



[CF515E]Drazil and Park

| | |
|------------------------------|---|
| ⋮ Algorithm | 前缀和 线段树 |
| 🕒 Created | @Jun 20, 2020 11:45 PM |
| ⬇ Difficulty | NOI/NOI+/CTSC |
| ↗ Related to 近期更新 (Property) | |
| 🔗 URL | https://codeforces.com/contest/515/problem/E |

题目链接：

Problem - E - Codeforces

Drazil is a monkey. He lives in a circular park. There are n trees around the park. The distance between the i -th tree and $(i + 1)$ -st trees is a_i , the distance between the n -th tree and the

 <https://codeforces.com/contest/515/problem/E>



<https://codeforces.ml/contest/515/problem/E> (Codeforces镜像站)

Drazil and Park

Drazil is a monkey. He lives in a circular park. There are n trees around the park. The distance between the i -th tree and $(i+1)$ -st trees is d_i , the distance between the n -th tree and the first tree is d_n .

<https://www.luogu.com.cn/problem/CF515E>

题目大意：

就当是练习一下英语，大家自己去读题吧！

题解：

看到环想到什么？ 没错，断环为链，拷贝两份，方便处理。

看到两点间距离想到什么？ 没错，计算前缀和。

看到询问多次最大值想到什么？ 没错，线段树维护最值

那么

从第一步开始。

读入，

然后断环为链并拷贝，计算前缀和

```
for(int i=1;i<=n;i++)
{
    cin>>d[i];
    d[n+i]=d[i]; //断环为链
}
for(int i=1;i<=n;i++)
{
    cin>>h[i];
    h[i]*=2;
    h[n+i]=h[i]; //断环为链
}
for(int i=1;i<=2*n;i++)
    d[i]+=d[i-1]; //处理为前缀和的形式
```

由题意可知，若Drazil选择了两棵树 x 和 y ，那么他可以消耗的能量即为 $d_x + d(x + 1) + \dots + d(y - 1) + 2 * (h_x + h_y)$

转换一下，即 $(d_1 + d_2 + \dots + d(y - 1) + 2 * h_y) + (2 * h_x - (d_1 + d_2 + \dots + d(x - 1)))$

设 a_k 为 $(d_1 + d_2 + \dots + d(k-1) + 2 * h_k)$, 设 b_k 为 $(2 * h_k - (d_1 + d_2 + \dots + d(k-1)))$

```
t[x].a=h[l]+d[l-1]; //前面已经处理为前缀和了, 这里直接加减
t[x].b=h[l]-d[l-1];
```

题目中说孩子玩耍的区间为 $[a, b]$, 那么Dravil就不能经过该区间, 若 $a \leq b$, 则他可以经过区间 $[b+1, a+n+1]$ (因为此处已断环成链), 否则, 他可以经过的区间为 $[b+1, a-1]$, 设该区间为 I 。 $u, v \in I$, 要使消耗的能量最大, 即使 $a_u + b_v$ 最大, 即RMQ (区间最值) 问题, 因此可以用线段树维护每个区间的最大值。

```
int mid=(l+r)/2;
build(x*2,l,mid);
build(x*2+1,mid+1,r);
t[x].a=max(t[x*2].a,t[x*2+1].a);
t[x].b=max(t[x*2].b,t[x*2+1].b);
t[x].maxn=max(t[x*2].maxn,max(t[x*2+1].maxn,t[x*2].b+t[x*2+1].a)); //区间最值
```

m 天中每天孩子玩耍的区间可以看做是 m 个询问, 这样就转化为线段树上的问题了

最后附上AC代码:

```
#include <bits/stdc++.h>
using namespace std;
const int MAXN = 2e5+7;
const long long inf = 1LL*1e17;
long long d[MAXN],h[MAXN];
int n,m,a,b;
struct segment_tree //线段树
{
    int l,r;
    long long a,b,maxn;
}t[4*MAXN];
void build(int x,int l,int r)
{
    t[x].l=l;
    t[x].r=r;
    if(l==r)
    {
        t[x].a=h[l]+d[l-1];
        t[x].b=h[l]-d[l-1];
        t[x].maxn=-inf;
    }
    else
    {
        int mid=(l+r)/2;
        build(x*2,l,mid);
        build(x*2+1,mid+1,r);
```

```

        t[x].a=max(t[x*2].a,t[x*2+1].a);
        t[x].b=max(t[x*2].b,t[x*2+1].b);
        t[x].maxn=max(t[x*2].maxn,max(t[x*2+1].maxn,t[x*2].b+t[x*2+1].a)); //区间最
值
    }
}
segment_tree ask(int x,int l,int r) //询问操作
{
    int lv=t[x].l,rv=t[x].r;
    if(l<=lv && rv<=r)
        return t[x];
    int mid=(lv+rv)/2; //二分
    segment_tree t1,t2,t3;//左子树,右子树,合并后树
    t1.a=t1.b=t1.maxn=t2.a=t2.b=t2.maxn=t3.a=t3.b=t3.maxn= -1*inf; //全部初始化为负无
穷
    if(l<=mid)
        t1=ask(x*2,l,r);
    if(r>mid)
        t2=ask(x*2+1,l,r);
    t3.a=max(t1.a,t2.a);
    t3.b=max(t1.b,t2.b);
    t3.maxn=max(t1.maxn,max(t2.maxn,t1.b+t2.a)); //区间最值
    return t3;
}
int main()
{
    ios::sync_with_stdio(false); //优化
    cin>>n>>m;
    for(int i=1;i<=n;i++)
    {
        cin>>d[i];
        d[n+i]=d[i]; //断环为链
    }
    for(int i=1;i<=n;i++)
    {
        cin>>h[i];
        h[i]*=2;
        h[n+i]=h[i]; //断环为链
    }
    for(int i=1;i<=2*n;i++)
        d[i]+=d[i-1]; //处理为前缀和的形式
    build(1,1,2*n); //建树
    for(int i=1;i<=m;i++)
    {
        cin>>a>>b;
        if(a<=b)
            cout<<ask(1,b+1,a+n-1).maxn<<endl; //取补集进行操作
        else
            cout<<ask(1,b+1,a-1).maxn<<endl;
    }
    return 0;
}

```