NON - DETERMINISTIC FINITE AUTOMATA :

```c
#include<stdio.h>
#include<string.h>
int main()
{
int i,j,k,l,m,next_state[20],n,mat[10][10][10],flag,p,exit;
int num_states,final_state[5],num_symbols,num_final;
int present_state[20],prev_trans,new_trans;
char ch,input[20];
int symbol[5],inp,inp1;
printf("how many states in the nfa: ");
scanf("%d",&num_states);
printf("how many symbols in the input alphabet: ");
scanf("%d",&num_symbols);
for(i=0;i<num_symbols;i++)
{
printf("enter the input symbol %d: ",i+1);
scanf("%d",&symbol[i]);
}
printf("how many final states: ");
scanf("%d",&num_final);
for(i=0;i<num_final;i++)
{
printf("enter the final state %d: ",i+1);
scanf("%d",&final_state[i]);
}
for(i=0;i<10;i++)
{
for(j=0;j<10;j++)
{
for(k=0;k<10;k++)
{
mat[i][j][k]=-1;
}
}
}
for(i=0;i<num_states;i++)
{
for(j=0;j<num_symbols;j++)
{
printf("how many transitions from state %d for the input %d:
",i,symbol[j]);
scanf("%d",&n);
```

```c
for(k=0;k<n;k++)
{
printf("enter the transition %d from state %d for the input %d:
",k+1,i,symbol[j]);
scanf("%d",&mat[i][j][k]);
}
}
}
printf("the transitions are stored as below\n");
for(i=0;i<10;i++)
{
for(j=0;j<10;j++)
{
for(k=0;k<10;k++)
{
if(mat[i][j][k]!=-1)
printf("mat[%d][%d][%d]=%d\n",i,j,k,mat[i][j][k]);
}
}
}
while(1)
{
printf("enter the input string: ");
scanf("%s",input);
present_state[0]=0;
prev_trans=1;
l=strlen(input);
for(i=0;i<l;i++)
{
 if(input[i]=='0')
inp1=0;
else if(input[i]=='1')
inp=1;
else
{
printf("invalid input\n");
exit;
}
for(m=0;m<num_symbols;m++)
{
if(inp1==symbol[m])
{
inp=m;
break;
```

```c
}
}
new_trans=0;
for(j=0;j<prev_trans;j++)
{
k=0;
p=present_state[j];
while(mat[p][inp][k]!=-1)
{
next_state[new_trans++]=mat[p][inp][k];
k++;
}
}
for(j=0;j<new_trans;j++)
{
present_state[j]=next_state[j];
}
prev_trans=new_trans;
}
flag=0;
for(i=0;i<prev_trans;i++)
{
for(j=0;j<num_final;j++)
{
if(present_state[i]==final_state[j])
{
flag=1;
break;
}
}
}
if(flag==1)
printf("accepted\n");
else
printf("not accepted\n");
printf("try with another input\n");
}
}
```