原创 位运算+树状数组

2019-05-21 10:17:59 _-Y-_-Y-_ 阅读数 35 文章标签: ACM 更多

按位求与 (&)

0&0=0
0&1=0
1&0=0
1&1=1

按位求或 (|)

0|0=0
0|1=1
1|0=1
1|1=1

按位取反 (~)

~0=1
~1=0

按位异或 (^)

0^0=0
1^0=1
0^1=1
1^1=0

优先级

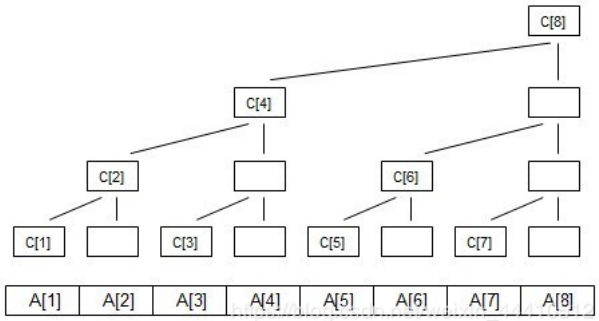
优先级	运算符	结合性
1	() [] .	从左到右
2	! +(正) -(负) ~ ++ --	从右向左
3	* / %	从左向右
4	+(加) -(减)	从左向右
5	<< >> >>>	从左向右
6	< <= > >= instanceof	从左向右
7	== !=	从左向右
8	&(按位与)	从左向右
9	^	从左向右
10	|	从左向右
11	&&	从左向右
12	||	从左向右
13	?:	从右向左
14	= += -= *= /= %= &= |= ^= ~= <<= >>= >>>=	从右向左

以下内容来自https://blog.csdn.net/Small_Orange_glory/article/details/81290634

树状数组

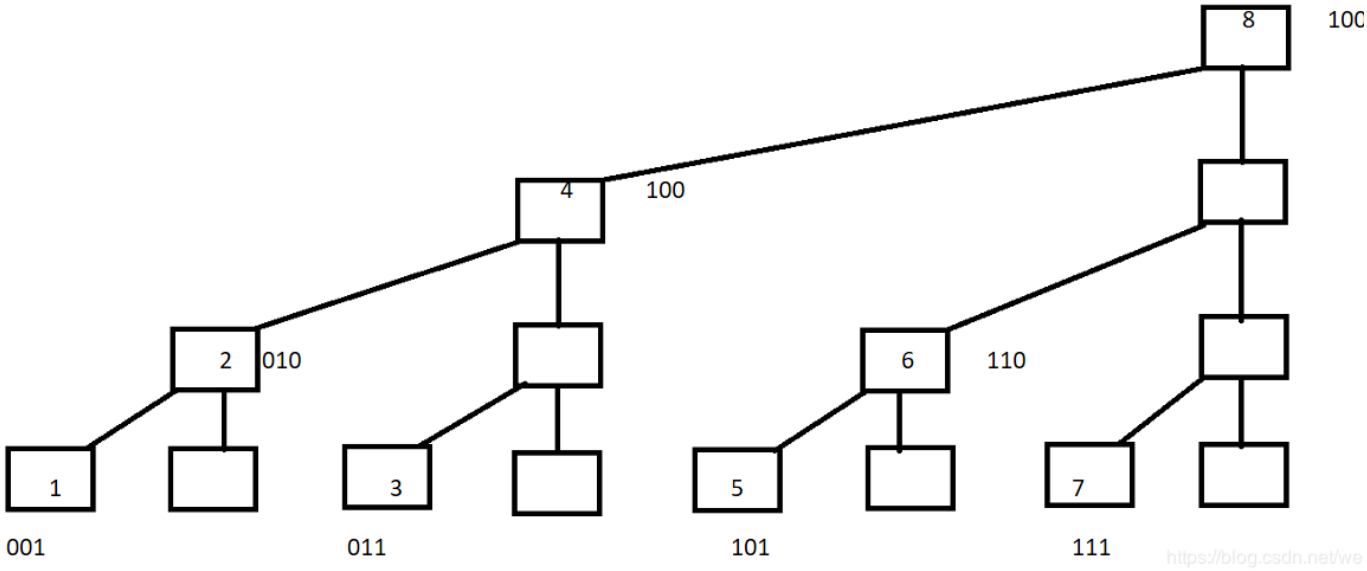
树状数组核心就是

```
int lowbit(int t){
    return t&(-t);
}
```



