**NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA**

**SURATHKAL**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**UNIX PROJECT**

****

**“Snakes and Ladders game “**

**Submitted to**

**IT 202 (UNIX)**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**Nov. 15, 2018**

**By**

Venkatesh B H Niteesh kumar

17IT248 17IT228

[venkubh@gmail.com](mailto:venkubh@gmail.com) niteesh0112.nk@gmail.com

Rashad Ahmed Vishal Nirne

17IT132 17IT250

[rashadahmed163@gmail.com](mailto:rashadahmed163@gmail.com) [vishalnirne98@gmail.com](mailto:vishalnirne98@gmail.com)

**Acknowledgement**

ENTRANCE…Hard work…gradual progress and second year. That’s how we have reached this level and now we stand at the two years which we have spent in this college.

Training is agglomeration of the theoretical and practical and technical concepts, which enhances our skills in the field of technology.

We sincerely acknowledge our thanks to the teachers for their guidance and motivation throughout the training and project work.

We would also like to record our gratitude to Miss Sangeetha SH for giving us a chance for learning here.

Last, but not the least, we would like to thank all our companions for their help which was in abundance

**Abstract**

This project aims to bring the fun and simplicity of snake game with some new features. It will include computer controlled intelligent opponents whose aim will be to challenge the human players.

This project explores a new dimension in the traditional snake game to make it more interesting and challenging. The simplicity of this game makes it an ideal candidate for a minor project as we can focus on advanced topics like multiplayer functionality and implementation of computer controlled intelligent opponents.

**Table of Contents Page No.**

**Acknowledgement ………………………………………….xi**

**Abstract ………………………………………………………x**

1. **Objectives ………………………………………………1**
   1. **Problem ………………………………………..2**
   2. **Fig 1 ……………………………………………3**
2. **Domain Methods ……………………………………...4**

**2.1 startgame() ……………………………………5**

**2.2 Singleplayer() …………………………………6**

**2.3 Multiplayer() ………………………………….7**

**2.4 Snake() …………………………………………8**

**2.5 Ladder() ………………………………………..9**

**2.6 CalculateValue() ……………………………...10**

**2.7 isWin() ………………………………………….11**

1. **Position Reference …………………………………….12**

**3.1 Ladder ………………………………………….13**

**3.2 Snake …………………………………………...14**

1. **References ………………………………………………15**

**Objectives**

This game aims to change the way people think of

Traditional snake game. It will offer the

experience of commercial multilayer games to the player retaining the simplicity of traditional snake game.

The major objectives of this project are: -

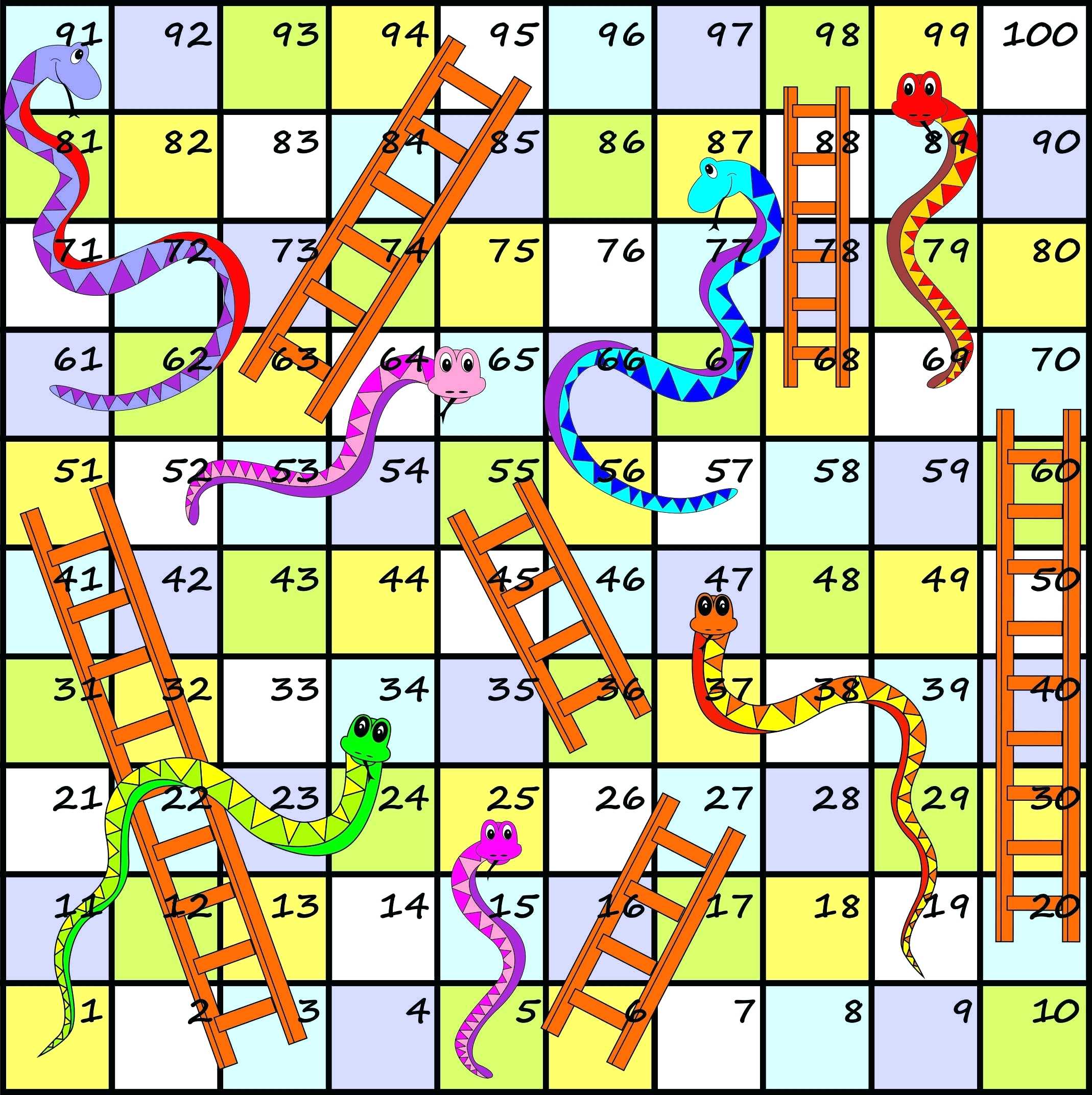
* Create a snake game that will have all the functionality of traditional snake games.
* Introduce multilayer functionality in the game that will allow two players to play a game simultaneously. It should be able to give the experience of a real time multiplayer game to the players.
* Introduce computer controlled intelligent opponent (unique feature of this game) to make the game more challenging and interesting. The movement and action of these intelligent opponents will be controlled by computer whose aim will be to eat the food before human players capture it.

