Heap

yanQval

IIIS, Tsinghua

2020 年 7 月 24 日

Heap

Heap

Definition

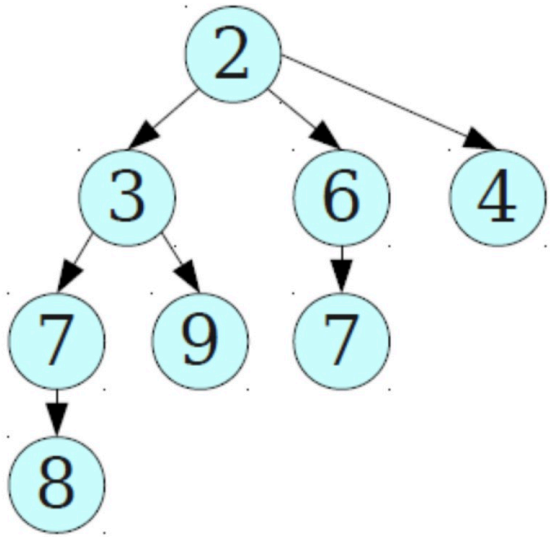
堆是一种特殊的树形数据结构，每个节点拥有一个键值 (Key)。
堆满足堆性质，若节点 A 是节点 B 的父亲，则 A 的键值比 B 的键值小(大)。
根据选择的不同被分别大根堆或小根堆。

Heap

Definition

堆是一种特殊的树形数据结构，每个节点拥有一个键值 (Key)。
堆满足堆性质，若节点 A 是节点 B 的父亲，则 A 的键值比 B 的键值小(大)。
根据选择的不同被分别大根堆或小根堆。

一个小根堆：



Operations

Operations

堆通常要支持的操作:

Operations

堆通常要支持的操作:

- 插入一个键值为 x 的新元素

Operations

堆通常要支持的操作:

- 插入一个键值为 x 的新元素
- 返回当前堆内的最小（大）值

Operations

堆通常要支持的操作:

- 插入一个键值为 x 的新元素
- 返回当前堆内的最小（大）值
- 返回并删除当前堆内的最小（大）值

Operations

堆通常要支持的操作:

- 插入一个键值为 x 的新元素
- 返回当前堆内的最小（大）值
- 返回并删除当前堆内的最小（大）值
- 将某一个元素的键值减小（增大）到 k

Operations

堆通常要支持的操作:

- 插入一个键值为 x 的新元素
- 返回当前堆内的最小（大）值
- 返回并删除当前堆内的最小（大）值
- 将某一个元素的键值减小（增大）到 k
- **将两个堆合并**

Operations

堆通常要支持的操作:

- 插入一个键值为 x 的新元素
- 返回当前堆内的最小（大）值
- 返回并删除当前堆内的最小（大）值
- 将某一个元素的键值减小（增大）到 k
- **将两个堆合并**
- 删除任意元素?

