#Task 01 – No. guessing game

#include <iostream>

#include <ctime>

#include <cstdlib>

using namespace std;

int main() {

// Seed the random number generator with the current time

srand(static\_cast<unsigned int>(time(0)));

// Generate a random number between 1 and 100

int random\_number = rand() % 100 + 1;

// Variable to store the user's guess

int user\_guess;

cout << "Guess the number between 1 and 100!" << endl;

// Loop until the user guesses the correct number

while (true) {

cout << "Enter your guess: ";

cin >> user\_guess;

if (user\_guess == random\_number) {

cout << "Congratulations! You guessed the correct number." << endl;

break;

} else if (user\_guess < random\_number) {

cout << "Your guess is too low. Try again." << endl;

} else {

cout << "Your guess is too high. Try again." << endl;

}

}

return 0;

}

Task 03 tic tac toe

#include <iostream>

#include <ctime>

#include <cstdlib>

using namespace std;

// Function to display the board

void displayBoard(char board[3][3]) {

cout << endl;

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

cout << board[i][j] << " ";

}

cout << endl;

}

}

// Function to check if the current player has won

bool checkWin(char board[3][3], char player) {

for (int i = 0; i < 3; i++) {

if (board[i][0] == player && board[i][1] == player && board[i][2] == player) {

return true;

}

}

for (int i = 0; i < 3; i++) {

if (board[0][i] == player && board[1][i] == player && board[2][i] == player) {

return true;

}

}

if (board[0][0] == player && board[1][1] == player && board[2][2] == player) {

return true;

}

if (board[0][2] == player && board[1][1] == player && board[2][0] == player) {

return true;

}

return false;

}

// Function to check if the game is a draw

bool checkDraw(char board[3][3]) {

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

if (board[i][j] == ' ') {

return false;

}

}

}

return true;

}

// Function to play the game

void playGame() {

// Seed the random number generator with the current time

srand(static\_cast<unsigned int>(time(0)));

// Create a 3x3 grid as the game board

char board[3][3] = {{' ', ' ', ' '},

{' ', ' ', ' '},

{' ', ' ', ' '}};