Codsoft project

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,r;
    float s;
    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:
            r=a+b;
            cout<<"\n The Addition of "<<a
<<"&"<< b<<" number is: "<<r;
            break;
        case 2:
            r=a-b;
            cout<<"\n The Subraction of "<<a
<<"&"<< b<<" number is: "<<r;
                break;
        case 3:
            r=a*b;
            cout<<"\n The Multiplication of
"<<a <<"&"<< b<<" number is: "<<r;
                break;
        case 4:
            s=a/b;
            cout<<"\n The Division of "<<a
<<"&"<< b<<" number is: "<<s;
                break;
```

```
            case 5:
                break;



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

}
```

Task4:

```
// Online C++ compiler to run C++ program
online
#include <iostream>
#include <vector>
```

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

class Task {
public:
    Task(string& desc) : description(desc),
completed(false) {}

    void markCompleted()
    {
        completed = true;

    }
    string getStatus()
    {
       return completed ? "Completed" :
"Pending";
    }
    string getDescription()
    {
        return description;
```

```
    }

private:
    string description;
    bool completed;
};

class T_List {
public:
    void addTask(string& desc)
    {
        tasks.push_back(Task(desc));

    }
    void viewTasks()
    {
        if (tasks.empty())
        {
            cout << "No tasks available.\n";
            return;
```

```
        }
        for (size_t i = 0; i < tasks.size(); ++i)
        {
            cout << i + 1 << ". [" <<
tasks[i].getStatus() << "] " <<
tasks[i].getDescription() << "\n";
        }
    }
    void markTaskCompleted(size_t index)
    {
        if (index < tasks.size())
        tasks[index].markCompleted();
    }
    void removeTask(size_t index)
    {
        if (index < tasks.size())
        tasks.erase(tasks.begin() + index);
    }

private:
  vector<Task> tasks;
```

```
};

int main() {
    T_List t;
    int choice;

    do {
        cout << "\n1. Add Task\t2. View
Tasks\t3. Mark Task as Completed\t4.
Remove Task\t5. Exit\nChoose an option: ";
        cin >> choice;

        switch (choice) {
            case 1: {
                string taskDesc;
                cout << "Enter task description: ";
                cin>>taskDesc;
                t.addTask(taskDesc);
                break;
            }
            case 2:
```

```
            t.viewTasks();
            break;
        case 3: {
            t.viewTasks();
            size_t taskNum;
            cout << "Enter task number to
mark as completed: ";
            cin >> taskNum;
            t.markTaskCompleted(taskNum -
1);
            break;
        }
        case 4: {
            t.viewTasks();
            size_t taskNum;
            cout << "Enter task number to
remove: ";
            cin >> taskNum;
            t.removeTask(taskNum - 1);
            break;
        }
```

```
        }
    } while (choice != 5);
    cout<<"Exit..";
    return 0;
}
```