```c
#include <stdio.h>
int parent[10];
int find(int i);
void kruskal(int cost[10][10], int n);
int main() {
    int n;
    int cost[10][10];
    printf("Enter the value of n: ");
    scanf("%d", &n);
    printf("Enter the cost matrix:\n");
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            scanf("%d", &cost[i][j]);
    kruskal(cost, n);
    return 0;
}
int find(int i) {
    while (parent[i] >= 0)
        i = parent[i];
    return i;
}
void kruskal(int cost[10][10], int n) {
    int min_cost = 0, edge_count = 0,a, b, u, v, min;
for (int i = 0; i < n; i++)
        parent[i] = -1;
while (edge_count < n - 1) {
        min = 999;
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (cost[i][j] < min) {
                    min = cost[i][j];
                    a = i;
                    b = j;
                }
            }
        }
        u = find(a);
        v = find(b);
        if (u != v) {
            parent[v] = u;
            printf("%d to %d = %d\n", a, b, min);
            min_cost = min_cost+min;
            edge_count++;
        }
        cost[a][b] = cost[b][a] = 999;
    }
    printf("Minimum Cost: %d\n", min_cost);
}

```