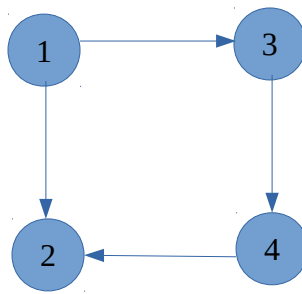


CO322 – LAB 05

1. Find out what is the Transitive Closure of a graph.

Transitive Closure is the reachability matrix to reach from vertex u to vertex v of a graph. One graph is given, we have to find a vertex v which is reachable from another vertex u , for all vertex pairs (u, v) . The final matrix is the Boolean type. When there is a value 1 for vertex u to vertex v , it means that there is at least one path from u to v .

2. Manually compute the Transitive Closure for the following graph:



1	1	1	0
0	1	0	0
0	1	1	1
0	1	0	1

3.

```
1 #include<stdio.h>
2
3 // define no of nodes in the graph
4 #define V 7
5
6 void print(int reach[V][V])
7 {
8     printf("\nTransitive closure of the given graph\n\n");
9     for (int i = 0; i < V; i++)
10     {
11         for (int j = 0; j < V; j++)
12             printf("%d ", reach[i][j]);
13         printf("\n");
14     }
15 }
16
17
18
19 void transitiveClosure(int graph[V][V])
20 {
21     int reach[V][V], i, j, k;
22
23     for (i = 0; i < V; i++)
24         for (j = 0; j < V; j++)
25             reach[i][j] = graph[i][j];
26
27     for (k = 0; k < V; k++)
28     {
29         for (i = 0; i < V; i++)
30         {
31             for (j = 0; j < V; j++)
32             {
33                 if (i==j){
34                     reach[i][j] = 1;
35                 }else{
36                     reach[i][j] = reach[i][j] || (reach[i][k] && reach[k][j]);
37                 }
38             }
39         }
40     }
41
42     print(reach);
43 }
44
45 int main()
46 {
47     // Insert adjacency matrix of the graph
48     int graph[V][V] = { {0, 1, 1, 0, 0, 0, 0},
49                         {0, 0, 0, 0, 1, 0, 0},
50                         {0, 0, 0, 0, 1, 0, 0},
51                         {0, 1, 0, 0, 1, 0, 1},
52                         {0, 0, 0, 0, 0, 0, 1},
53                         {0, 0, 0, 1, 0, 0, 0},
54                         {0, 1, 0, 0, 0, 0, 0} };
55
56     /*int graph[V][V] = {
57         {0,1,1,0},
58         {0,0,0,0},
59         {0,0,0,1},
60     };
61 */
62
63     transitiveClosure(graph);
64 }
```

```
Transitive closure of the given graph
1 1 1 0
0 1 0 0
0 1 1 1
0 1 0 1
```