

# EXPERIMENT 4

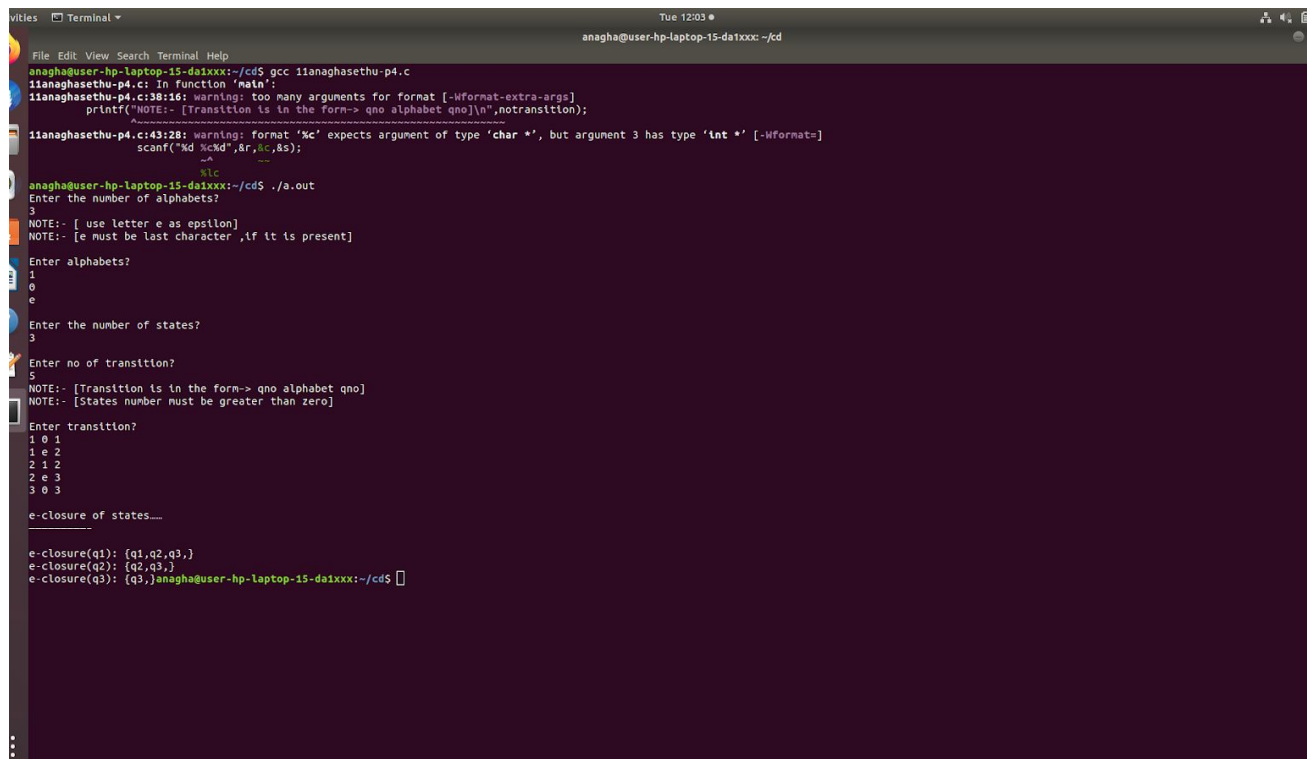
## AIM

Write program to find  $\epsilon$  – closure of all states of any given NFA  
with  $\epsilon$  transition

## ALGORITHM

1. Start
2. Enter number of alphabets, alphabets, number of states, no of transitions and transitions.
3. For  $i=0$  to  $i<\text{no of transitions}$  , insert the transitions to the structure defined.
4. For  $i<\text{no of states}$  , find the closure
5. Display the epsilon closure
6. Stop

## SCREENSHOT



```
vities Terminal * Tue 12:03 *
anagha@user-hp-laptop-15-da1xxx: ~/cd

anagha@user-hp-laptop-15-da1xxx:~/cd$ gcc 11anaghasethu-p4.c
11anaghasethu-p4.c: In function 'main':
11anaghasethu-p4.c:38:16: warning: too many arguments for format [-Wformat-extra-args]
   printf("NOTE:- [Transition is in the form-> qno alphabet qno]\n",notransition);
                  ^
11anaghasethu-p4.c:43:28: warning: format '%c' expects argument of type 'char *', but argument 3 has type 'int *' [-Wformat=]
   scanf("%d %c %d", &r, &c, &s);
                           ^
11anaghasethu-p4.c:43:28: note: (if you used '%i' to match an int, you could mean
   scanf("%d %i %d", &r, &c, &s);
                           ^
anagha@user-hp-laptop-15-da1xxx:~/cd$ ./a.out
Enter the number of alphabets?
3
NOTE:- [ use letter e as epsilon]
NOTE:- [e must be last character ,if it is present]
Enter alphabets?
1
0
e
Enter the number of states?
3
Enter no of transition?
5
NOTE:- [Transition is in the form-> qno alphabet qno]
NOTE:- [States number must be greater than zero]
Enter transition?
1 0 1
1 e 2
2 1 2
2 e 3
3 0 3
e-closure of states....
e-closure(q1): {q1,q2,q3,}
e-closure(q2): {q2,q3,}
e-closure(q3): {q3,}anagha@user-hp-laptop-15-da1xxx:~/cd$
```

## OUTPUT

gcc 11anaghasethu-p4.c

./a.out