# 可并堆之左偏树总结

**左偏树，顾名思义，是左边的结点权值较大的树形数据结构。主要用于两个优先队列的快速合并，是可并堆的一种实现方式。**

**定义与性质：**

1. **外结点：一个结点的右子结点为空就为外结点**
2. **距离：结点一直向右，直到外结点所经历的步数，每个结点距离等于右儿子的距离+1。**
3. **左偏树的父亲结点的优先级高于儿子结点**
4. **父亲结点的左子节点的距离大于等于右子节点的距离**

**合并操作：**

**大体流程：递归操作，将b结点与a结点的右孩子合并，同时会将b结点的左右孩子合并……以此类推，然后根据实际情况维护左孩子与右孩子的位置顺序**

**伪代码：**

* **merge(a,b)//a与b都是小顶堆**
* **{**
  + **If(a==null) return b;//a为空，根节点为b**
  + **If(b==null) return a;//b为空，根节点为a**
  + **If(key(a)>key(b)) swap(a,b);//将优先级高树的放在左边，左右子树调换位置**
  + **a.rchild=Merge(a.rchild,b);//a作为根，a的右孩子和b作为孩子递归**
  + **If(dis[a.rchild]>dis[a.lchild]) swap(a.lchild,a.rchild);//a的孩子节点排序**
  + **dis[a]=dis[a.rchild]+1;//a的距离更新**
  + **return a;//返回根节点**
* **}**

**插入操作：将新结点当作只有一个根的左偏树，merge合并。**

**删除操作：直接删除根结点，合并左右子树。**

## 1636: 猴王

时间限制: 1 Sec  内存限制: 128 MB  
提交: 32  解决: 6  
[[提交](http://fzoj.xndxfz.com/JudgeOnline/submitpage.php?id=1636)][[状态](http://fzoj.xndxfz.com/JudgeOnline/problemstatus.php?id=1636)][[讨论版](http://fzoj.xndxfz.com/JudgeOnline/bbs.php?pid=1636)]

## 题目描述

        很久很久以前，在一个广阔的森林里，住着n只好斗的猴子。起初，它们各干各的，互相之间也不了解。但是这并不能避免猴子们之间的争吵，当然，这只存在于两个陌生猴子之间。当两只猴子争论时，它们都会请自己最强壮的朋友来代表自己进行决斗。显然，决斗之后，这两只猴子以及它们的朋友就互相了解了，这些猴子之间将再也不会发生争论了，即使它们曾经发生过冲突。  
        假设每一只猴子都有一个强壮值，每次决斗后都会减少一半(比如10会变成5，5会变成2.5)。并且我们假设每只猴子都很了解自己。就是说，当它属于所有朋友中最强壮的一个时，它自己会站出来，走向决斗场。

## 输入

输入分为两部分。  
第一部分，第一行有一个整数n(n<=100000)，代表猴子总数。  
接下来的n行，每行一个数表示每只猴子的强壮值(小于等于32768)。  
第二部分，第一行有一个整数m(m<=100000)，表示有m次冲突会发生。  
接下来的m行，每行包含两个数x和y，代表第x个猴子和第y个猴子之间发生冲突。

## 输出

 输出每次决斗后在它们所有朋友中的最大强壮值。数据保证所有猴子决斗前彼此不认识。

## 样例输入

5

20

16

10

10

4

4

2 3

3 4

3 5

1 5

## 样例输出

8

5

5

10

**很经典的左偏树题，每次两只猴子打架之后将两只猴子所属的堆合并起来，并且将最强壮的猴子置于堆顶**