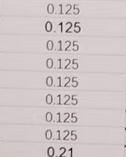
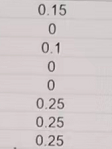
学生任务

1. 分组；
2. 选题（题目给出了讨论主题和一些简单思路，不限于给定思路）；
3. 小组协作完成任务，以小论文形式撰写讨论课报告（word文档）并在报告中总结收获； 撰写ppt，准备答辩，ppt中陈述每个人工作分工。
4. **组长**将讨论课报告、ppt、代码等相关资料提交到学在西电。
5. 组长评价每个成员对任务的贡献度，所有成员的贡献度和为1.

例如：每个组员贡献度都一样



例如：组员贡献度不一样，0代表没有贡献



1. 关于答辩：可以选择不参加答辩，参加答辩分数可能会有提升。 若参加答辩，要求所有小组成员参加答辩，参与主题讨论，答辩时间不超过8分钟。

截止时间：6月8日

关于贡献度：选完题后划分任务和DDL，所有按时完成的同学平分贡献度。

加油加油！！！

2023（春）讨论课题目

1. **中国邮递员问题研究及实现**

（1）给出中国邮递员问题的描述； （2）用离散数学的知识对其建模； （3）给出解决问题的算法并举例验证； （4）用到的定理（书上没讲到的）给出证明。

1. **最短路径算法研究及实现**

现有的最短路径算法有哪些？ 适用条件是什么？ 有哪些优缺点？

1. **图的连通性研究及实现**

给定任意图G，如何衡量图的连通度？ 图是否连通？ 如果不连通，如何给出所有的连通分图？ （将实现的操作用界面可视化展示）。

1. **二部图判定方法研究及实现**

判定一个图是不是二部图有哪些方法？ 用你提到的方法解决以下问题。 （自己设计3个不同的实例进行测试）

# 图是二分的吗？

有一个具有 n 个节点的无向图，其中每个节点的编号介于 0 和 n - 1 之间。你得到一个二维数组图，其中 graph[u] 是节点 u 相邻的节点数组。更正式地说，对于图[u]中的每个v，节点u和节点v之间都有一个无向边。该图具有以下属性：

没有自边（图[u]不包含u）。

没有并行的 edges（graph[u] 不包含重复值）。

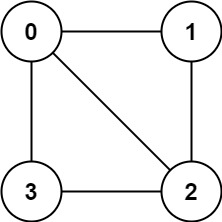
如果 v 在 graph[u] 中，则 u 在 graph[v] 中（图形是无向的）。

图形可能未连接，这意味着可能存在两个节点 u 和 v，因此它们之间没有路径。

如果节点可以划分为两个独立的集合 A 和 B，使得图形中的每个边连接集合 A 中的节点和集合 B 中的节点，则图形是二分的。

当且仅当它是二分的时返回 true。

示例 1：



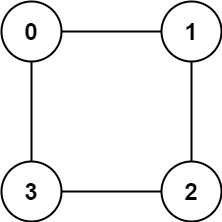
输入：图形 = [[1，2，3]，[0，2]，[0，1，3]，[0，2]]

注意：graph[0]=[1,2,3]，意思是：跟0相邻的结点有1,2,3，其他类似。

输出：假

说明：无法将节点划分为两个独立的集，以便每个边连接一个节点，另一个连接节点。

示例 2：



输入：图形 = [[1，3]，[0，2]，[1，3]，[0，2]]

输出：真

说明：我们可以将节点分为两组：{0， 2} 和 {1， 3}。

约束：

图形长度 == n

1 <= n <= 100

0 <= 图形[u].长度 < n

0 <= 图形[u][i] <= n - 1

图[u] 不包含 u。

图[u] 的所有值都是 唯一的。

如果 graph[u] 包含 v，则 graph[v] 包含 u。

1. **等价关系应用研究及实现**

什么是等价关系，它在哪些领域有应用，是怎样应用的，答辩时讲述2个典型的案例（论文里可以举多个例子）？

1. **平面图判定方法研究及实现**

什么是平面图？ 给定任意图G，如何判定它是否为平面图？

1. **旅行商问题研究及实现**

是否存在哈密尔顿回路问题。

1. **图的同构研究及实现**

什么是图的同构，如何判定两个图是否同构？

1. **偏序关系应用研究及实现。**

什么是偏序关系，它在哪些领域有应用，是怎样应用的，答辩时讲述2个典型的案例（论文里可以举多个例子）？

1. **最小生成树（MST）相关算法研究及实现**

求最小生成树（MST）的方法有哪些？ 用多种方法解决以下问题。

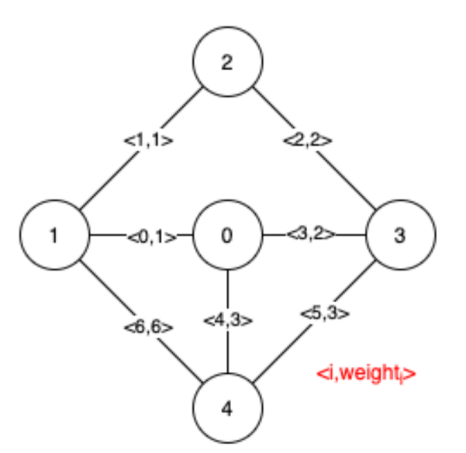
# 找到最小生成树 (MST)里的关键边和伪关键边

给定一个 包含n 个节点的带权无向连通图，节点编号为 0 到 n-1 ，还有一个数组 edges ，其中 edges[i] = [fromi, toi, weighti] 表示在 fromi 和 toi 节点之间有一条带权无向边，权值为 weighti。

