Aim: write a program in c to perform travelling salesman problem

Algorithm:

1. Start with the first city as the current city.
2. Try every possible city as the next city and recursively continue the search.
3. If all cities have been visited, calculate the distance of the path and compare it to the current minimum distance. If it is shorter, update the minimum distance and store the path.
4. Backtrack by marking the current city as unvisited and trying the next city.
5. Repeat steps 2-4 until all possible paths have been tried.
6. The minimum distance and path found in step 5 will be the solution to the traveling salesman problem.

Source Code: #include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#define MAX\_CITIES 100

#define MAX\_DISTANCE 10000

*int* n;

*int* dist[MAX\_CITIES][MAX\_CITIES];

*int* path[MAX\_CITIES];

*bool* visited[MAX\_CITIES];

*int* min\_distance = MAX\_DISTANCE;

*int* calculate\_distance(*int* path[]) {

*int* distance = 0;

  for (*int* i = 0; i < n - 1; i++) {

    distance += dist[path[i]][path[i + 1]];

  }

  distance += dist[path[n - 1]][path[0]];

  return distance;

}

*void* search(*int* current\_city, *int* current\_distance, *int* current\_index) {

  if (current\_index == n) {

    if (current\_distance < min\_distance) {

      min\_distance = current\_distance;

      for (*int* i = 0; i < n; i++) {

        path[i] = path[i];

      }

    }

    return;

  }

  for (*int* i = 0; i < n; i++) {

    if (!visited[i]) {

      visited[i] = *true*;

      path[current\_index] = i;

      search(i, current\_distance + dist[current\_city][i], current\_index + 1);

      visited[i] = *false*;

    }

  }

}

*int* main() {

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

  scanf("%d", &n);

  printf("Enter the distances between the cities (use -1 for infinity):\n");

  for (*int* i = 0; i < n; i++) {

    for (*int* j = 0; j < n; j++) {

      scanf("%d", &dist[i][j]);

    }

  }

  path[0] = 0;

  visited[0] = *true*;

  search(0, 0, 1);

  printf("Shortest path: ");

  for (*int* i = 0; i < n; i++) {

    printf("%d ", path[i]);

  }

  printf("\n");

  printf("Shortest distance: %d\n", min\_distance);

  printf("Path Cost: %d",calculate\_distance(path));

  return 0;

}

Output:

Enter the number of cities: 6

Enter the distances between the cities (use -1 for infinity):

-1 10 15 20 -1 8

5 -1 9 10 8 -1

6 13 -1 12 -1 5

8 8 9 -1 6 -1

-1 10 -1 6 -1 -1

10 -1 5 -1 -1 -1

Shortest path: 0 5 4 3 2 1

Shortest distance: 8

Path Cost: 40