

树状数组简单介绍

问题引入

- 已知一个数组 $a[]$, 要求完成一下的操作:
 - 1、第 i 个元素add m ;
 - 2、区间 $【x, y】$ 内的每个元素add m ;
 - 3、查询区间 $【x, y】$ 内所有元素的sum;
 -
 -
 -

基于以上问题设计数据结构

- 设计一：基于简单数据结构a[]
复杂度分析：
 $O(1)$, $O(n)$, $O(n)$
- 设计二：基于简单数据结构sum[]
复杂度分析：
 $O(n)$, $O(n)$, $O(1)$

树状数组简单介绍

- 建立 $c[]$ ，其中 $C[i]=a[i-2^k+1]+.....+a[i]$ (k 为 i 在二进制形式下末尾0的个数)。

- 例如：

$$c[1]=a[1]$$

$$c[2]=a[1]+a[2]=c[1]+a[2]$$

$$c[3]=a[3]$$

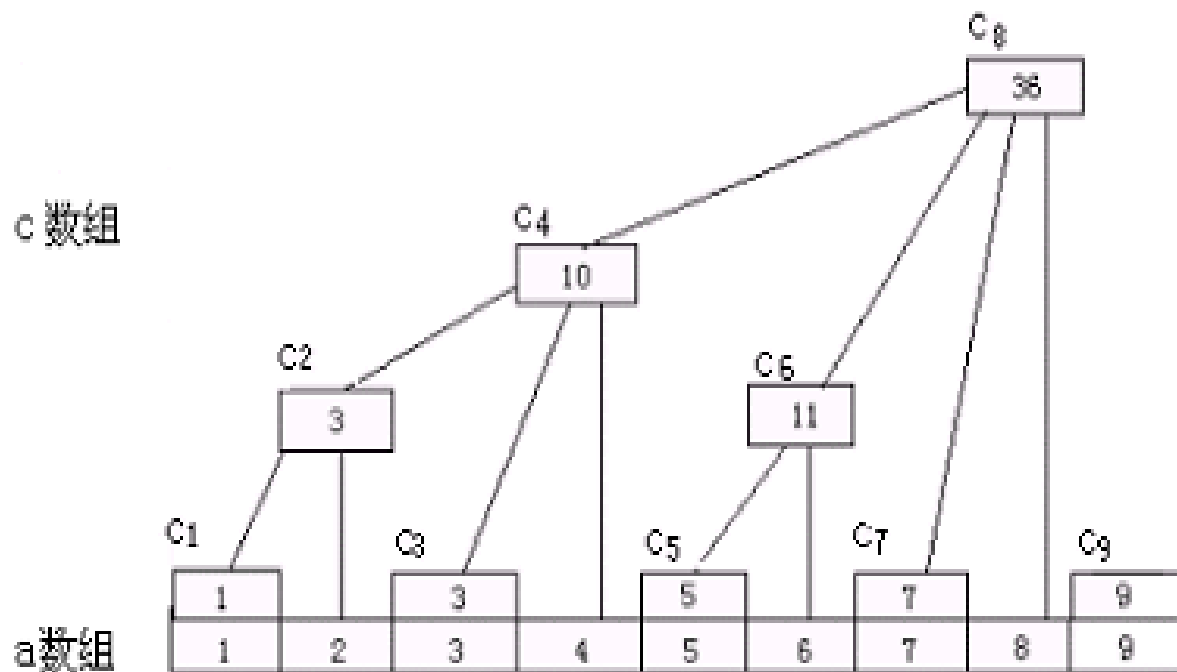
$$c[4]=a[1]+a[2]+a[3]+a[4]=c[2]+c[3]+a[4]$$

$$c[5]=a[5]$$

$$c[6]=a[5]+a[6]=c[5]+a[6]$$

.....

树状数组的简单结构



问题分析

- 信息更新复杂度 $O(\log n)$ ，求和查询复杂度 $O(\log n)$ 。
- ? ? ?
- 简单示例：
 $a[] = \{1, 1, 1, 1, 1, 1, 1, 1, 1, 1, \dots\}$
 $i[] = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, \dots\}$
 $c[] = \{1, 2, 1, 4, 1, 2, 1, 8, 1, 2, \dots\}$
对于一个数 n ，统计前 n 项和有什么规律呢？

求和规律分析

- 将整数 n 表示为二进制形式

如: 5 101 $\text{sum}[5] = c[5] + c[4]$

6 110 $\text{sum}[6] = c[6] + c[4]$

7 111 $\text{sum}[7] = c[7] + c[6] + c[4]$

8 1000 $\text{sum}[8] = c[8]$

10 1010 $\text{sum}[10] = c[10] + c[8]$

.....