请你找到给定图中最小生成树的所有关键边和伪关键边。 如果从图中删去某条边，会导致最小生成树的权值和增加，那么我们就说它是一条关键边。 伪关键边则是会出现在某些最小生成树中但不会出现在所有最小生成树中的边。

请注意，你可以分别以任意顺序返回关键边的下标和伪关键边的下标。

示例1

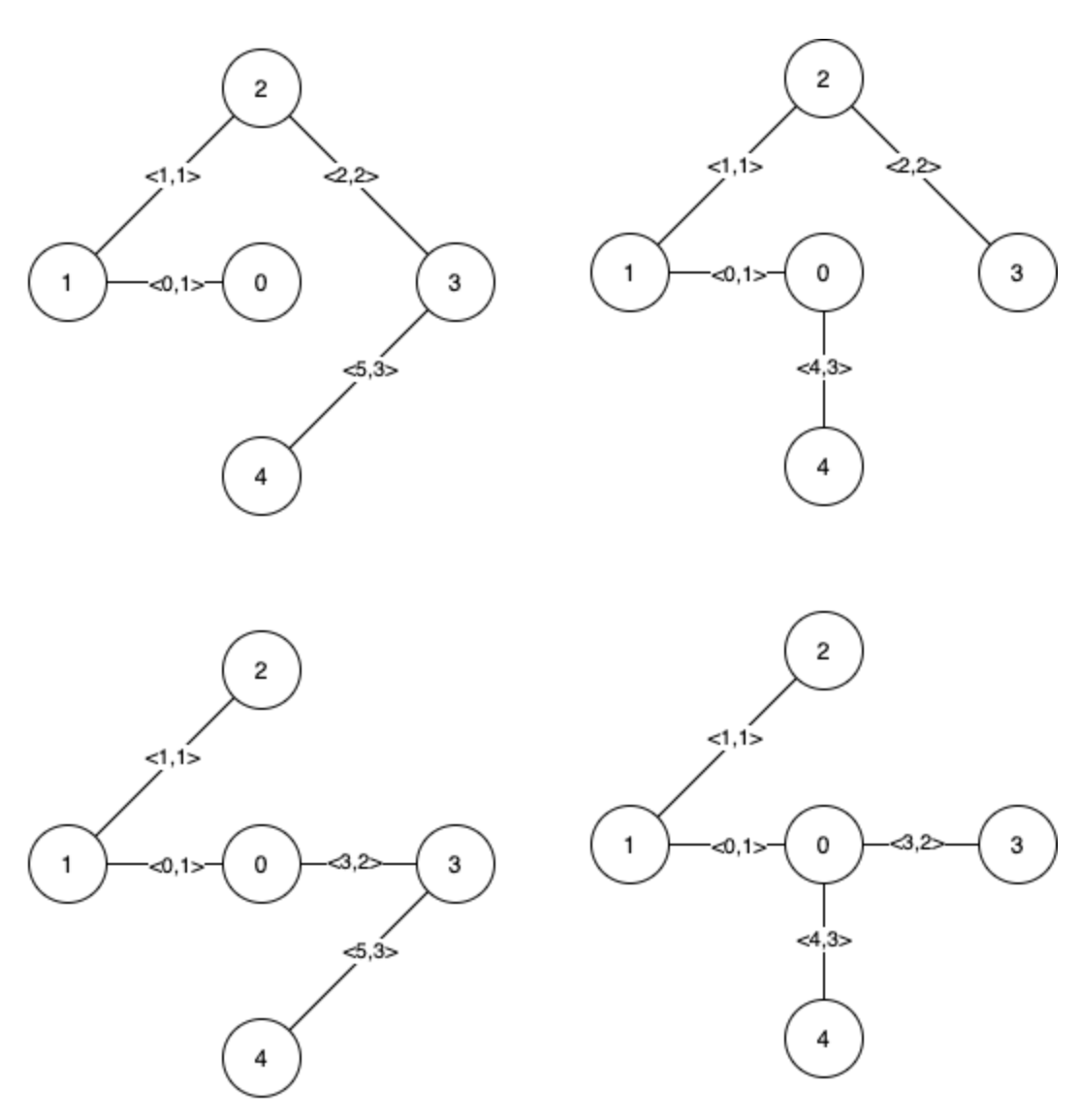


注意：<i,weithti>代表标号为i的边，权重为weithti 。

输入：n = 5, edges = [[0,1,1],[1,2,1],[2,3,2],[0,3,2],[0,4,3],[3,4,3],[1,4,6]]

输出：[[0,1],[2,3,4,5]]（[0,1]表示标号为0的边<0,1>和标号为1 的边<1,1>是关键边，[2,3,4,5]表示标号为2(<2,2>)， 3(<3,2>)，4(<4,3>)，5(<5,3>) 的边是伪关键边）

解释：上图描述了给定图。 下图是所有的最小生成树。



注意到从节点 0 到节点 1 这条边出现在了所有最小生成树中，所以它是关键边，将这两个下标作为输出的第一个列表。

边 2，3，4 和 5 是所有 MST 的剩余边，所以它们是 伪关键边。 我们将它们作为输出的第二个列表。

提示

2 <= n <= 100

1 <= 边.长度 <= min（200， n \* （n - 1） / 2）

边[i].长度 == 3

0 <= fromi < toi < n

1 <= 权重 <= 1000

所有 (fromi, toi) 数对都是互不相同的。