**National university of computer & emerging science**

**Karachi campus**

****

**Project name**: Snakes and ladder game

**Project proposal**

**Programming Fundamental**

**Section:** 1-B

**Project members:**

Mohid Raheel Khan (23k-3000)

Aaqib Shivji (23k-0625)

Hassan Nafees (23k-0769)

**Brief introduction:**

We are developing a game called Snakes and Ladders which is usually played on a board with a piece assigned to a player. The board contains scores from 1-100 and a player wins when he reaches the score 100. There are Snakes on scores that reduces the total score and there are Ladders on scores that increases the total score.

**Existing system:**

The existing board game Snakes and Ladders is a multiplayer game which follows the rules above. A player rolls a dice and moves his piece according to a random number that appears on the dice from 1-6. The game plays out in alternative turns between players

**Problem statement:**

We are turning the board game into a virtual game and found that the game cannot be played by a single player. If a single player does play on the existing board, there is no other way of the game ending other than the player winning it by reaching the score 100.

**Solution statement:**

We are making the game suitable for a single player by introducing a limited tries feature. If the player is struck with a Snake, he would not only reduce his total score but also would deplete his total lives of 3 tries. If the players life count turns to 0, the game will be ended, stating that the player has lost. Then he can restart the game for another round. He can play with himself, comparing his previously played rounds and see how less moves he took to complete the game through the leaderboard feature.

We solved our problem of snakes and ladders on the board by introducing structures and functions:

For Example:

addSnakeOrLadder(gameboard, 72, 91, 1);

addSnakeOrLadder(gameboard, 62, 19, 2);

Type 1 being Ladders and Type 2 being Snakes.

And we solved our problem of random numbers from the dice (1-6) by using standard library (rand () function) and our numbers should remain in between 1 to 6 we use this logic

(Rand%6) +1.

**Silent features:**

* Game board.
* 1 player.
* Dice rolling.
* Snakes.
* Ladders.
* Win condition (to complete game by reaching 100 with 1 or more lives left).
* Game loop (user will continuously play until he wins, loses, or decides to quit).
* The Leaderboards entry after winning the game which is done using Filing.

**Tools and technologies:**

1. Programming language “C”
2. 4 libraries:
3. Stdlib.h

2. Stdio.h

3. Time.h

4. Strings.h

1. Operating system “Windows”.