**合肥工业大学**

**物联网工程专业**

姓 名： 党存远

学 号： 2022217587

班 级： 物联网工程22-2

实验地点： 计算机中心楼

实验时间： 12.11

1. 实验目的和要求

实验一:油管铺设

实验目的：

运用最小生成树思想和求最小生成树程序解决实际问题

实验过程

八口海上油井相互间距离如下表，其中1号井离海岸最近，为5km。问从海岸经1号井铺设油管把各井连接起来，怎样连油管长度最短（为便于检修，油管只准在油井处分叉）？

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *从～到* | *2* | *3* | *4* | *5* | *6* | *7* | *8* |
| 1 | 1.3 | 2.1 | 0.9 | 0.7 | 1.8 | 2.0 | 1.8 |
| 2 |  | 0.9 | 1.8 | 1.2 | 2.8 | 2.3 | 1.1 |
| 3 |  |  | 2.6 | 1.7 | 2.5 | 1.9 | 1.0 |
| 4 |  |  |  | 0.7 | 1.6 | 1.5 | 0.9 |
| 5 |  |  |  |  | 0.9 | 1.1 | 0.8 |
| 6 |  |  |  |  |  | 0.6 | 1.0 |
| 7 |  |  |  |  |  |  | 0.5 |

实验要求：

(1)选择一种编程语言，利用最小生成树Prim算法给出最优方案

输入：加权图邻接矩阵

输出：最小生成树邻接矩阵，油管最短距离

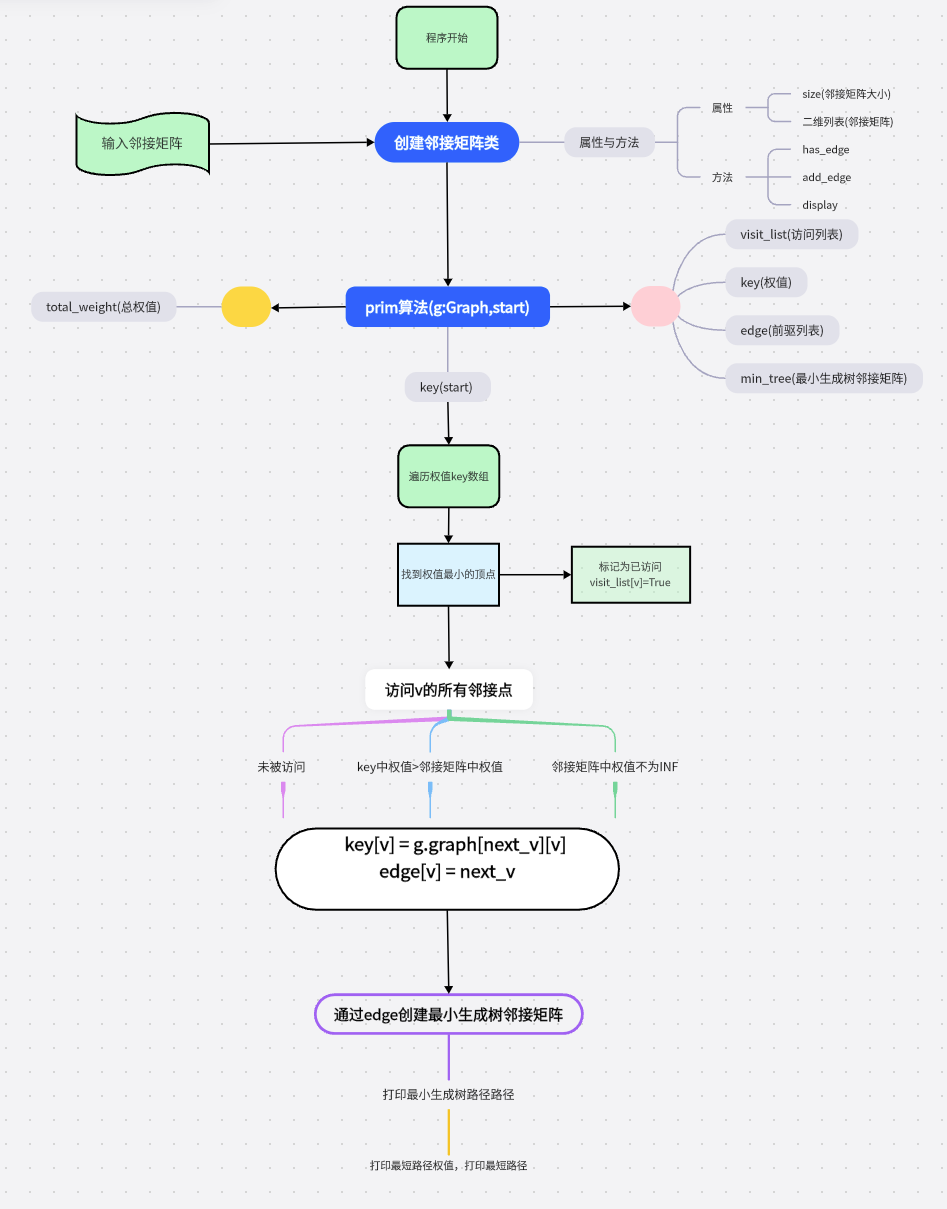
1. 实验环境和工具

编译器：Pycharm

运行环境：Anaconda

1. 实验结果
   1. 算法流程图

实验一算法流程图:



* 1. 程序核心代码

实验一Prim生成最小生成树:

1、Prim算法实现:

def prim(g: Graph, strat: int):  
 visit\_list = [False] \* g.size # 记录顶点是否在最小生成树中  