规律: n 每次减去最低位的1, 减到0为止

结论: 求和复杂度 $O(\log n)$

更新信息规律分析

- 同样整数 n 表示为二进制形式

- 如：

5	101	$c[5], c[6], c[8], \dots$
6	110	$c[6], c[8], c[16], \dots$
7	111	$c[7], c[8], c[16], \dots$

.....

规律： n 每次增加最低位的1，增到最大为止

结论：更新复杂度 $O(\log n)$

树状数组基本操作

- 计算机利用补码表示整数。

$$\text{lowbit}(x) = x \ \& \ (-x);$$

可以利用`lowbit`来动态维护和查询相关信息。

树状数组例题分析

- 例1(hoj 2275 Number sequence)

题目描述：统计序列中 $A_i < A_j > A_k$ ($i < j < k$) 的个数。

题目分析：

朴素思想：对于每个元素，分别查询之前小于它的元素的个数和查询之后小于他的元素的个数，然后相乘。最后把所有的相加即可。

这样的复杂度是 $O(n^2)$ 。

树状数组例题分析

- 问题关键，如何统计在 a_k 之前小于 a_k 的元素个数，同理也可以解决之后小于的 a_k 统计问题
- 可以有效利用树状数组动态维护更新信息。
- 对于序列 1 3 4 2 5 1
- 树状数组c对应的元素a数组：

游标 $i=1$ {0, 0, 0, 0, 0, 0, 0, 0, 0}

$\text{getsum}(1-1)=0$

游标 $i=2$ {1, 0, 0, 0, 0, 0, 0, 0, 0}

$\text{getsum}(3-1)=1$

游标 $i=3$ {1, 0, 1, 0, 0, 0, 0, 0, 0}

$\text{getsum}(4-1)=2$

游标 $i=4$ {1, 0, 1, 1, 0, 0, 0, 0, 0}

$\text{getsum}(2-1)=1$

游标 $i=5$ {1, 1, 1, 1, 0, 0, 0, 0, 0}

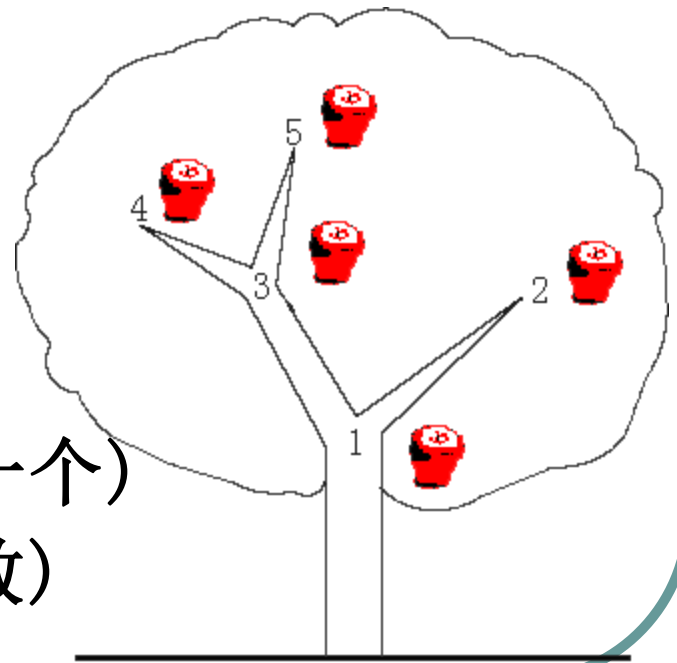
$\text{getsum}(5-1)=4$

游标 $i=6$ {1, 1, 1, 1, 1, 0, 0, 0, 0}

$\text{getsum}(1-1)=0$

树状数组例题分析

- 例2(po j 3321 Apple Tree)
- 题目描述:
- 有一棵苹果树，苹果树有N个叉，每个叉都可能有苹果，每个叉都有编号，root的编号为1。
kaka有如下需求:
C m(改变编号为叉m的信息，如果有苹果，kaka摘掉；如果没苹果，又会长出一个)
Q m(查询叉m子树的苹果个数)



树状数组例题分析

- 问题分析：

由于树型结构的特殊性，叉 m 的统计信息可以遍历子树得到，这要的复杂度为 $O(n)$ ，如果记录区间和，更新信息的复杂度为 $O(n)$ 。

很自然想到树状数组，关键如何确定给叉编号，如何确定查询区间。

由于树型结构的特殊性，子结点的子树一定在父节点子树的范围之内。

树状数组例题分析

- 解决方案:

- 1、DFS遍历“苹果树”，记录每个叉的遍历起始时间和遍历结束时间，这样就得到查询区间。这样可以将“苹果树”映射到一个树状数组上。

- 2、可以用一个flag[]标记该叉是否有苹果，然后动态更新“苹果树”（可以统一规定在起始时间上更新）。

- 3、查询叉m对应区间的苹果个数即可。如：区间【a, b】为 $\text{getsum}(b) - \text{getsum}(a-1)$

二维树状数组的简单介绍

- 对于一个二维数组 $a[][]$ 以及其对应的二维 $c[][]$

$\{1, 1, 1, 1\}$

$\{1, 2, 1, 4\}$

$\{1, 1, 1, 1\}$

$\{2, 4, 2, 8\}$

$\{1, 1, 1, 1\}$

$\{1, 2, 1, 4\}$

$\{1, 1, 1, 1\}$

$\{4, 8, 4, 16\}$

思想很简单，具体操作见所附代码。

树状数组例题分析

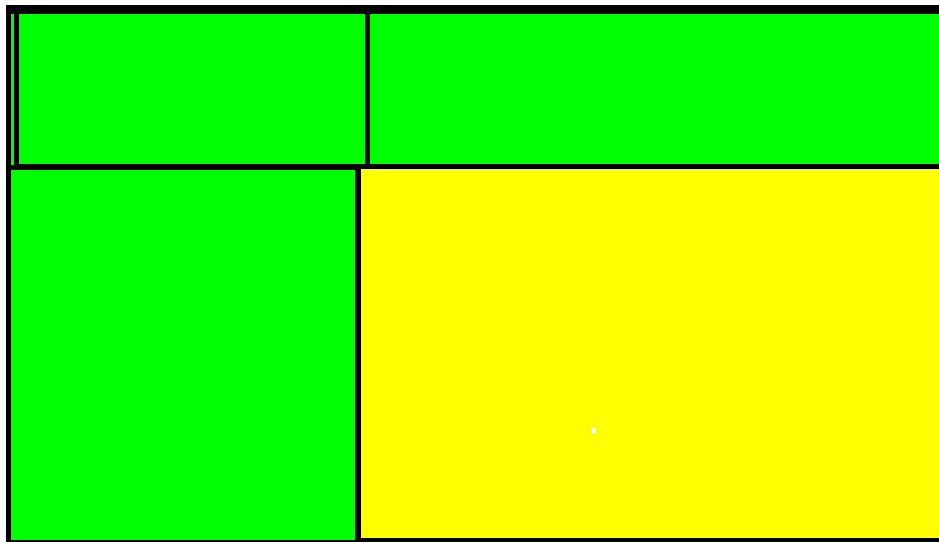
- 例3(hoj 1640 Mobile Phone)
- 具体操作如下:

Instruction	Parameters	Meaning
0	S	Initialize the matrix size to $S \times S$ containing all zeros. This instruction is given only once and it will be the first instruction.
1	$X Y A$	Add A to the number of active phones in table square (X, Y) . A may be positive or negative.
2	$L B R T$	Query the current sum of numbers of active mobile phones in squares (X, Y) , where $L \leq X \leq R, B \leq Y \leq T$
3		Terminate program. This instruction is given only once and it will be the last instruction.

- 该题为二维树状数组的基本操作，套模板即可。

树状数组例题分析

- 二维区间查数：如图



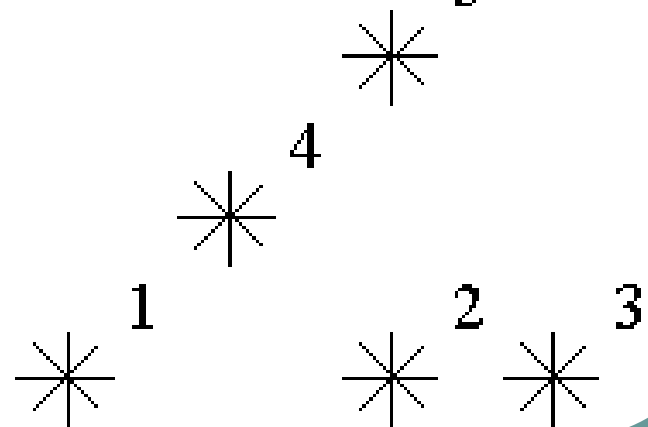
树状数组例题分析

● 例4(po j 2352 Stars)

天空中有一些星星，这些星星都在不同的位置，每个星星有个坐标。如果一个星星的左下方（包含正左和正下）有k颗星星，就说这颗星星是k级的。比如，在下面的例图中，星星5是3级的（1，2，4在它左下）。

星星2，4是1级的。例图中有1个0级，2个1级，1个2级，1个3级的星。

求出各个级别的星星的个数



树状数组例题分析

- 题目分析：
- 用二维树状数组直接解题，空间上无法接受！
☹
- 解决方案：
- 可以按行扫描，统计，掷点更新！！！时间和空间都是可以接收的！