a ladder, the player has to climb up that ladder and if reaches a cell is the mouth of the snake, and has to go down to the tail of the snake without a dice throw.

For example, the minimum number of dice throws required to reach cell 100 from cell 1 is 13.

Following are the steps:

a) First throw two dice to reach cell number 13 and then ladder to reach 50.

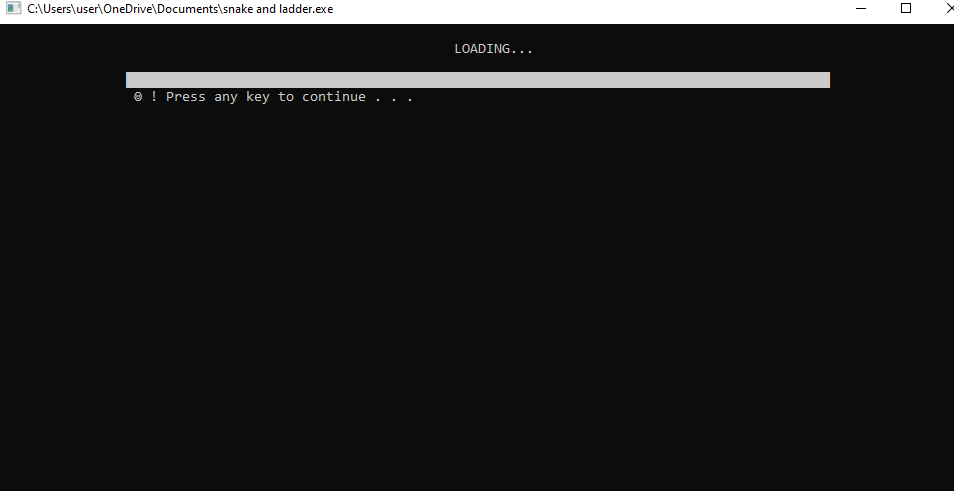
b) Then throw dice to reach 79.

c) Finally through 4 to reach 100.

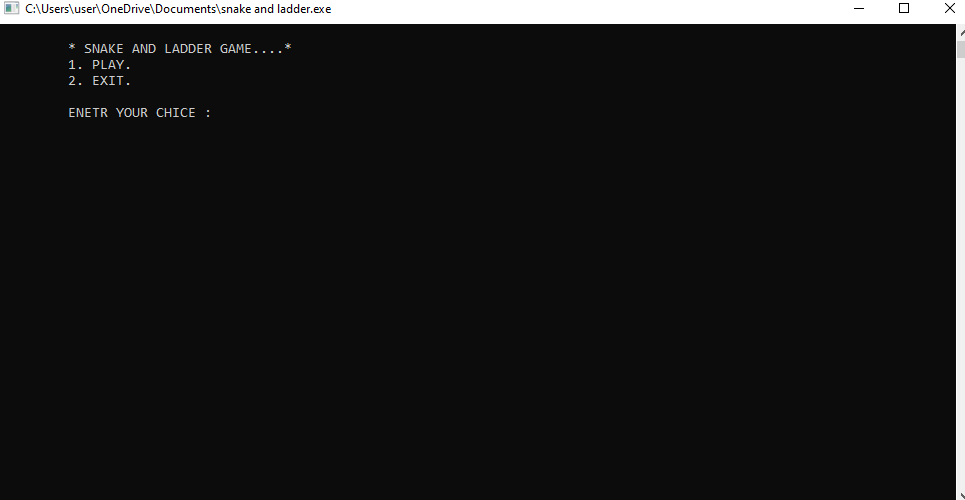
d) Till you reach 100 there can be ups and downs.

e)You can even get a snake bite at any cell and reach start point.