Binary Heap

Binary Heap

Definition

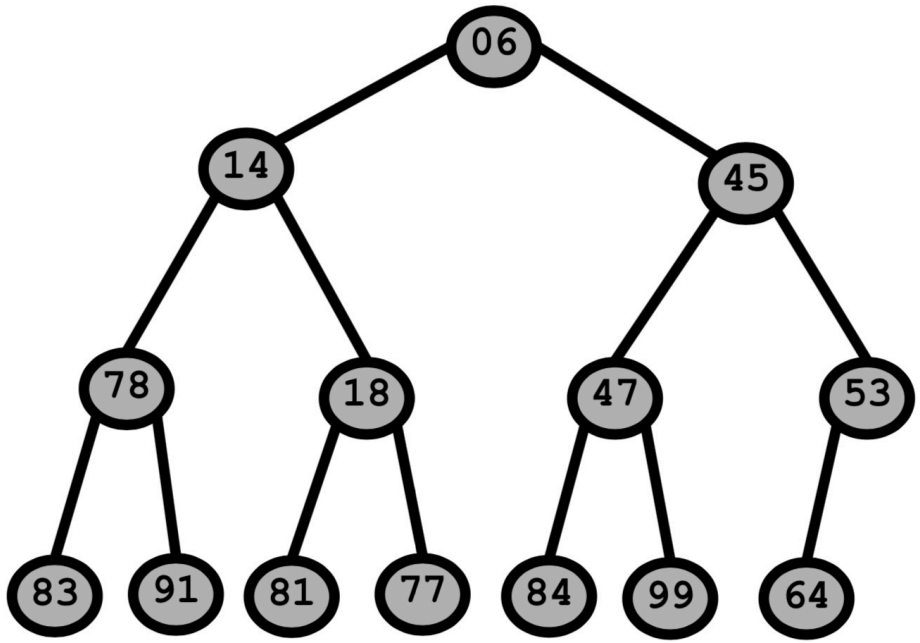
二叉堆是一个满足堆性质的完全二叉树。

Binary Heap

Definition

二叉堆是一个满足堆性质的完全二叉树。

一个二叉堆:



Binary Heap: Properties

Binary Heap: Properties

二叉堆拥有性质：

- 最小值出现在根上
- 深度为 $\lfloor \log_2 n \rfloor$

Binary Heap: Properties

二叉堆拥有性质：

- 最小值出现在根上
- 深度为 $\lfloor \log_2 n \rfloor$

这些优良的性质可以很方便的实现堆的功能。

Binary Heap: Implementation

Binary Heap: Implementation

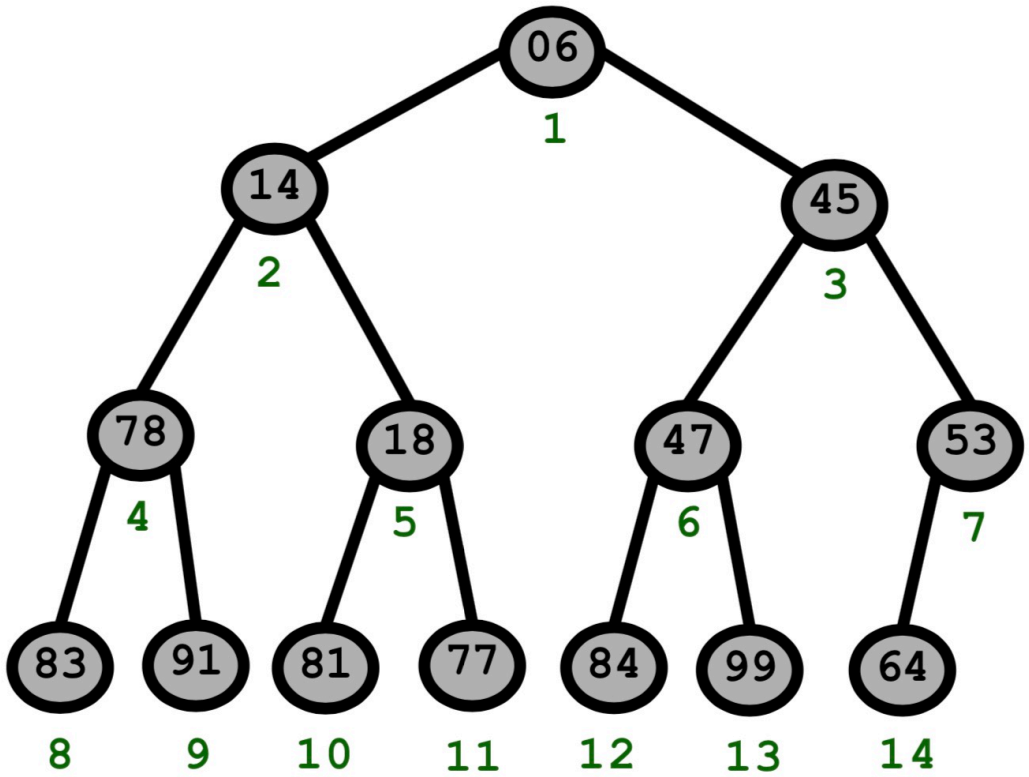
只需要数组就可以轻松实现二叉堆。

- Parent(i) : $\lfloor \frac{n}{2} \rfloor$
- Left Child(i): $2i$
- Right Child(i): $2i + 1$

