EXPERIMENT NO. 1

AIM :- To write a C program to simulate a DFA.

ALGORITHM:-

1. Ask the user to input the number of states
2. Ask the user to provide the number of final states
3. Get set of final states
4. Store set of final states in an array
5. Get transition function from the user
6. Store the transition function and some suitable data structure (2 dimensional array or Link list)
7. Ask the user to provide the input string
8. Calculate the length of string. Let it be k.
9. set i=1, state=initial state
10. state=transition\_function(state, ith character of the string)
11. if i!=k go to step 10
12. If state is in final state string accepted else string rejected

CODE:-

#include <stdio.h>

#define TOTAL\_STATES 2

#define FINAL\_STATES 1

#define ALPHABET\_CHARCTERS 2

#define UNKNOWN\_SYMBOL\_ERR 0

#define NOT\_REACHED\_FINAL\_STATE 1

#define REACHED\_FINAL\_STATE 2

enum DFA\_STATES{q0,q1};

enum input{a,b};

int Accepted\_states[FINAL\_STATES]={q1};

char alphabet[ALPHABET\_CHARCTERS]={'a','b'};

int Transition\_Table[TOTAL\_STATES][ALPHABET\_CHARCTERS];

int Current\_state=q0;

void DefineDFA()

{

Transition\_Table[q0][a] = q1;

Transition\_Table[q0][b] = q0;

Transition\_Table[q1][a] = q1;

Transition\_Table[q1][b] = q0;

}

int DFA(char current\_symbol)

{

int i,pos;

for(pos=0;pos<ALPHABET\_CHARCTERS; pos++)

if(current\_symbol==alphabet[pos])

break;//stops if any character other than a or b

if(pos==ALPHABET\_CHARCTERS)

return UNKNOWN\_SYMBOL\_ERR;

for(i=0;i<FINAL\_STATES;i++)

if((Current\_state=Transition\_Table[Current\_state][pos])

==Accepted\_states[i])

return REACHED\_FINAL\_STATE;

return NOT\_REACHED\_FINAL\_STATE;

}

int main(void)

{

char current\_symbol;

int result;

DefineDFA(); //Fill transition table

printf("Enter a string with 'a' s and 'b's:\n Press Enter Key to stop\n");

while((current\_symbol=getchar())!= '\n')

if((result= DFA(current\_symbol))==UNKNOWN\_SYMBOL\_ERR)

break;

switch (result) {

case UNKNOWN\_SYMBOL\_ERR:printf("Unknown Symbol %c",

current\_symbol);

break;

case NOT\_REACHED\_FINAL\_STATE:printf("Not accepted"); break;

case REACHED\_FINAL\_STATE:printf("Accepted");break;

default: printf("Unknown Error");

}

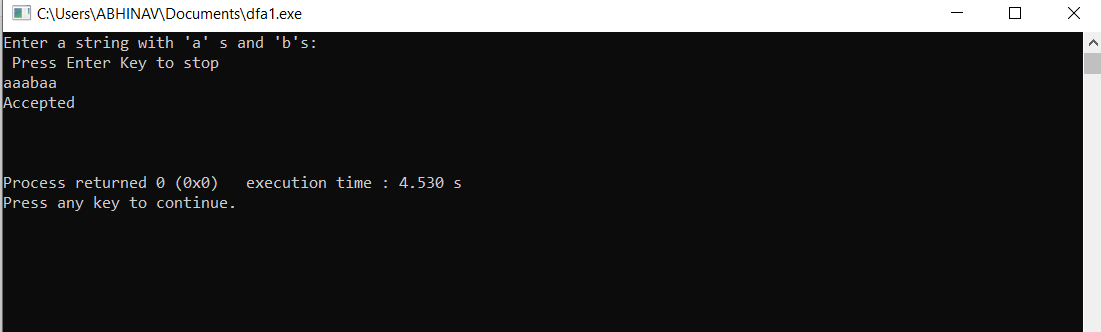
printf("\n\n\n");

return 0;

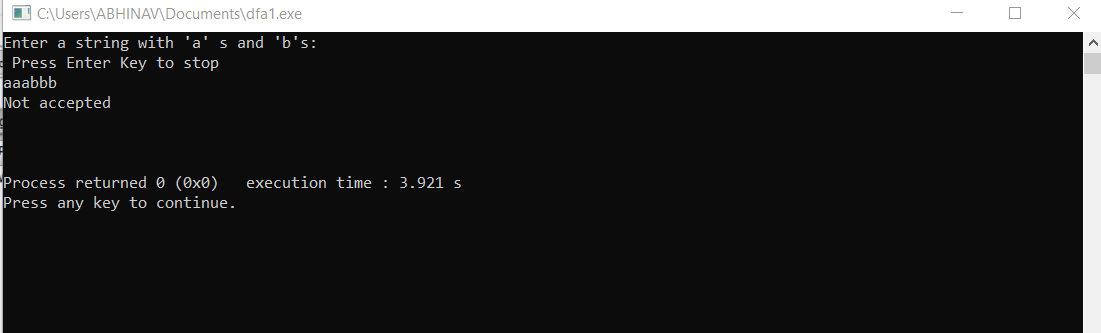
}

OUTPUT:-

CASE 1:-



CASE 2:-



RESULT:- The C Program to simulate DFA has been successfully executed.