6. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using distance vector routing algorithm.

```
#include <stdio.h>
#define INF 999
int main() {
   int nodes;
   int cost[10][10], dist[10][10], nextHop[10][10];
   printf("Enter number of nodes: ");
   scanf("%d", &nodes);
   printf("Enter cost matrix (use %d for INF):\n", INF);
   for (int i = 0; i < nodes; i++) {
      for (int j = 0; j < nodes; j++) {
         scanf("%d", &cost[i][j]);
         dist[i][j] = cost[i][j];
         if (i == j)
            nextHop[i][j] = i;
         else if (cost[i][j] != INF)
            nextHop[i][j] = j;
         else
            nextHop[i][j] = -1;
      }
   }

   // Distance Vector Algorithm (Bellman-Ford)
   int updated;
   do {
      updated = 0;
      for (int i = 0; i < nodes; i++) {
         for (int j = 0; j < nodes; j++) {
            for (int k = 0; k < nodes; k++) {
```

```
                if (dist[i][k] + dist[k][j] < dist[i][j]) {
                    dist[i][j] = dist[i][k] + dist[k][j];
                    nextHop[i][j] = nextHop[i][k];
                    updated = 1;
                }
            }
        }
    }
} while (updated);

// Print routing tables
for (int i = 0; i < nodes; i++) {
    printf("\nRouting table for Router %d:\n", i + 1);
    printf("Destination\tNext Hop\tCost\n");
    for (int j = 0; j < nodes; j++) {
        printf("    %d\t\t", j + 1);
        if (nextHop[i][j] == -1)
            printf(" - \t\t");
        else
            printf(" %d \t\t", nextHop[i][j] + 1);
        printf("%d\n", dist[i][j]);
    }
}
return 0;
}
```

OUTPUT

Enter number of nodes: 5

Enter cost matrix (use 999 for INF):

0 2 999 3 4
2 0 5 999 6
999 5 0 999 2

3 999 999 0 7
4 6 2 7 0

Routing table for Router 1:

Destination	Next Hop	Cost
1	1	0
2	2	2
3	5	6
4	4	3
5	5	4

Routing table for Router 2:

Destination	Next Hop	Cost
1	1	2
2	2	0
3	3	5
4	1	5
5	5	6

Routing table for Router 3:

Destination	Next Hop	Cost
1	5	6
2	2	5
3	3	0
4	5	9
5	5	2

Routing table for Router 4:

Destination	Next Hop	Cost
1	1	3
2	1	5
3	1	9
4	4	0
5	5	7

Routing table for Router 5:

Destination	Next Hop	Cost
1	1	4
2	2	6
3	3	2
4	4	7
5	5	0