IBM18CS002

Implement Dijkstra's algorithm to compute shortest path.

Pseudo Code

```
algo(int a[ ][10], int n)
{
        int s[n];
        for i 0 to n-1
            d[i] = a[source][i];
            p[i] = source
            s[i] = 0

        s[source] = 1
        for c 0 to n-1
            min = 999
            for j 0 to n-1
                if d[j] < min && s[j] != 1 then
                    min = d[j]
                    u = j
            end for
            s[u] = 1
            for i 0 to n-1
                if min + a[u][i] < d[i]
                    d[i] = min + a[u][i]
                    p[i] = u
            end for
        end for
}
```

```
void main()
{
    cout << "Enter no of vertices"
    cin >> n
    cout << "Enter adjacency matrix";
    for i 0 to n-1
        for j 0 to n-1
            cin >> a[i][j]
    cout << "Enter source vertex"
    cin >> source
    cout << "Shortest paths from vertex "<< source << " are:";
    Algo(a, n)
    for i 0 to n-1
        k = i
        while k != source
            cout << k;
            k = P[k];
    cout << source << " = ";
    cout << Path cost =" << d[i];
}
```