**Programs**

**[2] DFA Which Accepts Strings Start With ‘0’ And End With ‘1’**

#include <stdio.h>

#include <stdbool.h>

#include <string.h>

int transition(int state, char input)

{

switch (state)

{

case 0:

if (input == '0') return 1;

else return 0;

case 1:

if (input == '0') return 1;

else if (input == '1') return 2;

else return 0;

case 2:

if (input == '1') return 2;

else return 0;

}

return 0;

}

bool simulateDFA(char \*input)

{

int currentState = 0;

int inputLength = strlen(input);

for (int i = 0; i < inputLength; i++)

currentState = transition(currentState, input[i]);

return currentState == 2;

}

int main()

{

char input[100];

printf("Enter a String : ");

scanf("%s", input);

if (simulateDFA(input))

printf("Accepted\n");

else

printf("Not Accepted\n");

return 0;

}

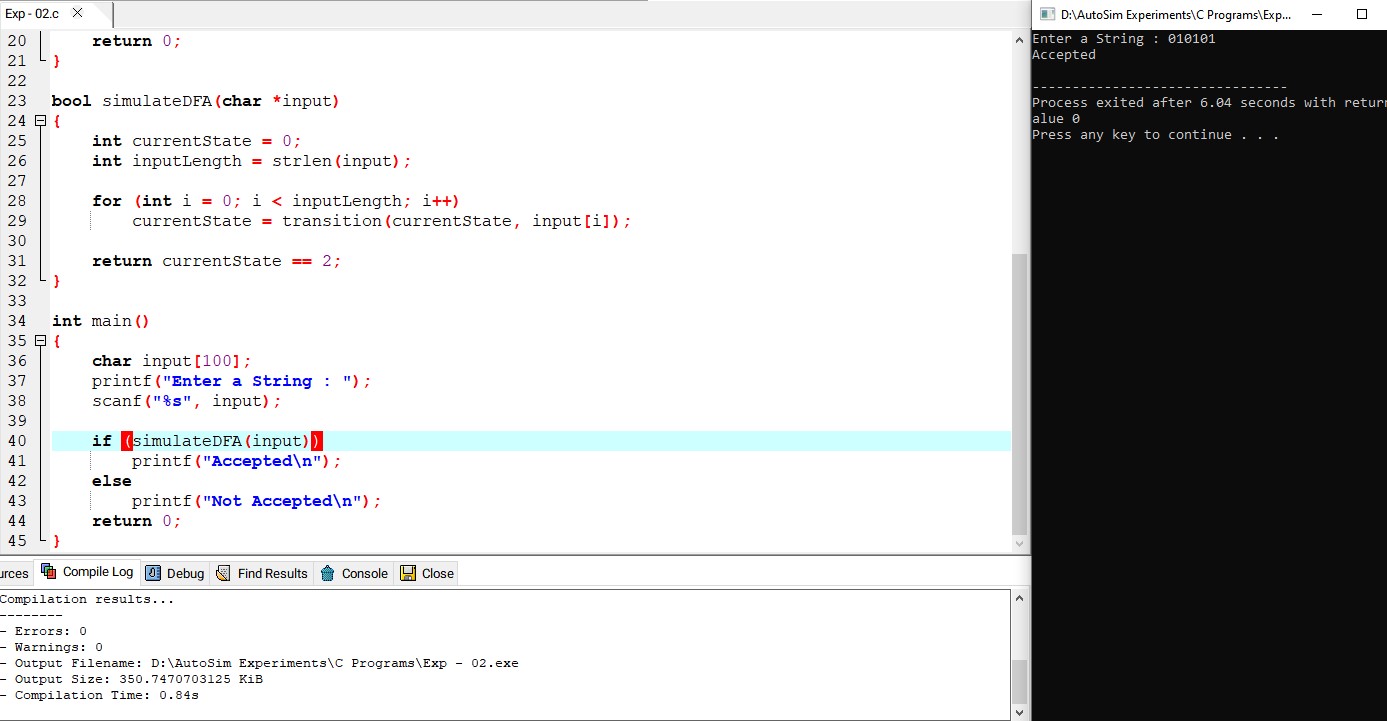
**Algorithm**

Initialize current state = q0 (start state).

For each symbol in the input string:

Apply the transition function: current state = δ (current state, symbol)

If the last symbol in the input string is 'A' and current state is q2, then the input string is accepted. Otherwise, it is rejected.



**Result**

Thus we have successfully implemented and executed the program and the strings given as inputs are verified