Prim's algorithm

```c
#include <stdio.h>
int n, m, e = 0;
float sum = 0;
float costs[100][100];
int VT[100], ET[100][2], vis[20];

void prims()
{
    int u, v;
    int x = 1, j, K, min;
    VT[x] = 1;
    vis[x] = 1;
    for (int i = 1; i < n; i++)
    {
        j = x;
        min = 999;
        while (j > 0)
        {
            K = VT[j];
            for (int m = 2; m <= n; m++)
            {
                if (costs[K][m] < min && vis[m] == 0)
                {
                    min = costs[K][m];
                    u = K;
                    v = m;
                }
            }
            j--;
        }
        VT[++x] = v;
        ET[i][0] = u;
        ET[i][1] = v;
        e++;
        vis[v] = 1;
        sum += costs[u][v];
    }
}
```

```
void main()
{
    int u, v;
    float w;
    printf("\nEnter the number of vertices: ");
    scanf("%d", &n);
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= n; j++)
        {
            if (i == j)
                costs[i][j] = 0;
            else
                costs[i][j] = 999;
        }
    }
    printf("Enter the number of egdes: ");
    scanf("%d", &m);
    printf("Enter vertices of edge with its weight: \n");
    for (int i = 1; i <= m; i++)
    {
        scanf("%d%d%f", &u, &v, &w);
        costs[u][v] = costs[v][u] = w;
    }
    for (int i = 1; i <= n; i++)
    {
        vis[i] = 0;
    }
    prims();
    printf("\nMinimum Cost is: %.2f\n", sum);
    printf("Edges of Minimum spanning tree\n");
    for (int i = 1; i <= e; i++)
    {
        printf("%d-->%d\n", ET[i][0], ET[i][1]);
    }
}
```

OUTPUT:

```
Enter the number of vertices: 5
Enter the number of egdes: 5
Enter vertices of edge with its weight:
1 2 1
1 4 2
1 3 5
3 4 3
4 5 1.5

Minimum Cost is: 7.50
Edges of Minimum spanning tree
1-->2
1-->4
4-->5
4-->3
```