**GUESS MAGIC NUMBER GAME USER GUIDE**

**Attention**

Please read this manual thoroughly before use my product

* DO NOT input double number or character when you are guessing because it will break the program.
* Follow the feedback of program when you are playing.

**Introduction**

**The game’s Objective**: Generates a random number with 4 digits. The player has to guess what the number is. After the player guess the correct number, it will calculate the lucky ratio of player and show five players had highest lucky ratio in history.

**Technical Details**: I chose C as the programming language because it is very fast and versatile. Furthermore, it has some useful library I can use to implement my program. The game will run base on Visual Studio Code Software.

**Functionality Overview**: These are some function that I use to implement the game.

* int random(minN, maxN): the function has two argument. Value return is a random number with it scope [minN, maxN].
* void insertBeginList(head, data): the function insert new player into the head of linked list.
* void insertAfter(node\_player, data): the function insert new player after a node player.
* void insertSorted(head, data): the function insert new player according to the decrease order of lucky ratio.
* void printList(head): the function displays five highest players in history.
* void deallocate(head): the function free the memory.
* int \* arrayNumber(number): the function convert argument number to an array with 4 elements is 4 digits.

**Instruction**

**Step 1**: The player starts the game by double left mouse click into application “Mock\_Le\_Minh\_Huy”.

**Step 2**: The program automatic generates a random magic number with 4 digits.

**Step 3**: The program will request the player enter player’s name. They can start guess the number after they enter a valid name.

**Step 4**: The program compares guess number to magic number. And show the result on the screen. If there are matched number, the program prints out the result and the magic number (which guested number in correct position, the remaining are ‘-‘). Example: - 4 - 7. Loop until the player guess a number that all digits is in correct position.

**Step 5**: After all digits is in correct position. The program calculates lucky ratio of player. Save data of player in structural form and then write this data to file List\_player.bin.

**Step 6**: The program gets all data in file List\_player.bin and insert into a linked list according to decrease order of lucky ratio. Loop until all data in file List\_player.bin has been loaded.

**Step 7**: After loop, the program displays five highest player in history.

**Step 8**: The program asks player: “Do you want to replay?(y/n)”

**Step 9**: If the player type ‘y’. The program will start at Step 2 again. If the player type ‘n’, they will exit the game.