Find Minimum Cost Spanning Tree of a given undirected graph using

Prim’s algorithm.

```
#include <stdio.h>
#define INF 9999
#define MAX 100

int main() {
   int cost[MAX][MAX], visited[MAX] = {0};
   int n, i, j, min, u, v, ne = 1, total_cost = 0;

   printf("Enter the number of vertices: ");
   scanf("%d", &n);

   printf("Enter the cost adjacency matrix (use %d for no edge):\n", INF);
   for (i = 0; i < n; i++)
      for (j = 0; j < n; j++)
         scanf("%d", &cost[i][j]);

   visited[0] = 1;

   while (ne < n) {
      min = INF;
      for (i = 0; i < n; i++) {
         if (visited[i]) {
            for (j = 0; j < n; j++) {
               if (!visited[j] && cost[i][j] < min) {
                  min = cost[i][j];
                  u = i;
                  v = j;
```

```
            }
          }
        }
      }

      if (!visited[v]) {
        printf("Edge %d: (%d -> %d) cost = %d\n", ne++, u, v, min);
        total_cost += min;
        visited[v] = 1;
      }

      cost[u][v] = cost[v][u] = INF;
    }

    printf("Minimum cost = %d\n", total_cost);

    return 0;
}
```

Output

```
Enter the number of vertices: 2
Enter the cost adjacency matrix (use 9999 for no edge):
2
3
3
3
Edge 1: (0 -> 1) cost = 3
Minimum cost = 3


=== Code Execution Successful ===
```