

Presented by: **Maryam Siddiqui**

**www.linkedin.com/in/maryam-siddiqui-ghani**

Task: **Number-Guessing Game**

**Project Report:**

**Number-Guessing Game:**

**Abstract:**

In this project, we implemented a Number Guessing Game using C++. The program generates a random number between 1 and 100, and the user has a maximum of 10 attempts to guess the correct number. This report outlines the objectives, implementation, testing, and outcomes of the project.

**Introduction:**

The "Number Guessing Game" project is a console-based game developed in C++ as an internship task for CodeSoft. The purpose of this project is to create an interactive and engaging number guessing game that challenges players to guess a secret number within a limited number of attempts. The game is designed to provide users with an enjoyable and challenging experience while practicing their guessing skills.

**Project Scope:**

The scope of this project involves implementing a console-based number guessing game where the player attempts to guess a randomly generated secret number between 1 and 100. The game provides the player with feedback on their guesses, offers a limited number of attempts, and allows players to play multiple rounds if desired.

**Features:**

* Visually appealing welcome message and game title displayed using ASCII art.
* Randomly generated secret number for each game.
* Introduction and rules displayed to guide players.
* User input for guessing the secret number.
* Feedback provided for each guess (too high, too low, or correct).
* A maximum of 10 attempts for each game.
* Option for players to play multiple rounds.

**Implementation:**

The "Number Guessing Game" is implemented using C++ programming language and structured into several functions for improved modularity and readability, utilizes the iostream, cstdlib, and ctime libraries. The core logic includes:

* Generating a random number using rand() and srand(time(0))
* Displaying user prompts and feedback
* Implementing a loop for multiple game sessions

**Functions:**

**displayWelcome():**

Displays the welcome message and visually styled game title using ASCII art.

**generateRandomNumber():**

Generates a random number between 1 and 100 using the rand() function and current time as a seed.

**displayIntro():**

Displays introductory messages and rules of the game.

**getUserGuess():**

Gets the player's guess as input.

**provideFeedback(int guess, int num, int &tries):**

Compares the player's guess with the secret number and provides feedback based on the guess.

Tracks the number of attempts made by the player.

**playAgain():**

Asks the player if they want to play another round and returns a boolean value accordingly.

**Main Function:**

The main function orchestrates the gameplay, calling the functions in a logical sequence to provide an engaging experience to the player. The game loop allows players to play multiple rounds.

**6. Testing:**

To ensure the program's functionality and robustness, we performed various testing scenarios, including:

* Guessing the correct number within the allowed attempts
* Exceeding the maximum number of attempts without guessing correctly
* Restarting the game and checking for smooth transition

**7. Results:**

Upon executing the program, the user is welcomed with a title and instructions for the game. Screenshots of the program's initial output can be included here. As the user guesses, the program provides feedback and ends the game either when the correct number is guessed or when the maximum attempts are reached. Screenshots illustrating various game scenarios can be included in this section.

**Conclusion:**

The "Number Guessing Game" project displays the implementation of an interactive and entertaining console-based game. By dividing the code into functions and providing clear feedback to the player, the project demonstrates effective use of programming concepts to create an enjoyable user experience. By providing an interactive experience, the program engages users and reinforces their understanding of random number generation, conditional statements, loops, and user input handling. The testing phase ensured the program's reliability and user-friendly nature, making it an enjoyable and educational project.

**Acknowledgments:**

Maryam Siddiqui as an internship task completed this project for CodeSoft. Special appreciation goes to mentors and guides for their support during the development process.

**Note:** This project report template serves as a foundation and can be further customized to match your specific project's details and requirements.

Feel free to expand on this project report template according to your needs.

