Problem A

创建一个名为tree的动态链表存入相应的结点，该列表大小每次改变近似于逐次加1；

在addEdge方法中采用contains方法遍历（相当于遍历两次）当前存的结点链表并执行相应操作；时间复杂度为O(n^2)

最后采用循环输出相应的父节点。时间复杂度为O(n^2)

因此总时间复杂度为O(n^2)

ProblemB

采用edge类进行边的存储。Edge类中存有边的总数。采用一个Deque的adjList根据当前数字存储其所有的边。

创建Edge Weighted Graph类将所有边加入，根据输入构建列表。当前时间复杂度为O(n)

在findPaths中将访问的结点，权值存入栈中，一次访问所有权值并计算每条路径的和，若和题目要求的num相等则将pathNumber加上1，最后输出总的相等数。此过程的时间的复杂度与边的具体构成情况密切相关，大致上复杂度为O(n^2)