1. **Write a C program to simulate a Non-Deterministic Finite Automata (NFA) for the given languagerepresenting strings that start with o and end with 1**

**AIM :**

**To Write a C program to simulate a Non-Deterministic Finite Automata (NFA) for the given languagerepresenting strings that start with o and end with 1**

**CODE :**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <stdbool.h>**

**#include <string.h>**

**#define MAX\_LENGTH 100**

**// Structure to represent a transition in the NFA**

**typedef struct Transition {**

**char symbol;**

**int nextState;**

**} Transition;**

**// Structure to represent an NFA**

**typedef struct NFA {**

**int numStates;**

**int numTransitions;**

**Transition \*transitions;**

**int \*acceptingStates;**

**int initialState;**

**} NFA;**

**// Check if the given string is accepted by the NFA**

**bool acceptString(NFA \*nfa, char \*input) {**

**int currentState = nfa->initialState;**

**int inputLength = strlen(input);**

**int i, j;**

**bool foundNextState;**

**// Iterate over the input symbols and transitions in the NFA**

**for (i = 0; i < inputLength; i++) {**

**foundNextState = false;**

**for (j = 0; j < nfa->numTransitions; j++) {**

**if (nfa->transitions[j].symbol == input[i] && nfa->transitions[j].nextState == currentState) {**

**// Transition is valid, move to next state**

**currentState = nfa->transitions[j].nextState;**

**foundNextState = true;**

**break;**

**}**

**}**

**if (!foundNextState) {**

**// No valid transition found, reject string**

**return false;**

**}**

**}**

**// Check if the final state is an accepting state**

**for (i = 0; i < nfa->numStates; i++) {**

**if (nfa->acceptingStates[i] == currentState) {**

**return true;**

**}**

**}**

**return false;**

**}**

**int main() {**

**// Define the NFA**

**NFA nfa = {**

**.numStates = 3,**

**.numTransitions = 4,**

**.transitions = (Transition[]) {**

**{ '0', 0 },**

**{ 'o', 1 },**

**{ '1', 2 },**

**{ 'o', 2 }**

**},**

**.acceptingStates = (int[]) { 2 },**

**.initialState = 1**

**};**

**char input[MAX\_LENGTH];**

**// Get input from user**

**printf("Enter a string: ");**

**scanf("%s", input);**

**// Check if the string is accepted by the NFA**

**if (acceptString(&nfa, input)) {**

**printf("String accepted.\n");**

**} else {**

**printf("String rejected.\n");**

**}**

**return 0;**

**}**