**Source code:**

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

int main() {

// Welcome message and game title

cout << " ################### Welcome To Code Guessing Game ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

int num, guess, tries = 0;

const int maxAttempts = 10; // Set a maximum number of attempts

char playAgain;

srand(time(0)); // Seed random number generator

do {

num = rand() % 100 + 1; // Generate a random number between 1 and 100

tries = 0;

cout <<endl<<endl<< " Let`s start the Game";

cout <<endl<<endl<< " NOTE!!!!!!!!!!!!!!!!";

cout <<endl<<endl<< " You Have only 10 chance to Guess";

cout <<endl<<endl<< " Guess My Number Game\n\n";

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << " ################### ##################### " << endl;

cout << endl << endl;

do {

// Input prompt for the user to guess the number

cout << " Enter a guess between 1 and 100: " << endl;

cout << " ";

cin >> guess;

tries++;

// Provide feedback based on the user's guess

if (guess > num) {

cout << " -------------" << endl;

cout << " | Too high! |" << endl;

cout << " -------------" << endl<<endl;

} else if (guess < num) {

cout << " -------------" << endl;

cout << " | Too low! |" << endl;

cout << " -------------" << endl<<endl;

}

// User guessed the correct number

else {

cout << endl;

cout << " -----------------------------------" << endl;

cout << " | Correct! You got it in " << tries << " guesses! |" << endl;

cout << " -----------------------------------" << endl<<endl;

}

// Check if the user has reached the maximum number of attempts

if (tries == maxAttempts) {

if (tries == maxAttempts) {

cout << " Out of attempts! The secret number was: " << num << endl;

break;

}

}} while (guess != num);

cout<<endl;

// Ask if the user wants to play again

cout << " Do you want to play again? (y/n): ";

cin >> playAgain;

cout << endl;

} while (playAgain == 'y' || playAgain == 'Y');

// Farewell message

cout<<" ------------------------------------------------"<<endl;

cout << " | Thanks for playing! Have a great day! Goodbye! |" << endl;

cout<<" ------------------------------------------------"<<endl;

return 0;

