Prog: Implement Dijkstra's algorith to compute the shortest path through a graph. 1BM18CS080

```
#include <stdio.h>
#include <conio.h>
#include <process.h>
#include<string.h>
#include< math.h>
#define IN 99
#define N 6
int dijkstra(int cost[][N], int source, int target);
int dijsktra(int cost[][N], int source , int target).
{
  int dist[N] , prev[N], selected[N]= {0}, i, m, min, start,d,j;
    char path[N];
    for(i=1; i<N ; i++)
  {
    dist[i] = IN ;
  } prev[i] = -1;

  start = source;
  selected [start] = 1;
  dist [start ] = 0;
 while (selected [target] == 0)
  {
    min = IN ;
    m = 0;
    for (i=1 ; i<N; i++)
    {
      d = dist[start] + cost [start][i];
      if (d< dist[i] && selected [i] == 0)
      {
        dist[i] = d ;
          prev[i] = start ;
      }
```

```
if (min > dist[i] && selected[i]==0)
    {
      min = dist[i];
      m = i;
    }
  }
  start = m;
  selected[start] = 1;
}
  start = target;
  j=0;
  while (start != -1)
{ path[j++] = start+65;
  start = prev[start];
}
  path[j] = '\0';
  strrev(path);
  printf("%s", path);
  return dist[target];
}
int main()
{
  int cost[N][N], i, j, w, ch, co;
  int source, target, x, y;
printf("\t Shortest Path Algorithm
(BY DIJKSRA's ALGORITHM\n\n");
for(i=1; i<N; i++)
 for(j=1; j<N; j++)
  cost[i][j] = IN;
  for(x=1; x<N; x++)
{
  for(y=x+1; y<N; y++)
{
  Printf("Enter the weight of the
  path b/n node %d and %d:",
     x, y);
  scanf("%d" &w);
 cost[x][y] = cost[y][x] = w;
}
  Printf("\n");
}
Printf("\nEnter the source:");
 scanf("%d", &source);
Printf("\n Enter the target");
 scanf("%d", &target);
 co = dijSlctra(cost, source, target);
   Printf("\n shortest path: %d",
       co);
}
```