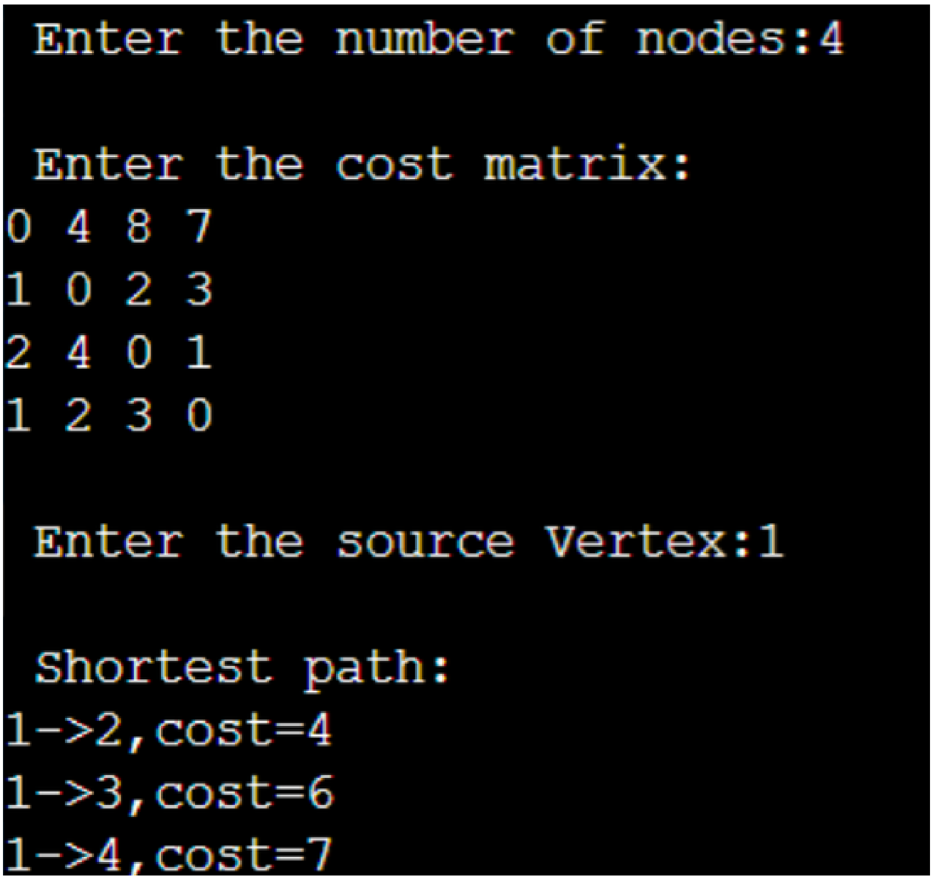
7) From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.

```
#include <stdio.h>
#define INFINITY 999

void dijkstra(int n, int v, int cost[10][10], int dist[100]) {
   int i, u, count, w, flag[10], min;

   for (i = 1; i <= n; i++)
      flag[i] = 0, dist[i] = cost[v][i];

   count = 2;
   while (count <= n) {
      min = INFINITY;
      for (w = 1; w <= n; w++)
         if (dist[w] < min && !flag[w])
            min = dist[w], u = w;

      flag[u] = 1;
      count++;

      for (w = 1; w <= n; w++)
         if ((dist[u] + cost[u][w] < dist[w]) && !flag[w])
            dist[w] = dist[u] + cost[u][w];
   }
}

int main() {
   int n, v, i, j, cost[10][10], dist[10];

   printf("\n Enter the number of nodes:");
   scanf("%d", &n);

   printf("\n Enter the cost matrix:\n");
   for (i = 1; i <= n; i++)
      for (j = 1; j <= n; j++) {
         scanf("%d", &cost[i][j]);
         if (cost[i][j] == 0)
            cost[i][j] = INFINITY;
      }
```

```
 printf("\n Enter the source Vertex:");
 scanf("%d", &v);

 dijkstra(n, v, cost, dist);

 printf("\n Shortest path:\n");
 for (i = 1; i <= n; i++)
 if (i != v)
 printf("%d->%d, cost=%d\n", v, i, dist[i]);

 return 0;
}
```



```
 Enter the number of nodes:4

 Enter the cost matrix:
0 4 8 7
1 0 2 3
2 4 0 1
1 2 3 0

 Enter the source Vertex:1

 Shortest path:
1->2,cost=4
1->3,cost=6
1->4,cost=7
```