

```
list< vector<string> > teams; // element 0 is the team Name
                             // elements 1- are the players names
```

a)

```
int count = 0;
list< vector<string>>:: iterator lit
for (lit = teams.begin(); lit != teams.end(); lit++)
    count += lit->size() - 1;

cout << count << endl;
```

b)

```
string tname,pname;
cin >> tname >> pname;
list< vector<string>>:: lit;
for (lit = teams.begin(); lit != teams.end(); lit++)
    if ( (*lit)[0] == tname ) {
        (*lit).push_back(pname);
        stable_sort( (*lit).begin()+1; (*lit).end() );
    }
```

```
template <class T>
bool bsearch( vector<T> & V, int low, int high, T val) {

    if ( low == high ) return false;

    int mid = (low+high)/2;
    if ( V[mid] == val ) return true;

    if ( V[mid] > val ) return bsearch( V, low, mid, val);

    if ( V[mid] < val ) return bsearch(V,mid+1, high, val);

    return false; // for overly enthusiastic compilers
}
```

Example call:

```
vector<int> X;
bsearch(X, 0, X.size(), 103);
```