Binary Heap: Implementation

只需要数组就可以轻松实现二叉堆。

- Parent(i) : $\lfloor \frac{n}{2} \rfloor$
- Left Child(i): $2i$
- Right Child(i): $2i + 1$



Binary Heap: Insertion

Binary Heap: Insertion

将新的元素放置在序列的末尾。

Binary Heap: Insertion

将新的元素放置在序列的末尾。
不断向上交换该元素，直到无法交换。

单次复杂度为 $O(\log_2 n)$。

Binary Heap: Decrease Key

与插入操作相似，不断向上进行交换直到无法交换。

单次复杂度为 $O(\log_2 n)$。

Binary Heap: Delete Min

Binary Heap: Delete Min

首先将根删除，并用序列末尾的元素代替根。

Binary Heap: Delete Min

首先将根删除，并用序列末尾的元素代替根。
不断执行向下交换，直到无法交换。

Binary Heap: Delete Min

首先将根删除，并用序列末尾的元素代替根。
不断执行向下交换，直到无法交换。
每次交换的时候，比较它的两个儿子，选择较小的一个进行交换。

单次复杂度为 $O(\log_2 n)$。

Binary Heap: Union

Binary Heap: Union

没有很好的合并方法。我们取出其中一个堆内的所有元素，逐个插入另外一个堆中。

单次复杂度至少为 $\Omega(n)$。

最小生成树

Prime 算法：使用堆解决。

Leftist Heap

Leftist Heap

Definition

左偏树是一棵特殊的树，每个节点拥有键值 (Key) 和距离 (Distance) 两个关键字。
该树关于键值满足堆性质，关于距离满足如下的性质。
定义一个节点为外节点，当且仅当它存在一个孩子是空的。
定义一个节点的距离为它到外节点的最短距离。
任意节点左儿子的距离不小于右儿子的距离。

Leftist Heap: Properties

Leftist Heap: Properties

性质一:

节点的距离等于其右儿子的距离加一

Leftist Heap: Properties

性质一:

节点的距离等于其右儿子的距离加一

性质二:

距离为 i 的左偏树至少有 $2^i - 1$ 个节点。

Leftist Heap: Union

Leftist Heap: Union

合并是一个递归过程。

Leftist Heap: Union

合并是一个递归过程。
如果当前合并中有一棵空树，直接返回另一颗树。

Leftist Heap: Union

合并是一个递归过程。
如果当前合并中有一棵空树，直接返回另一颗树。
选择两棵树中根节点键值较小的一个作为根，继续合并该节点的右儿子和另一棵树。

Leftist Heap: Union

合并是一个递归过程。
如果当前合并中有一棵空树，直接返回另一颗树。
选择两棵树中根节点键值较小的一个作为根，继续合并该节点的右儿子和另一棵树。
如果当前树不满足距离性质，交换左右孩子。

由于我们每次走的都是右孩子，执行的次数不超过两棵树的深度之和。
因此单次操作的复杂度是 $O(\log_2 n)$.

