Codsoft

C++ programming

Task1

```
#include <iostream>
using namespace std;

int main() {

    int  random_number= rand() % 100+1;
    int user_guess = 0;

    cout << "Guess the number between 1
and 100: ";

    while (true) {

        cin >> user_guess;
```

```
        if (user_guess < random_number)
        {
            cout << "Too low! Try again: ";
        }
        else if (user_guess > random_number)
        {
            cout << "Too high! Try again: ";
        }
        else
        {
            cout << "Congratulations! You've
guessed the number." << endl;
            cout<<"\n Random number=
"<<random_number;
            cout<<"\t,user_guess number=
"<<user_guess;

            break;
        }
    }
```

```
    return 0;
}
```

Task2

```
#include<iostream>
using namespace std;
int main()
{
    int a,b,c,result;
    float f;
    cout<<"Enter a and b value:";
    cin>>a>>b;
    do
    {

cout<<"\n1.add\t2.sub\t3.mul\t4.div\t5.exit
";
        cout<<"\nEnter your choice";
        cin>>c;
```

```
        switch(c)
        {
            case 1:
                result=a+b;
                cout<<"\n The Addition of "<<a
<<"&"<< b<<" number is: "<<result;
                break;
            case 2:
                result=a-b;
                cout<<"\n The Subraction of "<<a
<<"&"<< b<<" number is: "<<result;
                    break;
            case 3:
                result=a*b;
                cout<<"\n The Multiplication of
"<<a <<"&"<< b<<" number is: "<<result;
                    break;
            case 4:
                f=a/b;
                cout<<"\n The Division of "<<a
<<"&"<< b<<" number is: "<<f;
```

```
                break;

            case 5:
                break;



            default:
                cout<<"\nInvalid choice,Try
again.";
                break;
        }
    }while(c!=5);
    cout<<"\n...Exit...";
    return 0;

}
```

Task3

```
#include <iostream>
#include <vector>
```

```
using namespace std;

class TicTacToe {
private:
    vector<vector<char>> board;
    char currentPlayer;

public:
    TicTacToe() {
        board = vector<vector<char>>(3,
vector<char>(3, ' '));
        currentPlayer = 'X'; // X starts the game
    }

    void displayBoard() {
        cout << "Current Board:" << endl;
        for (int i = 0; i < 3; ++i) {
            for (int j = 0; j < 3; ++j) {
                cout << board[i][j];
                if (j < 2) cout << " | ";
```

```
            }
            cout << endl;
            if (i < 2) cout << "---------\n";
        }
    }

    bool placeMark(int row, int col) {
        if (row < 0 || row >= 3 || col < 0 || col >=
3 || board[row][col] != ' ') {
            cout << "Invalid move. Try again." <<
endl;
            return false;
        }
        board[row][col] = currentPlayer;
        return true;
    }

    bool checkWin() {
        // Check rows, columns, and diagonals
        for (int i = 0; i < 3; ++i) {
            if ((board[i][0] == currentPlayer &&
```

```
board[i][1] == currentPlayer && board[i][2]
== currentPlayer) ||
            (board[0][i] == currentPlayer &&
board[1][i] == currentPlayer && board[2][i]
== currentPlayer)) {
            return true;
        }
    }
    if ((board[0][0] == currentPlayer &&
board[1][1] == currentPlayer && board[2][2]
== currentPlayer) ||
        (board[0][2] == currentPlayer &&
board[1][1] == currentPlayer && board[2][0]
== currentPlayer)) {
        return true;
    }
    return false;
}

bool isBoardFull() {
    for (const auto& row : board) {
```

```
            for (char cell : row) {
                if (cell == ' ') return false;
            }
        }
        return true;
    }

    void switchPlayer() {
        currentPlayer = (currentPlayer == 'X') ?
'O' : 'X';
    }

    void play() {
        while (true) {
            displayBoard();
            int row, col;
            cout << "Player " << currentPlayer <<
", enter your move (row and column): ";
            cin >> row >> col;

            if (placeMark(row - 1, col - 1)) {
```

```
            if (checkWin()) {
                displayBoard();
                cout << "Player " <<
currentPlayer << " wins!" << endl;
                break;
            }
            if (isBoardFull()) {
                displayBoard();
                cout << "It's a draw!" << endl;
                break;
            }
            switchPlayer();
        }
    }
};

int main() {
    TicTacToe game;
    game.play();
    return 0;
```

```
}
```