 key = [INF] \* g.size # 记录权值  
 edge = [-1] \* g.size  
 min\_tree = Graph(g.size)  
  
 total\_weight = 0.0 # 最小生成树的总权值  
 key[strat] = 0.0 # 将开始权值赋值为最小值  
 for \_ in range(g.size):  
 next\_v = min(range(g.size), key=lambda i: key[i] if not visit\_list[i] else INF) # 找到当前节点所连边且未被访问的最小节点  
 visit\_list[next\_v] = True  
  
 for v in range(g.size):  
 if (g.graph[next\_v][v] > 0) and not (visit\_list[v]) and (key[v] > g.graph[next\_v][v]):  
 key[v] = g.graph[next\_v][v]  
 edge[v] = next\_v # 表示其前驱  
 for i in range(1, g.size):  
 from\_v = edge[i]  
 to\_v = i  
 val = g.graph[from\_v][to\_v]  
 min\_tree.add\_edge(from\_v, to\_v, val)  
 if val == INF:  
 pass  
 else:  
 total\_weight += val  
 print(f"最小生成树形状如图： ")  
 for i in range(1, g.size):  
 from\_v = edge[i]  
 to\_v = i  
 val = g.graph[from\_v][to\_v]  
 if val == INF:  
 pass  
 else:  
 print(f"{from\_v + 1} --- {to\_v + 1} val is {val}")  
  
