#include <stdio.h>

#include <limits.h>

int V = 0;

int minDistance(int dist[], int visited\_node[]){

int min = INT\_MAX, min\_index;

for (int v = 0; v < V; v++)

if (visited\_node[v] == 0 && dist[v] <= min) {

min = dist[v];

min\_index = v;

}

return min\_index;

}

void print(int dist[]) {

char a = 'A';

printf("Vertex \t\t Distance from Source\n");

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

printf(" %c \t\t %d\n", a + i, dist[i]);

}

void djkstra(int graph[V][V], int src) {

int dist[V], visited\_node[V];

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

dist[i] = INT\_MAX;

visited\_node[i] = 0;

}

dist[src] = 0;

for (int count = 0; count < V - 1; count++) {

int u = minDistance(dist, visited\_node);

visited\_node[u] = 1;

for (int v = 0; v < V; v++) {

if (!visited\_node[v] && graph[u][v] && dist[u] != INT\_MAX && dist[u] + graph[u][v] < dist[v])

dist[v] = dist[u] + graph[u][v];

}

}

print(dist);

}

int main(){

int n, e, i;

printf("Enter number of nodes: ");

scanf("%d", &n);

printf("Enter number of edges: ");

scanf("%d", &e);

int graph[n][n];

V = n;

for (i = 0; i < e; i++) {

int firstNode, secondNode, weight;

printf("Enter firstNode secondNode: ");

scanf("%d %d", &firstNode, &secondNode);

printf("Enter edge weight: ");

scanf("%d", &weight);

graph[firstNode][secondNode] = weight;

graph[secondNode][firstNode] = weight;

}

djkstra(graph, 0);

return 0;

}

