Write a c++ code for travelling salesman problem

#include <bits/stdc++.h>

using namespace std;

#define V 4

int travllingSalesmanProblem(int graph[][V], int s)

{

    vector<int> vertex;

    for (int i = 0; i < V; i++)

        if (i != s)

            vertex.push\_back(i);

    .

    int min\_path = INT\_MAX;

    do {

        // store current Path weight(cost)

        int current\_pathweight = 0;

        // compute current path weight

        int k = s;

        for (int i = 0; i < vertex.size(); i++) {

            current\_pathweight += graph[k][vertex[i]];

            k = vertex[i];

        }

        current\_pathweight += graph[k][s];

        // update minimum

        min\_path = min(min\_path, current\_pathweight);

    } while (

        next\_permutation(vertex.begin(), vertex.end()));

    return min\_path;

}

// Driver Code

int main()

{

    // matrix representation of graph

    int graph[][V] = { { 0, 10, 15, 20 },

                       { 10, 0, 35, 25 },

                       { 15, 35, 0, 30 },

                       { 20, 25, 30, 0 } };

    int s = 0;

    cout << travllingSalesmanProblem(graph, s) << endl;

    return 0;

}