Aim: write a program in c to implement dijkstra algorithm

Algorithm:

1. Initialize the distances from the source node to all other nodes as infinity and keep track of which nodes have been processed.
2. Choose the node with the shortest distance from the source that has not been processed.
3. Mark that node as processed.
4. Update the distances of the neighboring nodes of the processed node if the new distance is shorter.
5. Repeat steps 2-4 until all nodes have been processed.
6. The final distances from the source node will be the solution to the problem.

Source Code:

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#define MAX\_NODES 100

#define MAX\_DISTANCE 1000000

*int* n;

*int* adjacency\_matrix[MAX\_NODES][MAX\_NODES];

*int* distances[MAX\_NODES];

*bool* processed[MAX\_NODES];

*int* find\_min\_distance\_node() {

*int* min\_distance = MAX\_DISTANCE;

*int* min\_distance\_node = -1;

  for (*int* i = 0; i < n; i++) {

    if (!processed[i] && distances[i] < min\_distance) {

      min\_distance = distances[i];

      min\_distance\_node = i;

    }

  }

  return min\_distance\_node;

}

*void* dijkstra(*int* source) {

  for (*int* i = 0; i < n; i++) {

    distances[i] = MAX\_DISTANCE;

    processed[i] = *false*;

  }

  distances[source] = 0;

  for (*int* i = 0; i < n; i++) {

*int* min\_distance\_node = find\_min\_distance\_node();

    processed[min\_distance\_node] = *true*;

    for (*int* j = 0; j < n; j++) {

      if (!processed[j] && adjacency\_matrix[min\_distance\_node][j] != 0) {

*int* distance = distances[min\_distance\_node] + adjacency\_matrix[min\_distance\_node][j];

        if (distance < distances[j]) {

          distances[j] = distance;

        }

      }

    }

  }

}

*int* main() {

  printf("Enter the number of nodes: ");

  scanf("%d", &n);

  printf("Enter the adjacency matrix representation of the graph:\n");

  for (*int* i = 0; i < n; i++) {

    for (*int* j = 0; j < n; j++) {

      scanf("%d", &adjacency\_matrix[i][j]);

    }

  }

*int* source;

  printf("Enter the source node: ");

  scanf("%d", &source);

  dijkstra(source);

  printf("Distances from the source node:\n");

  for (*int* i = 0; i < n; i++) {

    printf("%d ", distances[i]);

  }

  printf("\n");

  return 0;

}

Output:

Enter the number of nodes: 6

Enter the adjacency matrix representation of the graph:

0 4 5 0 0 0

4 0 11 9 7 0

5 11 0 0 3 0

0 9 0 0 13 2

0 7 3 13 0 6

0 0 0 2 6 0

Enter the source node: 0

Distances from the source node:

0 4 5 13 8 14