Project 7B Ch7 Write-Up

**Meena Chandok**

**Pseudo-code**

1. Declare variables: ARRAY\_SIZE and MAX\_RANGE declared as const int. ARRAY\_SIZE initialized to size 5 and MAX\_RANGE to 9. Array variable int User[ARRAY\_SIZE] to hold user’s input and array variable int LotteryNum to hold randomly generated lottery numbers. LotteryNum were initialized to 0.
2. Defined 3 function prototypes: int generateNumbers (to generate random 5 digit lottery numbers, and to return them one by one to main); void findMatches (to compare lottery and user’s array and find out if they are equal or not, and to keep track of how many numbers matched); void displayValue (to display the lottery and user’s numbers)
3. Cout statement to display the purpose of the program.
4. Called the number generation function in main 5 times with a for loop to get lottery numbers and take user’s input. Each random number was generated within the function and sent back to main, to be stored in an array sequentially. User input was solicited within the same for loop, and stored in a separate array.
5. Function to generate lottery number uses minimum value =0 and maximum value =9 as const int.

Unsigned seed =time(0)+i was used to generate a different number through each iteration of the FOR loop. Used srand(seed) to seed the random number through

LotteryNum = (rand() % (MAX\_VAL - MIN\_VAL + 1)) + MIN\_VAL;

The random number generated lottery numbers were returned to main as values of array variable LotteryNum. Alternatively, the function could have been used with a for loop to generate all 5 numbers at the same time.

1. FOR loop which generated lottery numbers used to get users input. With each iteration of loop the users input was stored in array variable User[i]. While loop performs the input validation to of user’s entry to negative and numbers greater than 9. If user enters any negative number or numbers greater than 9 than the message of please enter a positive number between range of 0-9 is displayed until user enters the correct information.
2. Display value function called in main to display the Lottery and User’s array values.
3. Find match function called to compare the lottery and user’s arrays using bool arraysEqual. If all 5 digits are matched during the while loop, arraysEqual is true and the user is declared as winner. Otherwise the comparison declares arrays as unequal and finds the number of matches and displays them, asking user to try again.

**Test Cases:**

**Valid Data** for random lottery number generation:

Positive integer between 0 and 9

**Valid Data** for user input:

Positive integer between 0 and 9

**Test Case Artifacts:**

|  |  |  |
| --- | --- | --- |
| **Possible Input for Number** | **Expected Output** | **Actual Output** |
| **User number input** |  |  |
| Negative numbers  -1,-43,-67888 etc. | "Please enter a positive number between range of 0-9" | "Please enter a positive number between range of 0-9" |
| Positive numbers >9  e.g. 33, 67, 643, 98765 etc. | "Please enter a positive number between range of 0-9" | "Please enter a positive number between range of 0-9" |
| Positive number in range of 0-9.  e.g. 4 (in this case when tested) | Accepts number and proceeds to next step | Accepts number and proceeds to next step |
|  |  |  |
| **Comparison of Lottery and User Array** |  |  |
| User input numbers: 2 4 5 7 9  Lottery Numbers generated:  0 3 3 3 9 | "User array: "  2 4 5 7 9  "Lottery array: "  0 3 3 3 9  “The arrays are not equal. Only 1 number match”  “Sorry try again” | "User array: "  2 4 5 7 9  "Lottery array: "  0 3 3 3 9  “The arrays are not equal. Only 1 number match”  “Sorry try again” |
| User input numbers: 4 1 0 7 7  Lottery Numbers generated:  4 1 0 7 7 | "User array: "  4 1 0 7 7  "Lottery array: "  4 1 0 7 7  “The arrays are equal. All 5 numbers match”  "Congratulations, you are a grand prize winner." | "User array: "  4 1 0 7 7  "Lottery array: "  4 1 0 7 7  “The arrays are equal. All 5 numbers match”  "Congratulations, you are a grand prize winner." |
| User input numbers: 5 1 1 0 4  Lottery Numbers generated:  5 3 1 3 5 | "User array: "  5 1 1 0 4  "Lottery array: "  5 3 1 3 5  “The arrays are not equal. Only 2 number match”  “Sorry try again” | "User array: "  5 1 1 0 4  "Lottery array: "  5 3 1 3 5  “The arrays are not equal. Only 2 number match”  “Sorry try again” |
| User input numbers: 4 7 0 0 6  Lottery Numbers generated:  4 5 0 0 6 | "User array: "  4 7 0 0 6  "Lottery array: "  4 5 0 0 6  “The arrays are not equal. Only 4 number match”  “Sorry try again” | "User array: "  4 7 0 0 6  "Lottery array: "  4 5 0 0 6  “The arrays are not equal. Only 4 number match”  “Sorry try again” |
| User input numbers: 1 0 0 6 6  Lottery Numbers generated:  2 3 0 6 6 | "User array: "  1 0 0 6 6  "Lottery array: "  2 3 0 6 6  “The arrays are not equal. Only 3 number match”  “Sorry try again” | "User array: "  1 0 0 6 6  "Lottery array: "  2 3 0 6 6  “The arrays are not equal. Only 4 number match”  “Sorry try again” |
|  |  |  |
|  |  |  |

Assumptions:

I am assuming that the user will follow instructions for guessing the random number.

Learning Experience:

I understood the how to compare two arrays and its use. I got an appreciation of the two arrays with use of loops and functions. It would be possible to put the whole process within a loop, to enable the user to try to input the 5 values repeatedly. It would be more challenging to keep track of which numbers matched between the two arrays and to display the matches.