Leftist Heap: Other Operations

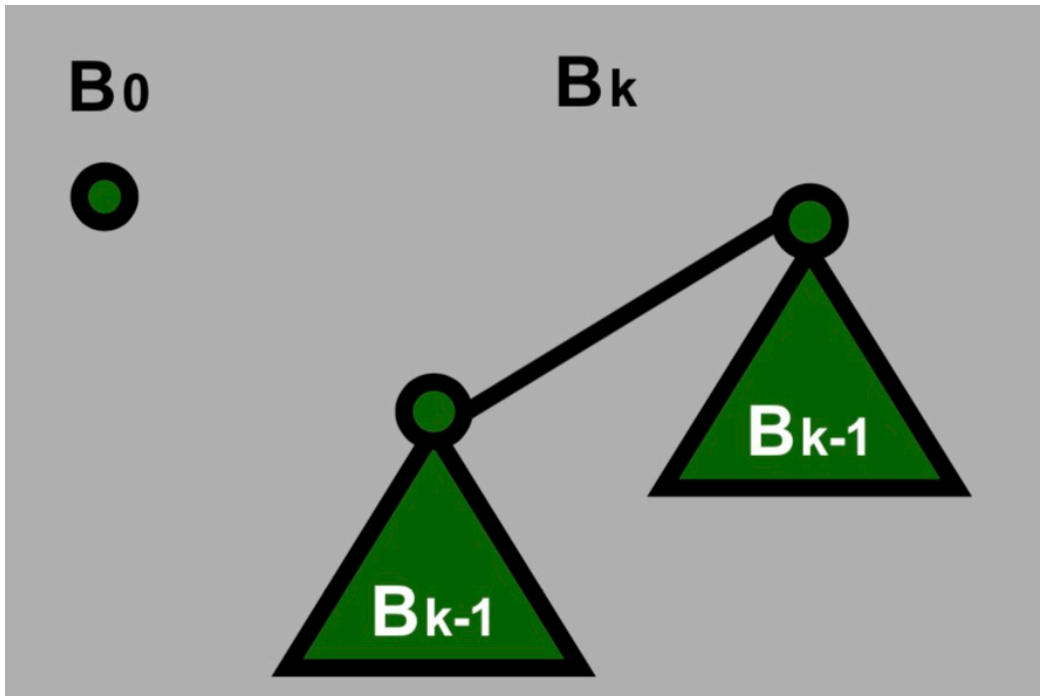
Leftist Heap: Other Operations

其他的所有操作均可以通过合并操作来实现：

- 插入操作：合并原堆和一个节点的堆
- 删除最小值：合并左右孩子

Binomial Tree

二项树是如下递归定义的：



Rank

Rank

我们定义一个节点的秩为它的孩子的个数，其中一棵树的秩是它的根的秩。

Rank

我们定义一个节点的秩为它的孩子的个数，其中一棵树的秩是它的根的秩。
它的孩子的秩如何?

Rank

我们定义一个节点的秩为它的孩子的个数，其中一棵树的秩是它的根的秩。
它的孩子的秩如何?
B_k 有多少节点?

Properties

Properties

关于二项树 B_k 有一些重要性质：

Properties

关于二项树 B_k 有一些重要性质:

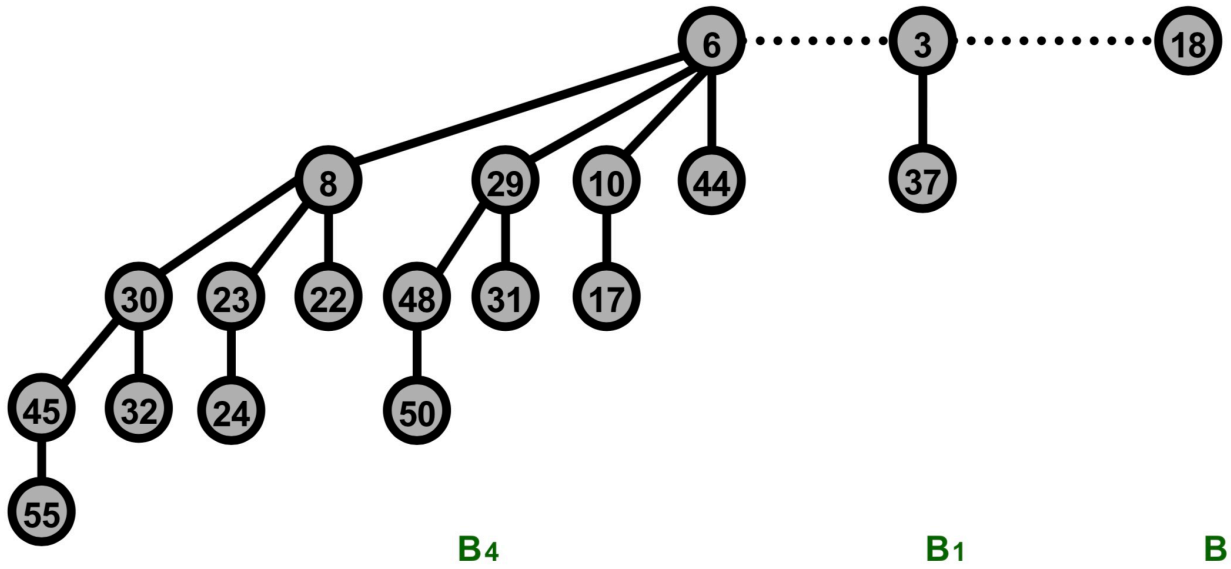
- 节点个数 2^k
- 深度 k
- 度数 k
- 若我们删除根节点，将得到 $B_{k-1}, B_{k-2}, \ldots, B_0$