**1) Problem:-**

Everyone should know the snakes and ladders game. The final goal is to build a computer game such that human players and maybe also the computer can play the game. At first, however, well just build a simulator, to make a good design and test its implementation. Anyway, a simulator may be useful on its own, for instance to calculate the probability (frequency) that a player reaches the last square after at most N turns on a board with M = 100,

This simulator accepts the following input:

* number, name and order of the players.
* ladders and snakes, specified as initial position and length and prints to console all the details of the game: at each turn,
* the state of the game (number where each player is in).
* name of the current player and the rolled dice number.
* the new position of the player, and if he/she falls into a ladder or snake, it tells so and the from-to numbers.



**Fig 1**- An image of a standard Snakes and Ladders game board

**2) Domain Methods and their responsibility:-**

The methods are simple:

Here Win point=100.

**1. startgame():**

In this method the choice will be given to the player, whether to play single player game or multiplayer game and even to read the rules before starting the game.

**2. Singleplayer():**

In this method the player has to enter his/her name before starting the game. When game begins the position of the player and the bot’s position will be shown and then the player has to press ‘r’ to roll the dice or even player can exit the game in between. When the dice is rolled the player and bot position will be shown again until one of them wins the game.

**3. Multiplayer():**

This method is same as the Singleplayer() method. Here instead of one player, two players can play the game. Before starting the game player1 and player2 has to enter their name and then game begins. To roll the dice press ‘r’ or even they can exit the game anytime. When snake bits the player moves down the board or when the player come across the ladder, the player can climb up the board. The position of the players will be shown every time until one of them wins the game.

**4. Snake():**

In this method when player come across the snake, then the player position will automatically move down as he/she is swallowed by the snake.

**5. Ladder():**

In this method when player come across the ladder, then the player position will automatically move up, as he/she will climb up the ladder.

**6. CalculateValue()**:

When the player rolls the dice by pressing ‘r’ after that this method is called to add the players position i.e., (previous position + current dice value). If this value results greater than 100 (i.e., Win point=100) then the dice value will not be added to players previous position as he/she is moving beyond the maximum value.

**7. isWin()**:

In this method it is checked that whether the player reached to the win point, if so then this player is declared has the winner of the game and the game ends.

The position where the ladder and snake occurs:

**1. Ladder:**

Position **6**: climb up to position **25**

Position **11**: climb up to position **40**

Position **17**: climb up to position **69**

Position **46**: climb up to position **90**

Position **60**: climb up to position **85**

**2. Snake:**

Position **25**: move down to position **2**

Position **52**: move down to position **42**

Position **70**: move down to position **55**

Position **95**: move down to position **72**

Position **99**: move down to position **54**

**References: -**

1. Sumitabha Das, Unix Concepts and Applications
2. Coloring command : [**http://linuxcommand.org/lc3\_adv\_tput.php**](http://linuxcommand.org/lc3_adv_tput.php)
3. Concept of Random function : [**https://blog.eduonix.com/shell-scripting/generating-random-numbers-in-linux-shell-scripting/**](https://blog.eduonix.com/shell-scripting/generating-random-numbers-in-linux-shell-scripting/)
4. How to use Functions in shell script : [**https://www.shellscript.sh/functions.html**](https://www.shellscript.sh/functions.html)
5. Rules of Snakes and Ladders Game : [**http://learnenglishkids.britishcouncil.org/en/crafts/snakes-and-ladders**](http://learnenglishkids.britishcouncil.org/en/crafts/snakes-and-ladders)