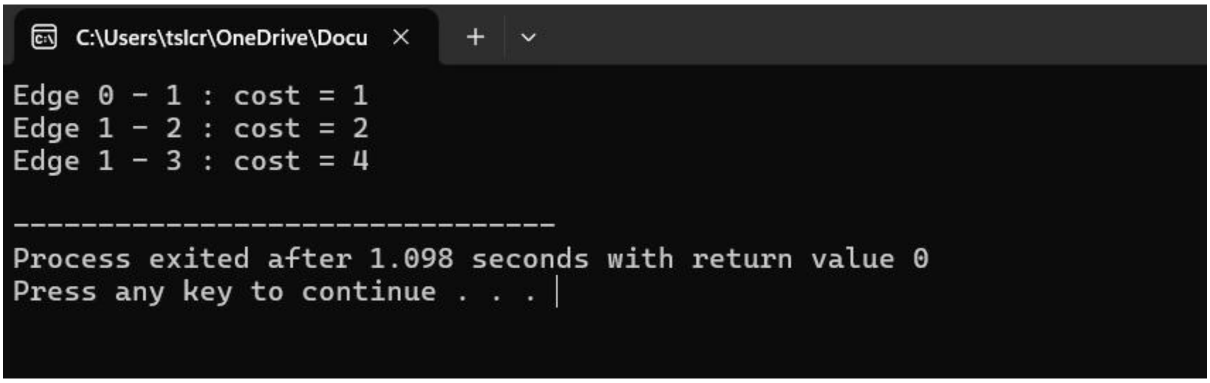
35. Implementation of Minimum Spanning Tree using Kruskal Algorithm.

```
#include <stdio.h>

int parent[4];

int find(int i)

{

   while (i != parent[i]) i = parent[i];

   return i;

}

void unite(int i, int j) {

   int a = find(i);

   int b = find(j);

   parent[b] = a;

}

int main() {

   int cost[4][4] = {

      {999, 1, 3, 999},

      {1, 999, 2, 4},

      {3, 2, 999, 5},

      {999, 4, 5, 999}

   };

   for (int i = 0; i < 4; i++) parent[i] = i;

   int edges = 0;

   while (edges < 3) {

      int min = 999, a = -1, b = -1;

      for (int i = 0; i < 4; i++)

         for (int j = 0; j < 4; j++)

            if (find(i) != find(j) && cost[i][j] < min) {

               min = cost[i][j];
```

```
                a = i; b = j;
            }
        unite(a, b);
        printf("Edge %d - %d : cost = %d\n", a, b, min);
        cost[a][b] = cost[b][a] = 999;
        edges++;
    }
    return 0;
}
```

OUTPUT



```
Edge 0 - 1 : cost = 1
Edge 1 - 2 : cost = 2
Edge 1 - 3 : cost = 4

--------------------------------
Process exited after 1.098 seconds with return value 0
Press any key to continue . . .
```