

10

#include <bits/stdc++.h>

int main () {

int t;

cin >> t;

while (--t) {

int n;

cin >> n;

vector<string> v; map<string, int> m;

for (int i = 0; i < n; i++) {

int temp;

cin >> temp;

v.push_back(temp);

}

for (auto c : v) {

auto iter = m.find(c);

if (iter == m.end()) ~~iter++;~~

m[c] = 1;

else

m[c]++;

}

~~int n = 0;~~

string alpha; int count = 1;

for (auto iter = m.begin(); iter != m.end(); iter++)

{

if (iter->second > count) {

count = iter->second;

alpha = iter->first;

}

}

cout << alpha << endl << count;

}

return 0;

Assignment

1

```
#include <bits/stdc++.h>
```

```
int main () {
```

```
int t;
```

```
cin >> t;
```

```
while (--t) {
```

```
int n;
```

```
cin >> n; int a[n];
```

```
for (int i = 0; i < n; i++) {
```

```
    cin >> a[i];
```

```
}
```

```
int key;
```

```
cin >> key;
```

```
sort(a, a + n);
```

```
int al = 0, bl = n - 1;
```

```
while (al < bl) {
```

```
    if (a[al] + a[bl] < key) {
```

```
        al++;
```

```
    }
```

```
    else if (a[al] + a[bl] > key) {
```

```
        bl--;
```

```
    }
```

```
    else if (a[al] + a[bl] == key) {
```

```
        bl--;
```

```
    }
```

```
}
```

```
return 0;
```

```
#include <bits/stdc++.h>
```

```
void main () {
```

```
    int m;
```

```
    cin >> m; int a[m];
```

```
    int n;
```

```
    cin >> n; for (int i=0; i<m; i++) {
```

```
        cin >> a[i];
```

```
    }
```

```
    int n; cin >> n;
```

```
    int b[n];
```

```
    for (int i=0; i<n; i++) {
```

```
        cin >> b[i];
```

```
    }
```

```
    map<int, int> m;
```

```
    for (int i=0; i<m; i++) {
```

```
        m.insert({a[i], 0});
```

```
    }
```

```
    for (int i=0; i<n; i++)
```

```
    for (auto c: b)
```

```
    for (auto c: b) {
```

```
        auto ite = m.find(c);
```

```
        if (ite != m.end()) m[c]++;
```

```
    }
```

```
    for (auto ite = m; ite != m.end(); ite++) {
```

```
        if (ite->second > 0)
```

```
            cout << ite->first << " ";
```

```
    }
```

```
    return;
```

```
}
```