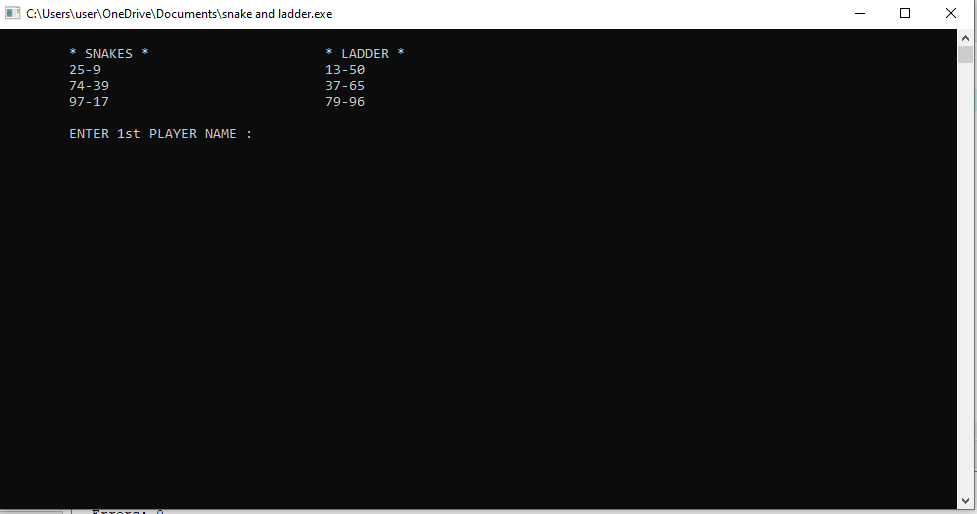
1. **Here, the game starts.**



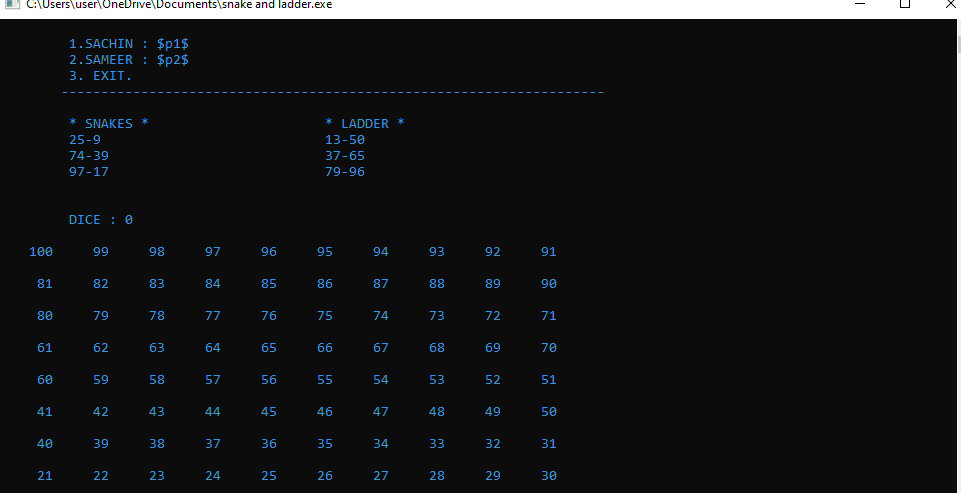
1. **We get two options.**



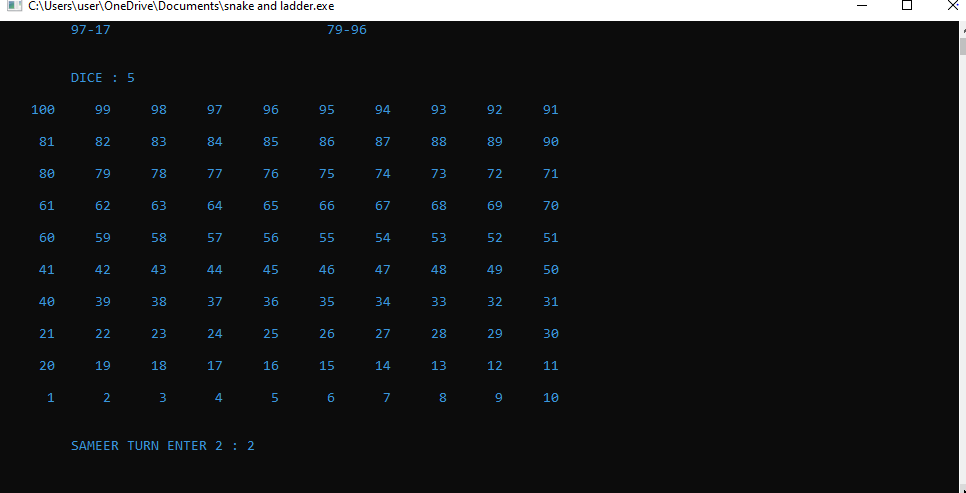
1. **After choosing first choice**



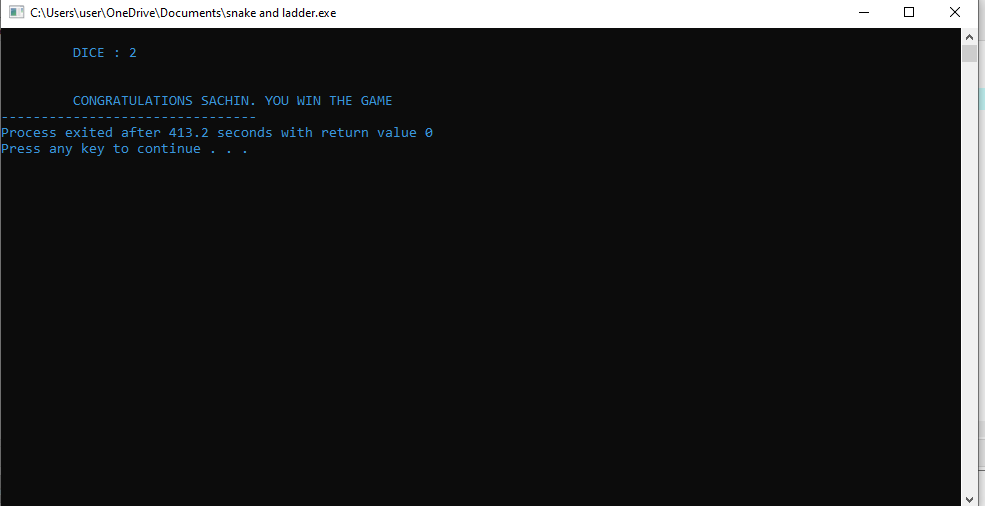
1. **The two players get their position and the game begins.**



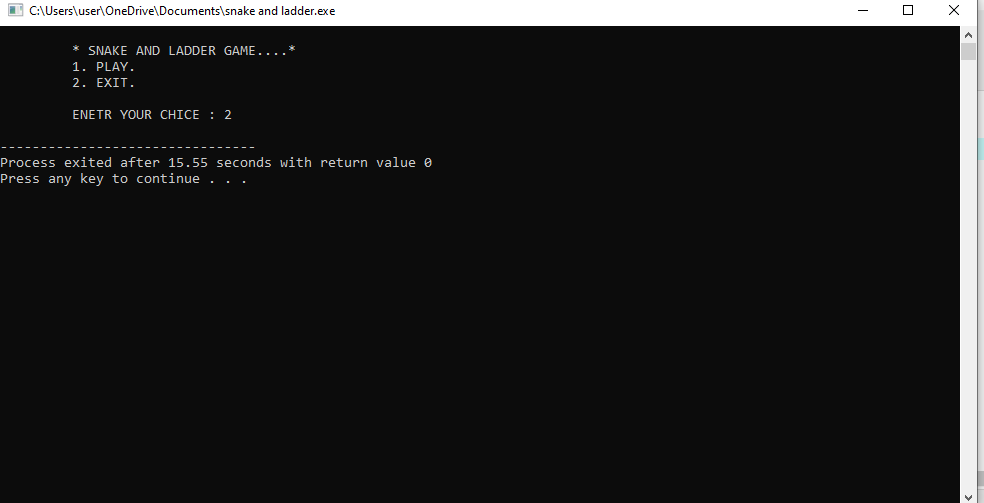
1. **The positions will go on changing as the dice is thrown.**



1. **At the end of the game one of the player reaches the end and wins the game.**



1. **For second choice we get this output.**



**CONCLUSION**

The game taught an important lesson; good deeds take you up, bad deeds take you down. At the bottom of the ladders good deeds like “hard work” is written which would lead up to “success” and bad deeds like “disobedience” would be written at the mouth of the snakes which could lead down to “bad luck”. This game of snakes and ladders is quite effective for the social and emotional development of children because the snakes and ladders contain the value of unity and unity, played in groups, and related to relationships between communities.

**FUTURE SCOPE**

As the technology changes or new requirements are expected by the user, to enhance the functionality of the product may require new versions to be introduced. Also a few changes can be done to the appearance of the board and we can upgrade for multiplayer game. We can modify the board and give the proper effect to the board.

**REFERENCE**

* [snake and ladder game · GitHub Topics · GitHub](https://github.com/topics/bank-management-system)
* [Menu-Driven Program for snake and ladder game - GeeksforGeeks](https://www.geeksforgeeks.org/menu-driven-program-for-bank-management-system/)