 return min\_tree, total\_weight

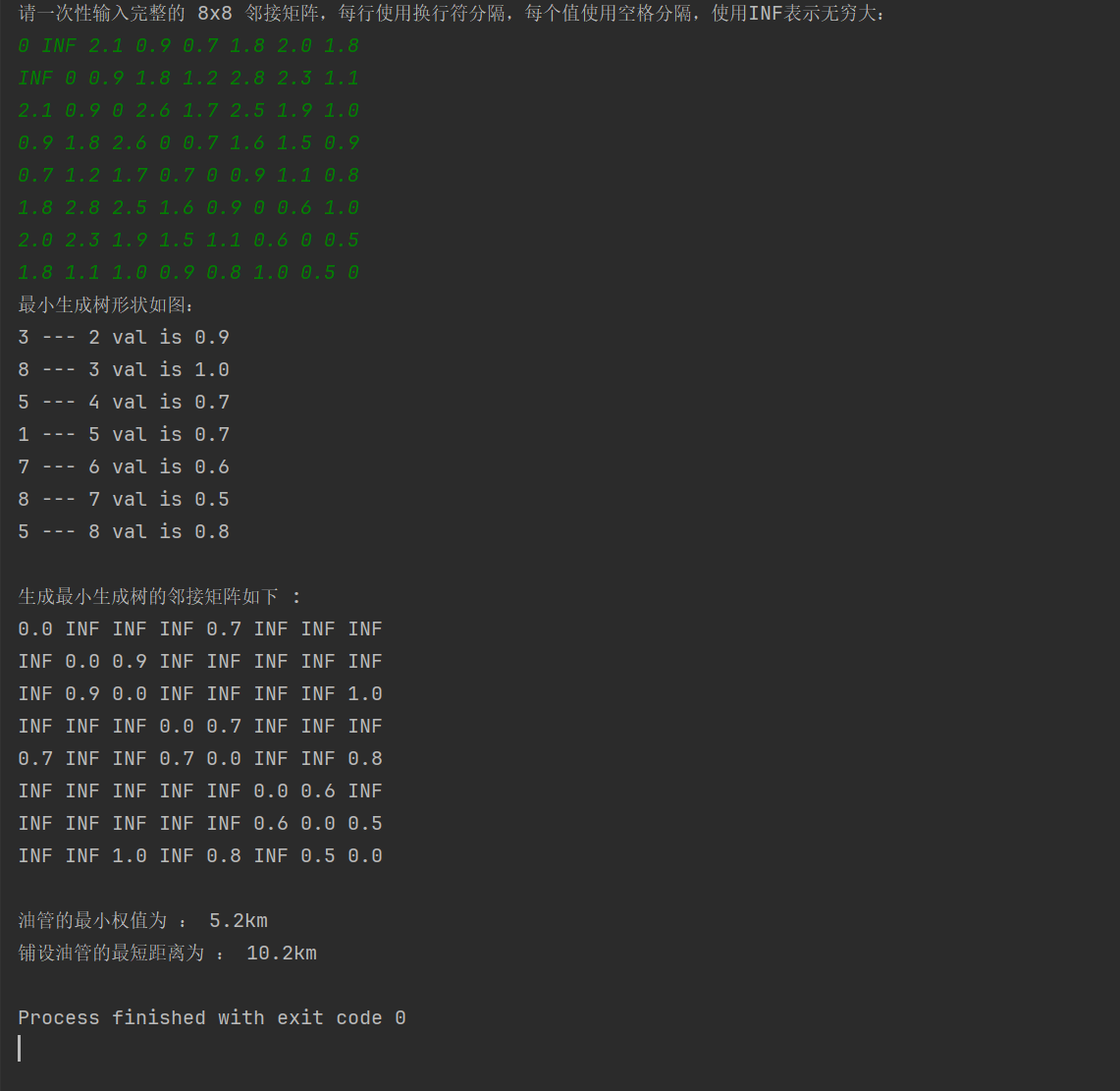
实验一测试数据:

程序运行输入内容:

0 INF 2.1 0.9 0.7 1.8 2.0 1.8  
INF 0 0.9 1.8 1.2 2.8 2.3 1.1  
2.1 0.9 0 2.6 1.7 2.5 1.9 1.0  
0.9 1.8 2.6 0 0.7 1.6 1.5 0.9  
0.7 1.2 1.7 0.7 0 0.9 1.1 0.8  
1.8 2.8 2.5 1.6 0.9 0 0.6 1.0  
2.0 2.3 1.9 1.5 1.1 0.6 0 0.5  
1.8 1.1 1.0 0.9 0.8 1.0 0.5 0

* 1. 运行结果

实验1运行结果:



* 1. 运行结果分析

1. 实验一中通过Prim算法计算出连接油管的最短方式，通过构造出包含所有顶点的最小生成树来计算出最短路径长度。同时利用edge列表存储每一个顶点的前驱节点并且记录当前节点的花费值。
2. 实验心得

通过这次实验，我学会了如何使用Prim算法解决实际问题，在编

写程序的过程中遇到了一些问题例如:

1. 编写Prim算法之初不知道如何计算出最小生成树的每个非父节点的前驱节点，之后通过学习查找资料利用edge列表进行存储，若无前驱节点则设置为-1反之设置为当前节点的前驱。
2. 计算最小生成树的每条边路径时，利用key列表对其进行存储操作初始时将开始的start节点为止key值设置为0。

总结来说，通过这次实验不仅了解了如何在实际生活中利用这些常见算法，并且学会了如何通过代码完成这些代码的编写与运行，掌握了简单的Debug方法对程序进行调试操作。