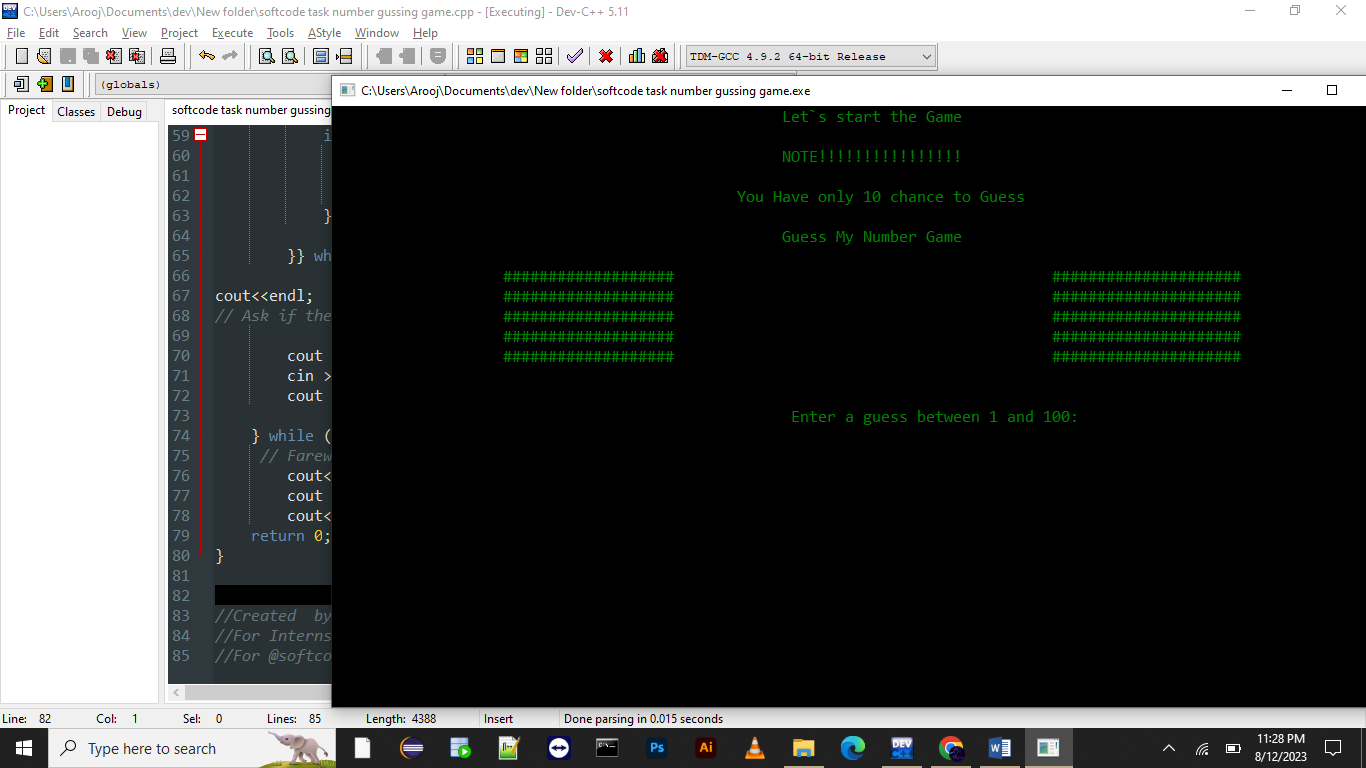
}

**//Created by Maryam Siddiqui**

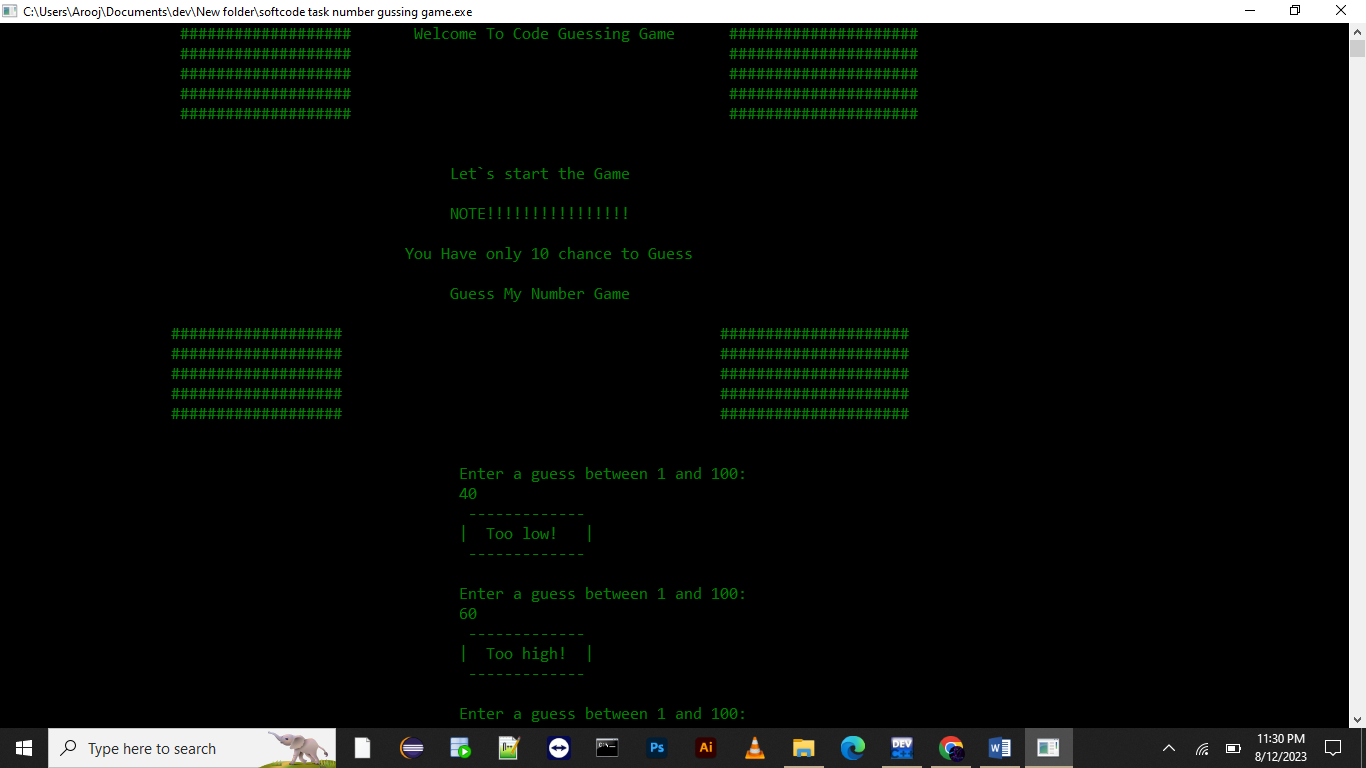
**//For Internship Task Number Guessing game**

