

25.Implementation of Minimum Spanning Tree using Prim's Algorithm

```
#include <stdio.h>

#define INF 9999

#define MAX 20

int main() {

    int n, i, j;

    int graph[MAX][MAX];

    int visited[MAX] = {0};

    int no_of_edges = 1;

    int min, total_cost = 0;

    int u, v;

    printf("Enter number of vertices: ");

    scanf("%d", &n);

    printf("Enter the adjacency matrix (use 0 if no edge):\n");

    for (i = 0; i < n; i++)

        for (j = 0; j < n; j++) {

            scanf("%d", &graph[i][j]);

            if (graph[i][j] == 0)

                graph[i][j] = INF;

        }

    visited[0] = 1;

    printf("\nEdges in the Minimum Spanning Tree:\n");

    while (no_of_edges < n) {

        min = INF;

        for (i = 0; i < n; i++) {

            if (visited[i]) {

                for (j = 0; j < n; j++) {

                    if (!visited[j] && graph[i][j] < min) {

                        min = graph[i][j];

                        u = i;

                        v = j;

                    }

                }

            }

        }

        // Add edge (u, v) to the MST
        total_cost += min;
        no_of_edges++;

        // Mark vertex v as visited
        visited[v] = 1;

    }

    printf("Total cost of the Minimum Spanning Tree: %d", total_cost);

}
```

```

    }

    }

}

printf("Edge %d: (%d - %d) cost: %d\n", no_of_edges, u, v, min);

total_cost += min;

visited[v] = 1;

no_of_edges++;

}

printf("\nMinimum Cost = %d\n", total_cost);

return 0;

}

```

main.c	Output
<pre> 1 #include <stdio.h> 2 3 #define INF 9999 4 #define MAX 20 5 6 int main() { 7 int n, i, j; 8 int graph[MAX][MAX]; 9 int visited[MAX] = {0}; 10 int no_of_edges = 1; 11 int min, total_cost = 0; 12 int u, v; 13 14 printf("Enter number of vertices: "); 15 scanf("%d", &n); 16 17 printf("Enter the adjacency matrix (use 0 if no edge):\n"); 18 for (i = 0; i < n; i++) 19 for (j = 0; j < n; j++) { 20 scanf("%d", &graph[i][j]); 21 if (graph[i][j] == 0) 22 graph[i][j] = INF; 23 } 24 25 visited[0] = 1; 26 27 printf("\nEdges in the Minimum Spanning Tree:\n"); 28 </pre>	<pre> Enter number of vertices: 5 Enter the adjacency matrix (use 0 if no edge): 1 2 3 4 5 4 5 6 7 8 9 8 7 45 6 1 2 3 6 5 4 9 8 7 4 Edges in the Minimum Spanning Tree: Edge 1: (0 - 1) cost: 2 Edge 2: (0 - 2) cost: 3 Edge 3: (0 - 3) cost: 4 Edge 4: (0 - 4) cost: 5 Minimum Cost = 14 === Code Execution Successful === </pre>