

```
#include<bits/stdc++.h>
using namespace std;
const int T=100009;
const int maxn=200009;
const int SIZE=500;
const int B=34000;

int n,m,Tn;

int a[maxn];
int b[maxn];

bitset<T>f[B+9],now;
int c[T];

int nn;
struct Section{
    int l,r,id;
    bool operator < (const Section &rhs) const{
        if(l/SIZE==rhs.l/SIZE)return r<rhs.r;
        else return l<rhs.l;
    }
}p[maxn];

void insert(int x){
    now.set(x+c[x],1);++c[x];
}
void del(int x){
    --c[x];now.set(x+c[x],0);
}

int l1[B+9],r1[B+9],l2[B+9],r2[B+9],l3[B+9],r3[B+9];
void solve(){
    nn=0;
    for(int i=1;i<=m;++i){
        p[++nn].l=l1[i];p[nn].r=r1[i];p[nn].id=i;
        p[++nn].l=l2[i];p[nn].r=r2[i];p[nn].id=i;
        p[++nn].l=l3[i];p[nn].r=r3[i];p[nn].id=i;
    }
    sort(p+1,p+1+nn);
    int L=1,R=0;
    memset(c,0,sizeof(c));
    now.reset();
    for(int i=1;i<=m;++i)f[i].set();
    for(int i=1;i<=nn;++i){
        while(R<p[i].r)insert(a[++R]);
        while(L>p[i].l)insert(a[--L]);
        while(R>p[i].r)del(a[R--]);
        while(L<p[i].l)del(a[L++]);
        f[p[i].id]&=now;
    }
}
```

```
for(int i=1;i<=m;++i){
    int siz=r1[i]-l1[i]+r2[i]-l2[i]+r3[i]-l3[i]+3-3*f[i].count();
    printf("%d\n",siz);
}

int main(){
    scanf("%d%d",&n,&Tn);
    for(int i=1;i<=n;++i){
        scanf("%d",&a[i]);b[i]=a[i];
    }
    sort(b+1,b+1+n);
    for(int i=1;i<=n;++i)a[i]=lower_bound(b+1,b+1+n,a[i])-b;

    while(Tn--){
        ++m;
        scanf("%d%d%d%d%d",&l1[m],&r1[m],&l2[m],&r2[m],&l3[m],&r3[m]);
        if(m==B){
            solve();
            m=0;
        }
    }
    if(m)solve();
    return 0;
}
```

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:looking_up_at_the_starry_sky:%E6%88%91%E8%B7%91%E5%BE%97%E5%BE%88%E6%85%A2%E7%9A%84%E4%BB%A3%E7%A0%81

Last update: 2020/05/15 22:51