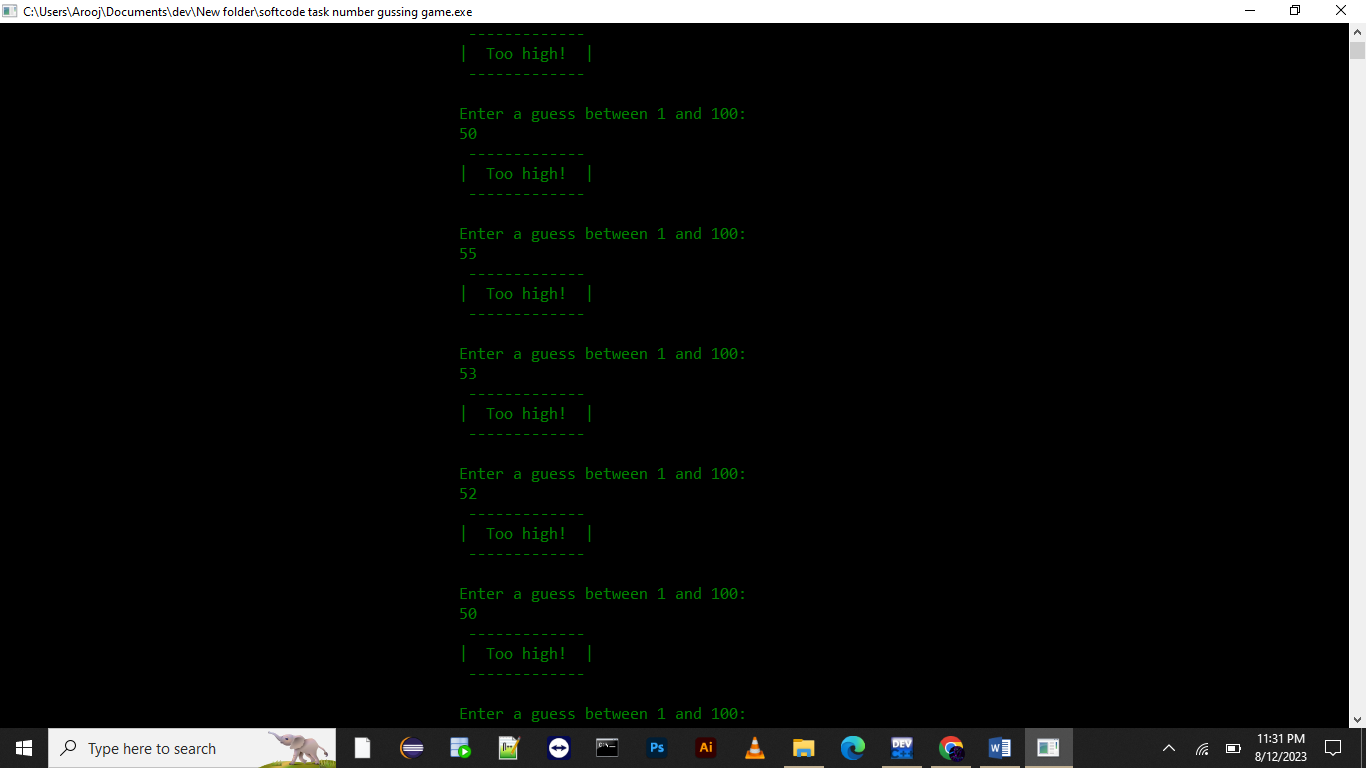
**//For @softcode**

**Output:**

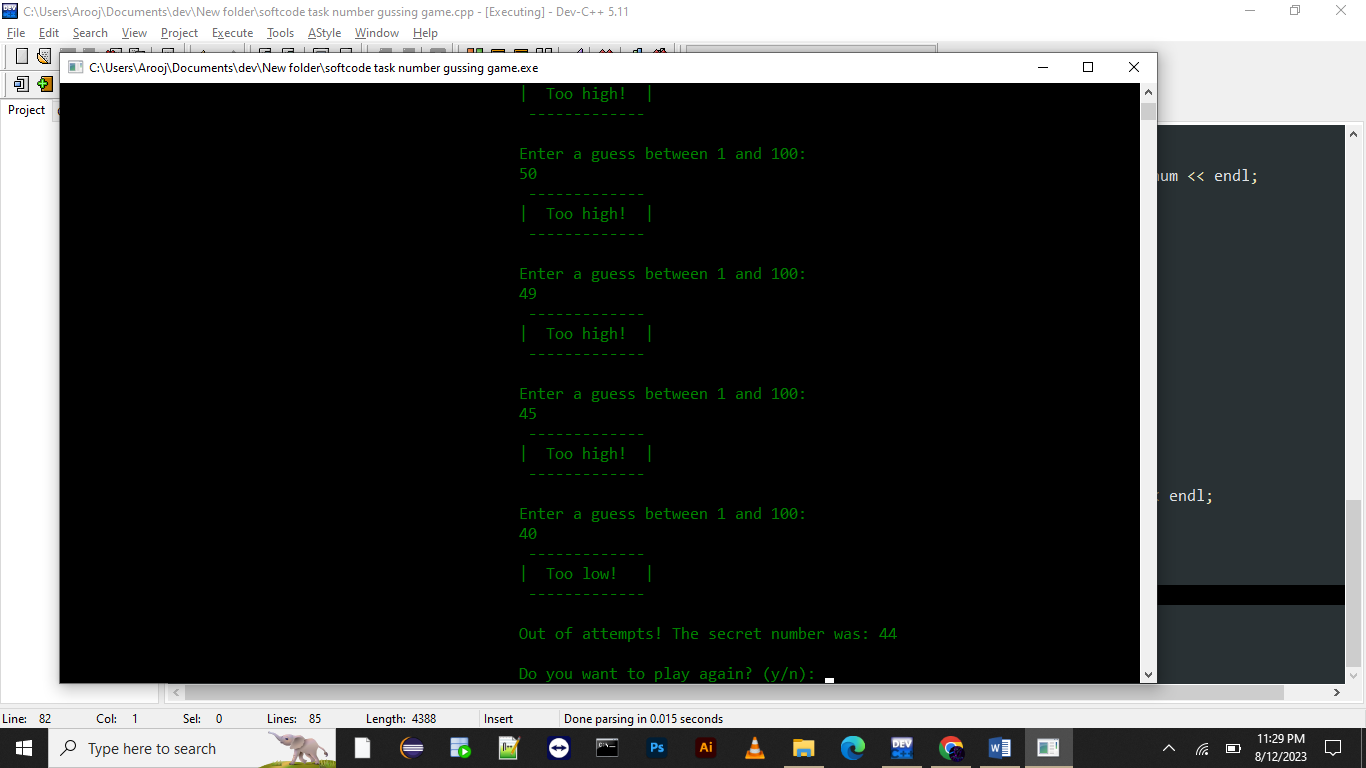


Opening messages or console:





After 10 attempts you failed and it revealed the secrete number



Farewell messages:

