

第九周实验报告

沈家成

2017 年 11 月 14 日

1 1216 heap

1.1 find 操作

find 操作只要求 $O(n)$ 的时间复杂度，因此，只需要在储存二叉堆的顺序数组中遍历即可。

但是，find 操作的逻辑需要仔细考量。首先，要求是大于 x 的最小优先级，然后才是在相等的优先级里找下标最小的。这就涉及到最大值最小值，刚开始选择最后一个元素作为“擂主”，但是它不一定比 x 小，因此最后设置了一个布尔变量来找到第一个大于 x 的元素。因此，编写程序的时候，要注意容易被默认的前提条件。

1.2 decrease 操作

刚开始设想利用原有的函数，先删除那个元素，再插入优先级减少后的元素，但是这样会打乱整个二叉堆的结构。因此需要把优先级减少后，再调整二叉堆的结构。

借鉴插入堆的操作，将优先级减少后，和父节点作比较，如果比父节点小，就和父节点交换，否则就跳出。

2 1588 导弹拦截

2.1 单系统最大拦截数

因为每一发导弹都不高于前一发导弹，所以导弹的高度就组成了一个递减数列。因此单系统最大拦截数，就是求最大递减子数列。而在 1580 中已经解决了最大递增子数列的问题，替换比较运算符即可。

2.2 最少拦截系统数

第一发导弹发出后，只能击落比它高度低的导弹，这样，一旦有突然增大的高度，就需要另一套系统来拦截。因此，这就是求最大递增子数列的问题。因为，对于最大递增子数列中的每两个元素，在原数列中对应位置之间的元素，一定小于前一个元素和后一个元素，否则递增子数列中的元素就是那个反例了。

3 4012 合并果子

3.1 问题的本质

每种果子都是要至少合并一次的，节省体力就是要数目多的那种果子合并的次数少，把数目看做权值，就很容易联想到我们学习过的编码问题，也是同样的，要用尽量少的编码表示出现频率高的字符。因此，可以选用霍夫曼算法解决这个问题。

3.2 具体实现

这个问题可以借用霍夫曼算法的思想，但是最后只是需要输出最小体力值，并不需要得到编码，因此使用贪心法就可以了。

根据贪心法的思路，每次要找出最小和次小的元素，然后再把结果放回原来的数据中。容易想到用动态查找表保证次序，但其实只需要用最小化堆实现的优先级队列即可。

刚开始把元素都放入优先级队列。然后两次读队，就恰好得到了权值最小和次小的元素。权值相加，再插入优先级队列，就相当于完成了一次合并操作。每次操作的权值和就是消耗的体力值，累加，就得到了总体力值。