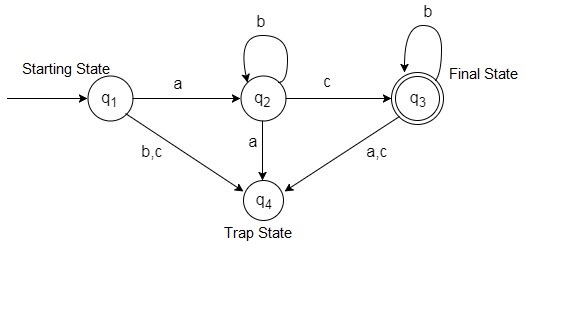
**Programming a DFA**

**DFA (Deterministic Finite Automaton or Acceptor)** is a finite state machine that accepts or rejects strings of symbols. DFA accepts the string if it reaches the final state and rejects otherwise.



**The Graph of the DFA**

Given Quintuple:

Alphabet, Σ = {a,b,c} [Input alphabet Σ]

Set of states, Q = { q1, q2, q3, q4}

Start state, S = q1

Set of final states, F = {q3}

L=Σ\*= {ac, abc, acb, abcb, abbcbb, abbbcbbb, abcbbb, abbbcb…………} [Accept words from an inﬁnite language, L]

Transitions,

δ(q1,a)= q2

δ(q1,b)= q4

δ(q1,c)= q4

δ(q2,a)= q4

δ(q2,b)= q2

δ(q2,c)= q3

δ(q3,a)= q4

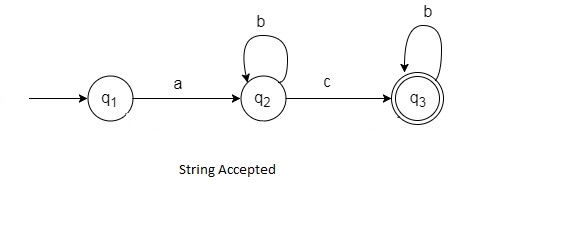
δ(q3,b)= q3

δ(q3,c)= q4

**Transition Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| **CurrentStates \ Input** | **a** | **b** | **c** |
| **q1** | q2 | q4 | q4 |
| **q2** | q4 | q2 | q3 |
| **q3** | q4 | q3 | q4 |

**String Accepted abcb**



**Implementation using C program:**

**Source Code:**

#include<stdio.h>

#define EOS '\0'

int main()

{

char m, inputstring[50]; //m is ALPHABET CHARCTERS(a,b,c)

int i, q;

printf("Enter Input String: ");

scanf("%s",inputstring);

q=1;

i=0;

m=inputstring[i];

printf("\n");

printf("%s ", inputstring);

while(m!=EOS)

{

if(q==1 && m=='a') // Transition, δ(q1,a)= q2

{

q=2;

}

else if(q==2 && m=='b') //Transition, δ(q2,b)= q2

{

q=2;

}

else if(q==2 && m=='c') //Transition, δ(q2,c)= q3

{

q=3;

}

else if(q==3 && m=='b') //Transition, δ(q3,b)= q3

{

q=3;

}

else

{

q=4;

break;

}

i++;

m=inputstring[i];

}

if(q==4)

printf(" String not Accepted by this Language");

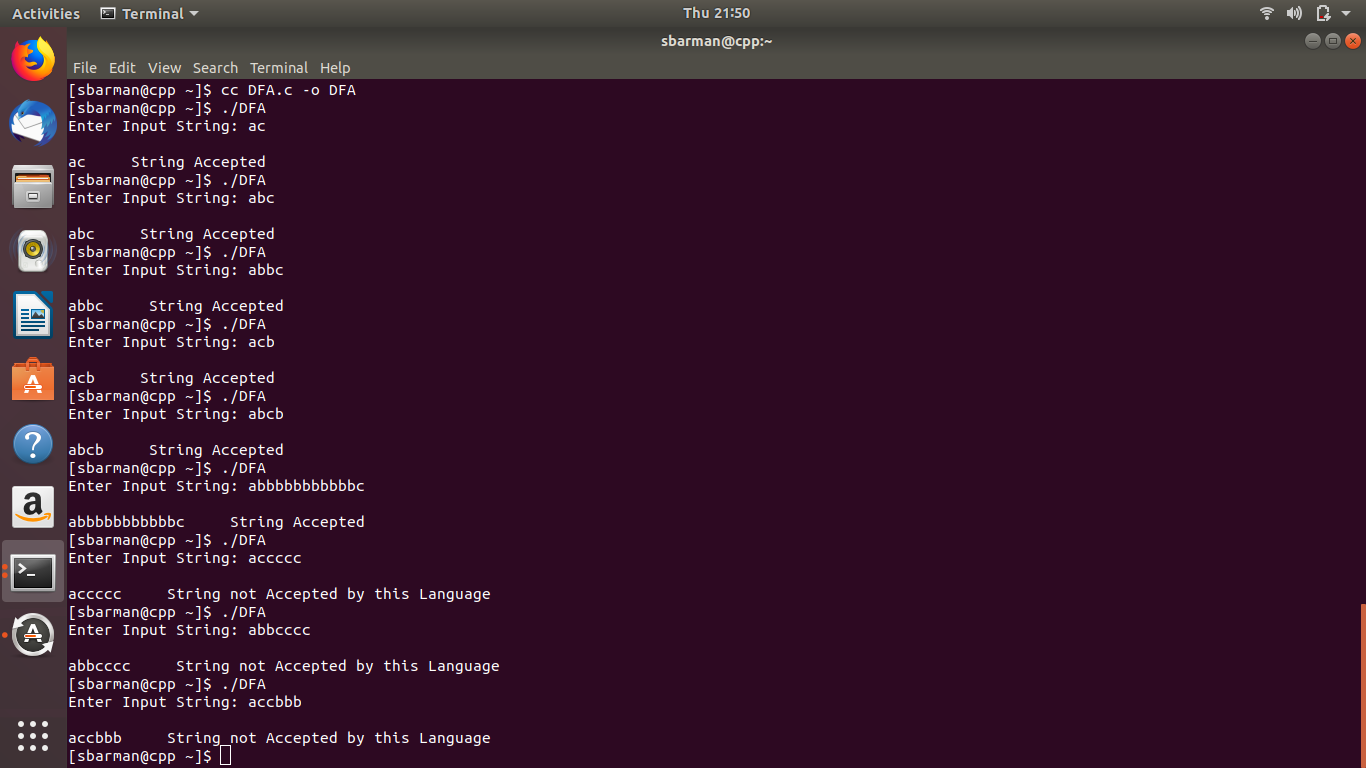
else

printf(" String Accepted");

printf("\n");

return 0;

}



**How to compile the program:**

cc DFA.c -o DFA or, gcc DFA.c -o DFA

This will generate the executable main and you can run it using

./DFA