**PRACTICAL-7**

**AIM : Given NFA for (11+110)\*0 . Write a program to find δ\* (q0, 110) and δ\* (q0, 11)**

**Code:**

#include<stdio.h>

#include<string.h>

int get\_index(char states[],char a)

{

int i;

//check if state is present in set or not.. if present then return its index

for(i=0;states[i]!='\0';i++)

{

if(states[i]==a)

{

return i;

}

}

return -1;

}

void union\_of\_states(char src[],char des[])

{

int i,len;

for(i=0;src[i]!='\0';i++)

{

if(strchr(des,src[i])=='\0')

{

len=strlen(des);

des[len]=src[i];

des[len+1]='\0';

}

}

}

int main()

{

int no\_of\_states,pos\_of\_state;

char states[5];

char zero[5][5],one[5][5];

char temp1[5],temp2[5];

char str[10];

char initial\_state,choice;

int i,j;

//get total no. of states in NFA

printf("Enter number of States: ");

scanf("%d",&no\_of\_states);

//get names of all states

for(i=0;i<no\_of\_states;i++)

{

printf("\nEnter symbol for state %d: ",(i+1));

scanf("\n%c",&states[i]);

}

states[i]='\0';

//get 0 and 1 transition of every state

for(i=0;i<no\_of\_states;i++)

{

printf("\nEnter 0 transition of state %c: ",states[i]);

scanf("%s",&zero[i]);

printf("Enter 1 transition of state %c: ",states[i]);

scanf("%s",&one[i]);

}

//printing the table

printf("\nState 0\t 1\n");

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

for(i=0;i<no\_of\_states;i++)

{

printf("%c \t\t %s \t %s\n",states[i],zero[i],one[i]);

}

while(1)

{

printf("Enter String to be checked.: ");

scanf("%s",&str);

temp1[0]=states[0];

temp1[1]='\0';

temp2[0]='\0';

for(i=0;str[i]!='\0';i++)

{

temp2[0]='\0';

for(j=0;temp1[j]!='\0';j++)

{

pos\_of\_state=get\_index(states,temp1[j]);

if(str[i]=='0')

{

union\_of\_states(zero[pos\_of\_state],temp2);

}

else

{

union\_of\_states(one[pos\_of\_state],temp2);

}

}

strcpy(temp1,temp2);

}

printf("Final States are: %s\n",temp1);

printf("\n\nCheck for new String?\nPress 'y' for yes and 'n' for no.\n");

scanf("\n%c",&choice);

if(choice=='n'||choice=='N')

{

return 0;

}

}

return 0;

}

**Output**:

