

header

```
#define FastIO ios::sync_with_stdio(false);
cin.tie(0);cout.tie(0)

#include <ext/pb_ds/assoc_container.hpp> // Common file
using namespace __gnu_pbds;

/*
find_by_order(k) --> returns iterator to the kth largest
element counting from 0
order_of_key(val) --> returns the number of items in a
set that are strictly smaller than our item
*/
```

```
typedef tree<
int,
null_type,
less<int>,
rb_tree_tag,
tree_order_statistics_node_update>
ordered_set;

#include <ext/pb_ds/assoc_container.hpp>
using namespace __gnu_pbds;
gp_hash_table<int, int> table;
struct custom_hash {
static uint64_t splitmix64(uint64_t x) {
// http://xorshift.di.unimi.it/splitmix64.c
```

```
x += 0x9e3779b97f4a7c15;
x = (x ^ (x >> 30)) * 0xbf58476d1ce4e5b9;
x = (x ^ (x >> 27)) * 0x94d049bb133111eb;
return x ^ (x >> 31);
}
size_t operator()(uint64_t x) const {
static const uint64_t FIXED_RANDOM =
chrono::steady_clock::now().time_since_epoch().count();
return splitmix64(x + FIXED_RANDOM);
}
};
gp_hash_table<long long, int, custom_hash>
safe_hash_table;
```