

WARSHALLS

```
#include <stdio.h>
```

```
#define MAX_VERTICES 10
```

```
void warshall(int graph[MAX_VERTICES][MAX_VERTICES], int V) {
```

```
    int reach[MAX_VERTICES][MAX_VERTICES];
```

```
    for (int i = 0; i < V; i++) {
```

```
        for (int j = 0; j < V; j++) {
```

```
            reach[i][j] = graph[i][j];
```

```
        }
```

```
    }
```

```
    for (int k = 0; k < V; k++) {
```

```
        for (int i = 0; i < V; i++) {
```

```
            for (int j = 0; j < V; j++) {
```

```
                if (reach[i][k] && reach[k][j]) {
```

```
                    reach[i][j] = 1;
```

```
                }
```

```
            }
```

```
        }
```

```
    }
```

```
    printf("Transitive Closure (Reachability Matrix):\n");
```

```
    for (int i = 0; i < V; i++) {
```

```
        for (int j = 0; j < V; j++) {
```

```
            printf("%d ", reach[i][j]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```

}

int main() {
    int V, E;

    printf("Enter the number of vertices: ");
    scanf("%d", &V);

    int graph[MAX_VERTICES][MAX_VERTICES] = {0};

    printf("Enter the number of edges: ");
    scanf("%d", &E);

    printf("Enter the edges (u, v) for each edge (0-indexed):\n");
    for (int i = 0; i < E; i++) {
        int u, v;
        scanf("%d %d", &u, &v);
        graph[u][v] = 1; // Set the edge (u -> v) as 1 (reachable)
    }
    warshall(graph, V);
    return 0;
}

```

OUTPUT:

```

Enter the number of vertices: 5
Enter the number of edges: 5
Enter the edges (u, v) for each edge (0-indexed):
1 2
3 4
5 6
7 8
9 8
Transitive Closure (Reachability Matrix):
0 0 0 0 0
0 0 1 0 0
0 0 0 0 0
0 0 0 0 1
0 0 0 0 0

Process returned 0 (0x0)   execution time : 18.700 s
Press any key to continue.

```