

Google Code Jam Round 1C 2016

Juan Pablo Ramírez Guerra
A01226136
ITESM GDA

Problem A. Senate Evacuation

The problem



In the senate, a fire has just started and the whole room needs to be evacuated. There are senators voting inside. Each senator belongs to a party.

But there are some rules in the senate that makes this evacuation harder:

- Voting can occur at any time, even during evacuations
- Therefore, senators must be evacuated in a way that ensures that no party has an absolute majority.
- This means that: of all senators still in the building, there can't be more than 50% of the s political party.

Political Party

A political party will be represented by one letter of the alphabet. This means that there are, at most, 26 political parties. {A, B, C, D, ..., X, Y, Z}

The Input

The input for this problem is given as follows:

T The first line of the input represents the number of test cases that will be used.

Every two lines of input a test case will be declared.

N The first line of a test case includes a single integer N, being the # of parties present.

P The second line consists of P, where P is the number of senators of the party.

Sample:

Two test cases
Case 1 { 2 → 2 political parties
{ 2 2 → A:2, B:2
Case 2 { 3 → 3 political parties
{ 3 2 2 → A:3, B:2, C:2

The output

For every case scenario, output one line as follows: case #X : y

where X is the number of the test, and y is the evacuation plan.

The evacuation plan is a set of instructions in singles or pairs. It represents the parties of the senators to evacuate.

case #1: AB BA
case #2: AA BC C AB

The Process

For example, case #1, in this case, where the two political parties have the same number of members in it. So, it's pretty straightforward.

So, apparently, the only rule to evaluate is to check if the remaining parties are an absolute majority. For case #2, we need to get rid of the A party first. So, the first two to evacuate are AA. And so on...

Some limits

Test cases T $1 \leq T \leq 50$
 # of Parties N $2 \leq N \leq 26 \rightarrow 26$ letters in the alphabet.
 $\sum P_i$ $P_i = P[i]$ $P_i \leq 9$

- * It's assumed that, from the beginning, there's no absolute majority within the given set.
- * If there's no majority, when 2 parties are present, they MUST have the same amount of senators. So, evacuate in pairs until empty.

Possible Solution

```

Input Handling  $\rightarrow O(n^2)$ 
int test-cases = 0;
int parties = 0;
int senators[255] = 0;
scanf ("%d", test-cases);
for(int i=0; i<test-cases; i++){
    scanf ("%d", parties);
    for (int j=0; j<parties; j++) {
        scanf ("%d", &senators[j]);
    }
}
?? Solve the problem here;
}
    
```

The problem solution

```

string Solution (parties, senators[])
char soln[255];
if (parties <= 2)
    int flag=0;
    int max=senators[0];
    for(int i=0; i<max; i++){
        if (flag == 0)
            strcat (soln, "\sAB\s");
        else
            strcat(soln, "\sBA\s");
    }
}
else
    quicksort (senators, parties, sizeof(senators[1]), cmpfunc);
    for (int i=0; i<senators[0]; i++){
        if (senators[i] == 0)
            break; // Party is now empty.
        else { // Party not empty yet
            senators[i]--;
            strcat (soln, (char)i);
        }
    }
return soln;
}
    
```

Sort from biggest to smallest. This guarantees no absolute majority.

Samples used while testing.

TEST #1

Parties = 4	
senators = [4, 3, 5, 0]	Sorting
output: ABCD ABCD ABCD ABCD ABC AB A	
initial	
7 6 5 4 3 2 1 0	
6 5 4 3 2 1 0	
5 4 3 2 1 0	
4 3 2 1 0	
Σ 22 18 14 10 6 3 1 ✓	
↳ Remaining senators in the room	

TEST #2:

Parties = 3	
senators = [3, 2, 1]	→ sorted ✓
output = ABC ABC A	
initial	
3 2 1 0	
2 1 0	
1 0	
Σ 6 3 1 ✓	

* Even though it looks like 3 or 4 people at the same time, it only evacuates 1 senator in every cycle.

Complexity Analysis

In order to avoid returning to other positions in the array, I used the quicksort included in stdlib.h. Throughout the source code, I decided to break it into 3 main pieces as described next:

- ① Quicksort Function $O(n^2)$ $\Omega(n \log n)$
- ② Input Handling $O(n^2)$
- ③ The solution to the problem $O(n^2)$, unless parties < 2, which is impossible because it will occasionate absolute majority.

So, the average complexity for this project is $O(n^2)$

References :

Ronell, E. Big O Cheat Sheet. Retrieved from bigocheatsheet.com.