prims-final.c++

```cpp
#include <iostream>
using namespace std;

const int INFI = 99;
int adj_matrix[10][10];

void prims(int n) {
    int mst[n-1][2], near[n+1], cost = 0;

    for (int i=1; i<=n; i++)
        near[i] = INFI;

    int u,v,min = INFI;
    for (int i=1; i<=n; i++) {
        for (int j=i; j<=n; j++) {
            if (i != j && adj_matrix[i][j] < min){
                min = adj_matrix[i][j];
                u = i;
                v = j;
            }
        }
    }
    cost += min;
    mst[0][0] = u;
    mst[0][1] = v;
    near[u] = near[v] = 0;

    for (int i=1; i<=n; i++) {
        if (near[i] != 0) {
            if (adj_matrix[i][u] < adj_matrix[i][v])
                near[i] = u;
            else
                near[i] = v;
        }
    }

    for (int i=1; i<n-1; i++) {
        min = INFI;
        int k;
        for (int j=1; j<=n; j++) {
            if (near[j] != 0 && adj_matrix[j][near[j]] < min) {
                min = adj_matrix[j][near[j]];
                k = j;
            }
        }
        cost += min;
        mst[i][0] = k;
        mst[i][1] = near[k];
        near[k] = 0;

        for(int j=1; j<=n; j++) {
```

```
            if (near[j] != 0 && adj_matrix[j][k] < adj_matrix[j][near[j]])
                near[j] = k;
        }
    }


    cout << "\nThe edges of minimum spanning tree:\n";
    for (int i=0; i<n-1; i++)
        cout << "(" << mst[i][0] << ", " << mst[i][1] << ")\n";
    cout << "The cost of minimum spanning tree:\n" << cost << endl;
}

int main()
{
    int n;
    printf("\nEnter the number of vertices: ");
    cin >> n;
    printf("\nEnter the adjacency matrix: ");
    for(int i=1; i<=n; i++)
        for(int j=1; j<=n; j++)
            cin >> adj_matrix[i][j];
    prims(n);
    return 0;
}
```