

通信与信息工程学院

嵌入式系统设计与应用

题 目 图最小生成树

专业及班级  电子科学与技术1901班

姓 名  吴栋

学 号 19207107004

日 期 2020年5月7日

目录

[1 最小生成树 3](#_Toc39777633)

[2 Prims算法 3](#_Toc39777634)

[3 Kruskal算法 4](#_Toc39777635)

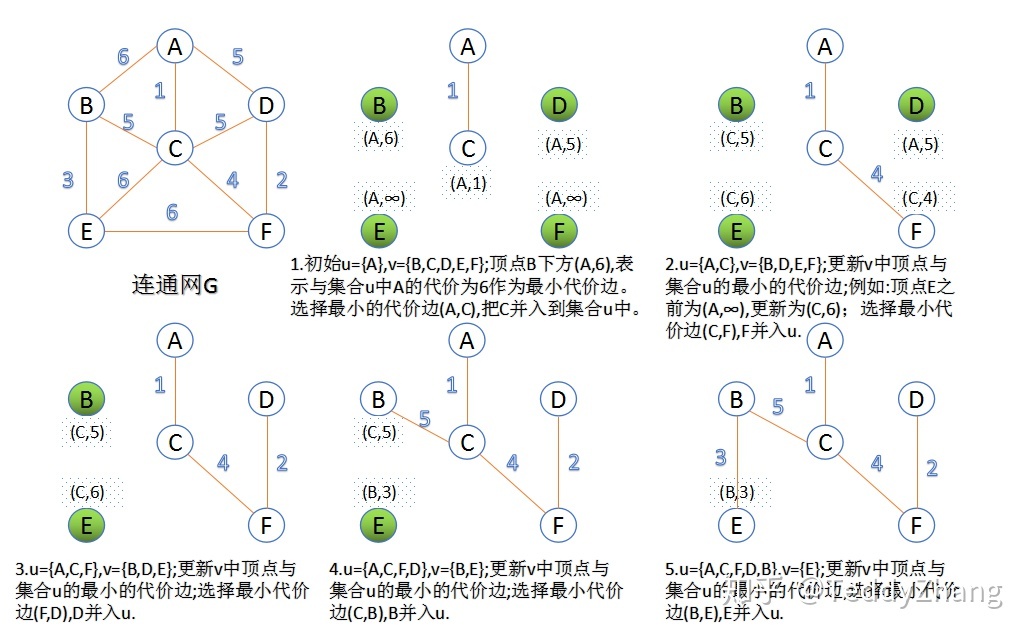
# 最小生成树

最小生成树（Minimum Spanning Tree，简称MST）也称为最小权重生成树（Minimum Weigth Spanning Tree）、最小支撑树是所有生成树中，总权值最小的那棵适用于有权的连通图。

# Prims算法

Prim算法是一种贪心算法，和Kuskral算法的贪心策略不同，Kuskral算法主要对边进行操作，而Prim算法则是对节点进行操作，每次遍历添加一个点，这时候我们就不需要使并查集了。

主要思想：首先选取图中任意一个顶点 v 作为生成树的根，之后继续往生成树中添加顶点 w，则在顶点 w 和顶点 v 之间必须有边，且该边上的权值应在所有和 v 相邻接的边中属最小。在一般情况下，假设图 G=(V,E) 中已落在生成树上的顶点集为U，则尚未落在生成树上的顶点集为 V-U，则从 (V-U) 顶点集中选取加入生成树的顶点 w 应满足下列条件：它和生成树上的顶点之间的边上的权值是在联接这两类顶点的所有边中权值属最小。



**图1 Prims算法示例**

具体步骤：

1、建立边set用来存放结果，建立节点set用来存放节点同时用于标记是否被访问过，建立边的最小堆。

2、开始遍历所有节点，如果没有访问，则添加到节点set，然后将其相连的边入堆。

3、从堆中取最小的边，然后判断to节点是否被访问过，如果没有，将这个边加入生成树（我们想要的边），并标记该节点访问。

4、然后将to节点所相连的边添加到最小堆中，不然这个网络就不会向外扩展了（这个步骤是必须的）。

5、循环上面的操作，直到所有的节点遍历完。

# Kruskal算法

Kruskal算法是一种贪心算法，我们将图中的每个edge按照权重大小进行排序，每次从边集中取出权重最小且两个顶点都不在同一个集合的边加入生成树中！如果这两个顶点都在同一集合内，说明已经通过其他边相连，因此如果将这个边添加到生成树中，那么就会形成环！这样反复做，直到所有的节点都连接成功！

主要思想：为使生成树上总的权值之和达到最小，则应使每一条边上的权值尽可能地小，自然应从权值最小的边选起，直至选出 n-1 条互不构成回路的权值最小边为止。具体作法如下：首先构造一个只含 n 个顶点的森林，然后依权值从小到大从连通网中选择不使森林中产生回路的边加入到森林中去，直至该森林变成一棵树为止，这棵树便是连通网的最小生成树。

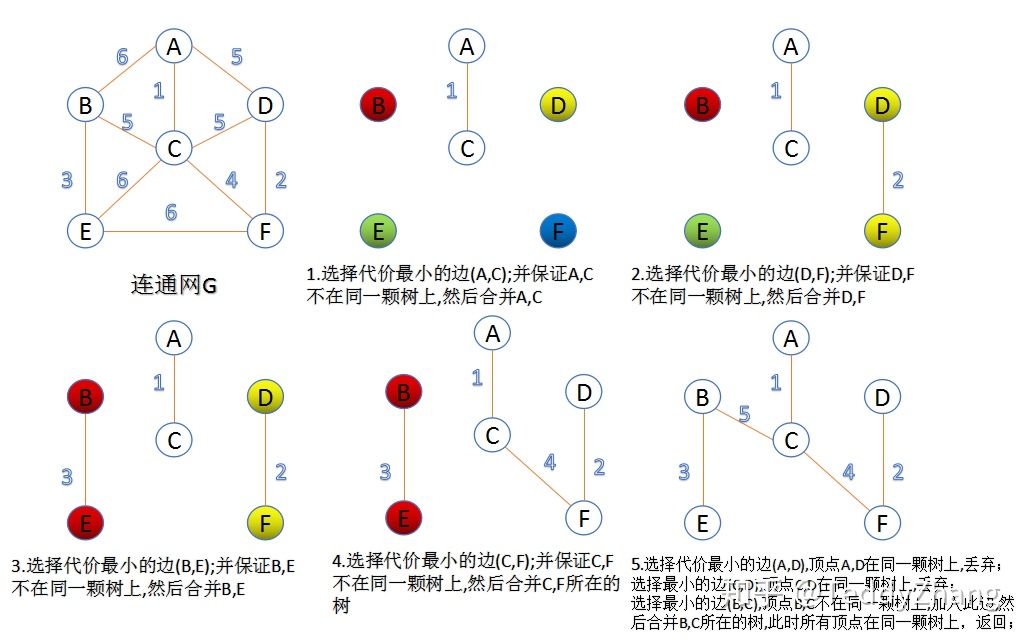
具体步骤：

1、对所有节点遍历建立并查集，按照边的权重建立最小堆

2、取出最小堆堆顶数据，并判断两端节点是否在同一集合

3、如不在，则将这两个节点添加到同一集合，接着将边加入生成边，如在，则不进行操作，为无效边

4、重复上面的操作，直到所有的边都检查完



**图2 Kruskal算法示例**