

## 实验 2：图算法实现

内容：用自己喜欢的编程语言实现以下算法并求解问题。

1. 用 Prim 和 Kruska 算法分别求解给定数据集(mst\_data)中带权连通图上的最小生成树,要求输出从顶点 0 开始的一棵最小生成树的构建过程(节点顺序)和其所有边的权值之和。

a) 可以用 mst\_test 数据集作为测试集, 其 mst 权值之和为 14。

b) 数据集格式为:

[number\_of\_vertices] [number\_of\_edges]

[one\_endpoint\_of\_edge\_1] [other\_endpoint\_of\_edge\_1] [edge\_1\_cost]

[one\_endpoint\_of\_edge\_2] [other\_endpoint\_of\_edge\_2] [edge\_2\_cost]

2. 实现 Dijkstra 算法求解给定数据集(Dijkstra\_data)中带权有向图上的从节点 1 到节点 7,37,59,82,99,115,133,165,188,197 的最短路径长度。

a) 可以用 Dijkstra\_test 数据集作为测试集, 其中节点 1 到其他各个节点(1 到 8)的最短路径长度分别为: 0,1,2,3,4,4,3,2 。

b) Dijkstra\_data 数据集以邻接表的形式存储, 包含 200 个节点(标号为 1 到 200)。每行表示指定节点的边及其长度(非负)。例如, 第 6 行第一个数据项为“6”, 表示该行对应节点 6 的边。第 2 个数据项是“141,8200”, 表示节点 6 和节点 141 之间有一条长度为 8200 的边。该行剩下的其他项对应节点 6 其余的边。

3. 实现 Bellman-Ford 算法求解给定数据集(Allpairs\_data1, Allpairs\_data2, Allpairs\_data3)中带权有向图上的最短路径。注意: 边的权值可能为负, 图有可能含有负环。如果有负环的话请输出 “contains a negative cycle”。

a) 可以用 Allpairs\_test1 和 Allpairs\_test2 数据集作为测试集, 其输出分别为-2 和 “contains a negative cycle”。

b) 数据集格式同最小生成树数据集。

提交:

1. 源代码

2. 程序运行结果截图

3. 截止时间: 12 月 30 日 9pm