C[i]代表 子树的叶子结点的权值之和// 这里以求和举例
如图可以知道

$$C[1] = A[1];$$
$$C[2] = A[1] + A[2];$$
$$C[3] = A[3];$$
$$C[4] = A[1] + A[2] + A[3] + A[4];$$
$$C[5] = A[5];$$
$$C[6] = A[5] + A[6];$$
$$C[7] = A[7];$$
$$C[8] = A[1] + A[2] + A[3] + A[4] + A[5] + A[6] + A[7] + A[8];$$



将C[]数组的结点序号转化为二进制

$1 = (001)\ \ C[1] = A[1];$
$2 = (010)\ \ C[2] = A[1] + A[2];$
$3 = (011)\ \ C[3] = A[3];$
$4 = (100)\ \ C[4] = A[1] + A[2] + A[3] + A[4];$
$5 = (101)\ \ C[5] = A[5];$
$6 = (110)\ \ C[6] = A[5] + A[6];$
$7 = (111)\ \ C[7] = A[7];$
$8 = (1000)\, C[8] = A[1] + A[2] + A[3] + A[4] + A[5] + A[6] + A[7] + A[8];$

向上建树

```
void update(int x,int val){
    while(x<=n){
        t[x]+=val;
        x+=x&(-x);
    }
}
```

例如二进制数 0 0 0 1向上建树
原码 0 0 0 1 （1）
反码 1 1 1 0
补码 1 1 1 1
x&(-x) 0 0 0 1

原码 0 0 1 0 （2）
反码 1 1 0 1
补码 1 1 1 0
x&(-x) 0 0 1 0

原码 0 1 0 0 （4）
反码 1 0 1 1
补码 1 1 0 0
x&(-x) 0 1 0 0

例如 二进制 0 0 1 1

原码 0 0 1 1 （3）
反码 1 1 0 0
补码 1 1 0 1
x&(-x) 0 0 0 1

原码 0 1 0 0 （4）
反码 1 0 1 1
补码 1 1 0 0
x&(-x) 0 1 0 0

原码 1 0 0 0 （8）
反码 0 1 1 1
补码 10 0 0
x&(-x) 1 0 0 0

例如 二进制 0 1 0 1

原码 0 1 0 1（5）
反码 1 0 1 0
补码 1 0 1 1
x&(-x) 0 0 0 1

原码 0 1 1 0（6）
反码 1 0 0 1
补码 1 0 1 0
x&(-x) 0 0 1 0

区间查找（1~x）

```
int sum(int x){
    int ans=0;
    while(x>=1){
        ans+=t[x];
        x-=x&(-x);
    }
    return ans;
}
```

例一

https://cn.vjudge.net/problem/POJ-2299

AC code

```
#include <iostream>
#include <algorithm>
#include <cstring>
#include <cstdio>
using namespace std;
```

```cpp
#define maxn 500010
int n;
int a[maxn];
int t[maxn*4];
struct node{
    int x;
    int id;
}f[maxn];
bool cmp(node a, node b){
    return a.x<b.x;
}
void update(int x,int val){
    while(x<=n){
        t[x]+=val;
        x+=x&(-x);
    }
}
int sum(int x){
    int ans=0;
    while(x>=1){
        ans+=t[x];
        x-=x&(-x);
    }
    return ans;
}
int main(){
    while(~scanf("%d", &n),n){
        for(int i=1;i<=n;i++){
            scanf("%d", &f[i].x);
            f[i].id=i;
        }
        sort(f+1,f+n+1,cmp);
        for(int i=1;i<=n;i++) a[i]=f[i].id;
        memset(t,0,sizeof(t));
        long long ans=0;
        for(int i=1;i<=n;i++){
            update(a[i],1);
            ans+=(i-sum(a[i]));
        }
        printf("%lld\n", ans);
    }
    return 0;
}
```