```cpp
#include<cstdio>
#include<cassert>
#include<iostream>
#include<algorithm>
#include<vector>
#define pb push_back
using namespace std;
typedef long long LL;
const int N=2e5+5;
const LL INF=1e18;
vector<LL>pos,neg;
int main()
{
    int n;
    LL k;
    scanf("%d%lld",&n,&k);
    for(int i=1;i<=n;i++)
    {
        int a;
        scanf("%d",&a);
        if(a>0)pos.pb(a);
        else if(a<0)neg.pb(a);
    }
    sort(pos.begin(),pos.end());
    sort(neg.begin(),neg.end());
    LL ne=1LL*neg.size()*pos.size();
    LL po=1LL*pos.size()*(pos.size()-1)/2+1LL*neg.size()*(neg.size()-1)/2;
    LL ze=1LL*n*(n-1)/2-ne-po;
    if(k<=ne)
    {
        LL l=-INF,r=0,ans;
        while(l<=r)
        {
            LL mid=(l+r)/2,num=0;
            for(int i=0;i<neg.size();i++)
            {
                LL t=mid%neg[i]?mid/neg[i]+1:mid/neg[i];
                int x=lower_bound(pos.begin(),pos.end(),t)-pos.begin();
                num+=pos.size()-x;
            }
            if(num<k)l=mid+1;
            else r=mid-1,ans=mid;
        }
        cout<<ans;
    }
    else if(k<=ne+ze)puts("0");
    else
    {
        LL l=0,r=INF,ans;
        while(l<=r)
        {
```

```
            LL mid=(l+r)/2,num=0;
            for(int i=0;i<pos.size();i++)
            {
                if(pos[i]*pos[i]<=mid)num--;
                LL t=mid/pos[i];
                int x=lower_bound(pos.begin(),pos.end(),t+1)-pos.begin();
                if(x)assert(pos[x-1]*pos[i]<=mid);
                if(x<pos.size())assert(pos[x]*pos[i]>mid);
                num+=x;
            }
            for(int i=0;i<neg.size();i++)
            {
                if(neg[i]*neg[i]<=mid)num--;
                LL t=mid/neg[i];
                int x=lower_bound(neg.begin(),neg.end(),t)-neg.begin();
                assert(neg[x]*neg[i]<=mid);
                if(x)assert(neg[x-1]*neg[i]>mid);
                num+=neg.size()-x;
            }
            assert(num%2==0);
            num=num/2+ze+ne;
            if(num<k)l=mid+1;
            else r=mid-1,ans=mid;
        }
        cout<<ans;
    }
    return 0;
}
```