题目给的l和r，有可能是l>r的，要判断交换！！！！

就是一个线段树，但是是开根号，没有办法用延迟更新，但是取根号在6，7次就会到1.

那么我们模拟区间更新，如果父节点的权值和r-l+1相等，那么就不用忘下更新了。因为都是1

#include <cstdio>

#include<iostream>

#include<vector>

#include<algorithm>

#include<string>

#include<cstring>

#include<set>

#include<map>

using namespace std;

#define ll long long

#define INF 1000000007

ll a[100010\*4];

inline void build(int root,int l,int r)

{

if(l==r)

{

scanf("%lld",&a[root]);

return ;

}

int mid=(l+r)>>1;

build(root<<1,l,mid);

build(root<<1|1,mid+1,r);

a[root]=a[root<<1]+a[root<<1|1];

}

inline void update(int x,int y,int root,int l,int r)

{

if(x<=l && r<=y && a[root]==(ll)(r-l+1))

return;

if(l==r)

{

a[root]=(ll)floor(sqrt(1.0\*a[root]));

return;

}

int mid=(l+r)>>1;

if(y<=mid)//要找的区间整个在mid左侧

update(x,y,root<<1,l,mid);

else if(x>mid)//要找的区间整个在mid右侧

update(x,y,root<<1|1,mid+1,r);

else

{

update(x,mid,root<<1,l,mid);

update(mid+1,y,root<<1|1,mid+1,r);

}

a[root]=a[root<<1]+a[root<<1|1];

}

inline ll getsum(int x,int y,int root,int l,int r)

{

if(x<=l && r<=y)

return a[root];

int mid=(l+r)>>1;

ll ans=0;

if(x<=mid)//中间的mid点还是在左区间上

ans+=getsum(x,y,root<<1,l,mid);

if(y>mid)

ans+=getsum(x,y,root<<1|1,mid+1,r);

return ans;

}

int main()

{

//freopen("input.txt","r",stdin);

int n,q;

int cas=0;

while(~scanf("%d",&n))

{

build(1,1,n);

scanf("%d",&q);

int x,y,opt;

printf("Case #%d:\n",++cas);

while(q--)

{

scanf("%d%d%d",&opt,&x,&y);

if(x>y)swap(x,y);

if(opt==0)

update(x,y,1,1,n);

else

printf("%lld\n",getsum(x,y,1,1,n));

}

puts("");

}

return 0;

}