#include <iostream>

#include <set>

#include <map>

using namespace std;

long long d[100005];

int n;

int a[100005];

int lowbit(int x)

{

return -x & x;

}

void update(int p, int val)

{

while (p <= n)

{

d[p] += val;

p += lowbit(p);

}

}

long long query(int p)

{

long long res = 0;

while (p)

{

res += d[p];

p -= lowbit(p);

}

return res;

}

int main()

{

cin >> n;

set<int> s;

map<int, int> m;

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

{

cin >> a[i];

s.insert(a[i]);

}

set<int>::iterator it;

int cnt = 0;

for (it = s.begin(); it != s.end(); it++)

{

m[\*it] = ++cnt;

}

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

{

a[i] = m[a[i]];

}

long long ans = 0;

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

{

ans += query(n) - query(a[i]);

update(a[i], 1);

}

cout << ans << endl;

return 0;

}