B_k 的第 i 层有多少节点?

Binomial Heap

Definition

一个二项树组成的序列，其中每种秩只存在一棵树等于它。



Properties

Properties

最小值一定保存在所有的根上。

Properties

最小值一定保存在所有的根上。
B_i 是否存在由 n 的二进制分解决定。

Properties

最小值一定保存在所有的根上。
B_i 是否存在由 n 的二进制分解决定。
至多存在 $\log_2 n$ 棵树，最大秩同样不超过这个值。

Union

Union

只合并具有相同秩的树，并保证最终的堆中不存在秩相同的树。

Union

只合并具有相同秩的树，并保证最终的堆中不存在秩相同的树。

合并两棵树需要的复杂度是 $O(1)$，总共合并的次数和二进制加法是一样的，总复杂度 $O(\log_2 n)$。

Other Operations

Other Operations

插入操作同样可以通过合并来实现。

Other Operations

插入操作同样可以通过合并来实现。
删除操作根据之前的结论会将某棵树分解成若干颗小的树，继续执行合并操作。

Other Operations

插入操作同样可以通过合并来实现。
删除操作根据之前的结论会将某棵树分解成若干颗小的树，继续执行合并操作。
减小键值可以直接向上交换，由于二项堆只有 $O(log_2 n)$ 的深度。

Fibonacci Heap

只有删除操作具有 $O(\log_2 n)$ 的单次复杂度，其他操作均只消耗常数时间。
运用了延迟修改的思想。

合并果子

有 n 堆果子，第 i 堆有 a_i 个果子，现在需要把所有果子合成一堆，每次合并消耗的体能是合并之后的那一堆的大小，最小化体能消耗。
$n \leq 10^5$

哈夫曼编码

有 n 个单次组成的文章，第 i 个单词出现了 a_i 次，现在需要给所有单词用 01 来编码，为保证不会出现歧义，不能有任一编码是另外一个的前缀。求最短传输长度。

$n \leq 10^5$

[NOI2015] 荷马史诗

用 k 进制来编码。且要求在总长度最短的情况下，最小化最长编码的长度。

罗马游戏

给定 n 个人的分数，若干次操作，每次将两个人所在的团合并，或是杀死一个团内得分最低的人。
$n \leq 10^6$

[BOI 2004] 数字序列

给定一个长度为 n 的序列 a_i，求出一个单调上升的序列 b_i，最小化 $\sum |a_i - b_i|$。
$n \leq 10^5$

堆

给定数轴上的 n 个区间，请选择其中恰好 k 个区间，使得它们的交的长度最大。
$n \leq 10^6$.

搬东西

有 n 堆东西，第 i 堆初始有 A_i 个东西，我们希望最终第 i 堆有 B_i 个东西。现在可以进行的操作有，给某一堆直接添加一个东西，代价是 X，直接删除一个代价是 Y，还可以花费 $Z * |i - j|$ 的代价，把一个东西从 i 搬到 j。求最小代价。

$1 \leq n \leq 10^5, 1 \leq A_i, B_i \leq 10, 0 \leq X, Y \leq 10^8, 0 \leq Z \leq 1000$