1. **问题描述**

旅行商从驻地出发，经过每个需要访问的城市一次且只有一次，并最终返回出发点。

如何安排路线，使旅行总路程最短？

1. **问题分析**

该问题使用了两种方法进行解决，回溯法与分支限界法。

回溯法为基于递归并剪枝的深度优先搜索；分支限界法为基于优先队列并进行下限判断的广度优先搜索

1. **输入输出格式**

输入：城市数、出发城市、城市间距离的邻接矩阵

输出：最小路程、访问城市的路线

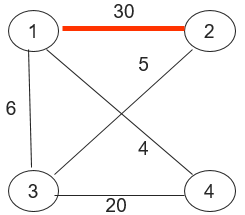
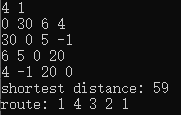
1. **回溯法**
2. 算法思想

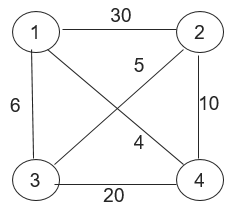
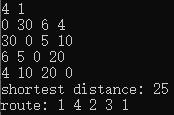
该方法在枚举法的基础上进行了两种剪枝操作：

如果即将前往的城市与当前所在城市不相连，则放弃该分支。

如果已经找到了一条回路，记录该回路的权值，在接下来的搜索中，如果权值已经大于记录值，则放弃该分支。

1. 测试情况

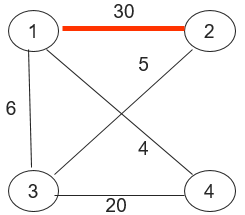
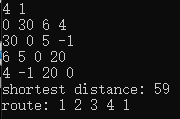
 

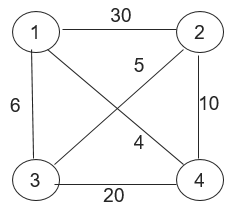
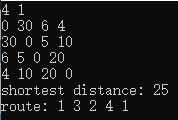
1. **分支限界法**
2. 算法思想

在算法的最开始，通过深度优先或贪心等方法取得一条环路作为上界，即在接下来的搜索中至少不应超过该上界。

此时使用一个优先队列进行”广度”优先遍历，优先队列中存储一个状态结点，其包括已走的路径、路径长度等信息，每次从优先队列中取出权值最小的结点，将其能到达的接下来的结点加入优先队列。当得到一次环路的结果后，更新上界。

1. 测试情况

1. **总结**

通过本次作业，我对回溯法与分支限界法有了更深入的认识。两种方法其实都是为了对枚举法进行剪枝，从而降低遍历的时空复杂度。相较而言回溯法的编写较为容易，但是深度优先搜索在数据量较大时容易由于无法及时更新上界而陷入极深的递归当中，相较而言广度优先遍历的分支限界法对于此情况较为友好，特别是ppt中所讲的电路布线问题，让我对分支限界法在解决某些问题当中的必要性有了全新的认识。

在以往，对于遍历解空间的问题我通常采用深度优先的递归进行求解，但通过这次作业将两者进行对比，也学习到了另一种遍历解空间的思路，拓宽了我解决问题的思路和方法，收获颇丰。

非常感谢老师的作业设计！