

云南大学数学与统计学院
《运筹学通论实验》上机实践报告

课程名称：运筹学实验	年级：2015 级	上机实践成绩：
指导教师：李建平	姓名：刘鹏	专业：信息与计算科学
上机实践名称：Prim 算法求图的支撑树与联通子图	学号：20151910042	上机实践日期：2018-06-13
上机实践编号：4	组号：	

一、实验目的

- 1. 学习 Prim 算法的使用；
- 2. 了解 Prim 算法作为贪心算法能达最优的理论证明。

二、实验内容

- 1. 写出 Prim（反圈法）算法^[1]的伪码描述^[2]；
- 2. 用 C 语言^[3]编程实现 Prim 算法，找出一幅图的最小生成树；
- 3. 写出 Prim 算法求解一个图的所有联通子图的算法；
- 4. 用 C 语言编程实现求一个图的所有联通子图的 Prim 算法程序。

三、实验平台

Microsoft Windows 10 Pro Workstation 1803;
Microsoft Visual Studio 2017 Enterprise。

四、算法设计

4.1 算法背景

普里姆算法（Prim 算法），图论中的一种算法，可在加权联通图里搜索最小生成树。意即由此算法搜索到的边子集所构成的树中，不但包括了连通图里的所有顶点，且其所有边的权值之和为最小。该算法于 1930 年由杰克数学家沃伊捷赫·亚尔尼克发现；并在 1957 年由美国计算机科学家罗伯特·普里姆独立发现；1959 年，艾兹格·迪科斯彻再次发现了该算法。因此，在某些场合，普里姆算法又被称为 DJP 算法、亚尔尼克算法或普里姆—亚尔尼克算法。Prim 算法的工作原理与 Dijkstra 的最短路径算法相似。本策略数域贪心策略，因为每一步所加入的边都必须是使得树的总权重增加量最小的边。

4.2 时间复杂度

这个算法的时间复杂度与图的实现方法有关。设图 $G = (V, E)$ ，其中 V 是图的所有节点的集合， E 是图的所有边的集合。图是由如果采用比较低级的邻接矩阵实现，那么 Prim 算法的时间复杂度是 $O(|V|^2)$ ；

如果用二叉堆、邻接表来实现，那么时间复杂度是 $O((|V| + |E|) \log |V|) = O(|E| \log |V|)$ ；如果用斐波那契堆来实现复杂度可以降低至 $O(|E| + |V| \log |V|)$

五、程序代码

5.1 程序描述

5.2 程序代码

程序代码 1

六、运行结果

代码分析

七、实验体会

八、参考文献

- [1] HILLIER F S, LIEBERMAN G J. 运筹学导论 [M]. 9th ed. 北京: 清华大学出版社, 2010.
- [2] CORMEN T H, LEISERSON C E, RIVEST R L, et al. 算法导论 [M]. 3rd ed. 北京: 机械工业出版社, 2013.
- [3] 林锐. 高质量 C++/C 编程指南 [M]. 1.0 ed., 2001.