

#include<bits/stdc++.h>

#define ll long long

#define inf 1e18

#define mn 100005

using namespace std;

ll ans=0,n,m,bl,s[mn],pos[mn],c[mn];

struct w{ll l,r,id,a,b;}a[mn];

bool cmp(w a,w b)

{

if(pos[a.l]==pos[b.l])return a.r<b.r;

return a.l<b.l;

}

bool cmpid(w a,w b){return a.id<b.id;}//根据id调整答案顺序

void add(ll x){ans+=s[x]\*2+1;s[x]++;}//添加操作

void del(ll x){ans+=1-s[x]\*2;s[x]--;}//删除操作

void solve()

{ll i,l,r;

for(i=1,l=1,r=0;i<=m;i++)

{

while(l<a[i].l)del(c[l++]);//l小了就删

while(l>a[i].l)add(c[--l]);//l大了就加

while(r>a[i].r)del(c[r--]);//r大了就删

while(r<a[i].r)add(c[++r]);//r小了就加

if(a[i].l==a[i].r)

{

a[i].a=0;

a[i].b=1;

continue;

}

a[i].a=ans-(a[i].r-a[i].l+1);

a[i].b=(a[i].r-a[i].l+1)\*(a[i].r-a[i].l);

ll gcd=\_\_gcd(a[i].a,a[i].b);

a[i].a/=gcd;

a[i].b/=gcd;

}

}

int main()

{

ll x,y,z,i,j,k;

char ch;

scanf("%lld%lld",&n,&m);

for(i=1;i<=n;i++)

scanf("%lld",&c[i]);

bl=sqrt(n);

for(i=1;i<=n;i++)

pos[i]=(i-1)/bl+1;

for(i=1;i<=m;i++)

{

scanf("%lld%lld",&a[i].l,&a[i].r);

a[i].id=i;

}

sort(a+1,a+m+1,cmp);

solve();

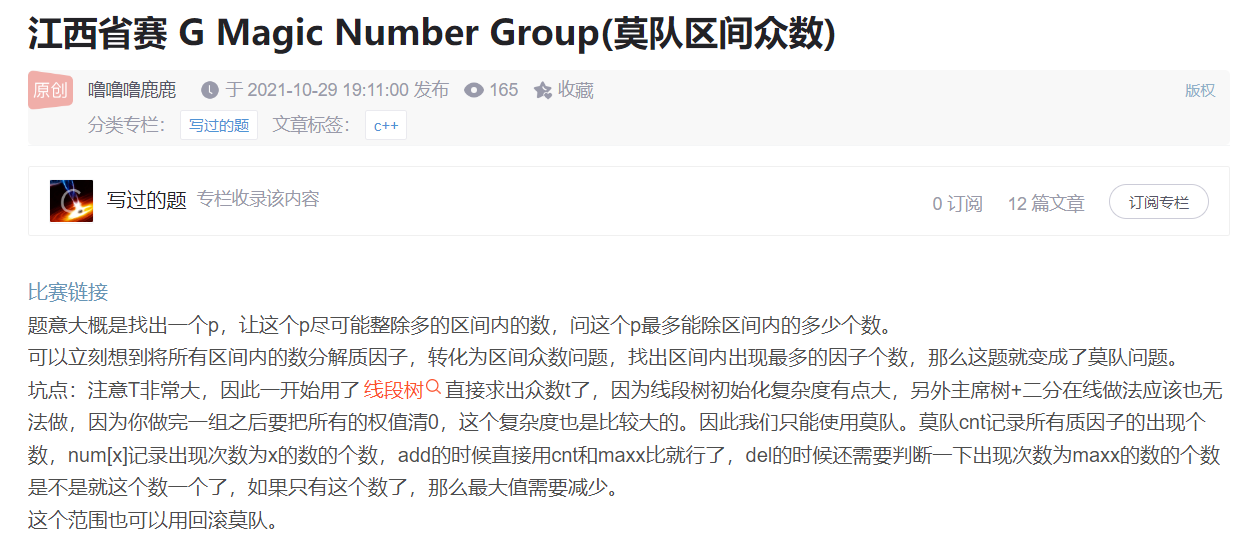
sort(a+1,a+m+1,cmpid);

for(i=1;i<=m;i++)

printf("%lld/%lld\n",a[i].a,a[i].b);

return 0;

}



using namespace std;

#define endl '\n'

const int maxn = 1e6 + 5;

const int maxm = 1e6 + 2;

int bel[maxn],sq,n,m,a[maxn],ans[maxn],cnt[maxm],maxx,num[maxn];

vector<int> g[maxn];

void getfactor(){

for (int i = 1; i <= n; ++i) {

int temp = a[i];

for (int j = 2; j <= sqrt(a[i]); ++j) {

if (temp % j == 0){

g[i].push\_back(j);

while (temp % j == 0) temp /= j;

}

}

if (temp > 1) g[i].push\_back(temp);

}

}

struct ask{

int l,r,id;

friend bool operator < (const ask a,const ask b){

if (bel[a.l] == bel[b.l]){

if (bel[a.l] & 1) return a.r < b.r;

else return a.r > b.r;

}

return a.l < b.l;

}

}ask[maxn];

void del(int pos){

for(auto &p : g[pos]){

num[cnt[p]]--;

if (maxx == cnt[p] && num[cnt[p]] == 0)

maxx--;

cnt[p]--;

num[cnt[p]]++;

}

}

void add(int pos){

for(auto &p : g[pos]){

num[cnt[p]]--;

cnt[p]++;

num[cnt[p]]++;

maxx = max(maxx, cnt[p]);

}

}

void solve(){

cin >> n >> m;

sq = sqrt(n);

for (int i = 1; i <= n; ++i) {

cin >> a[i];

bel[i] = (i - 1) / sq + 1;

}

getfactor();

for (int i = 1; i <= m; ++i) {

cin >> ask[i].l >> ask[i].r;

ask[i].id = i;

}

sort(ask + 1,ask + 1 + m);

int l = 1,r = 0;

for (int i = 1; i <= m; ++i) {

while (r < ask[i].r) add(++r);

while (r > ask[i].r) del(r--);

while (l < ask[i].l) del(l++);

while (l > ask[i].l) add(--l);

ans[ask[i].id] = maxx;

}

for (int i = 1; i <= m; ++i) {

cout << ans[i] << endl;

}

while (l <= r) del(l++);

for (int i = 1; i <= n; ++i) {

g[i].clear();

}

maxx = 0;

}

int main(){

std::ios::sync\_with\_stdio(false);

cin.tie(0); cout.tie(0);

int t;

cin >> t;

while (t--) solve();

}