

## Bibi's Array

Bibi also has an array containing  $N$  elements. Like Lili, Bibi wants to know the highest frequency (most occurrences) and all elements which have that frequency.

### Format Input

The first line contains an integer  $T$  stating the number of test cases.

For each test case, the first line contains a single integer  $N$  which indicate the number of element in the array. The next line contains  $N$  integers  $X_i$  ( $1 \leq i \leq N$ ) which indicate  $i^{th}$  element in the array.

### Format Output

Consists of  $T$  lines where each line has the format “Case # $X$ :  $Y$ ”, where  $X$  is the test case number starting at 1 and  $Y$  is the highest frequency. Next line contains all elements which have that frequency sorted in ascending order.

### Constraints

- $1 \leq T \leq 20$
- $2 \leq N \leq 20.000$
- $1 \leq X_i \leq 2 \times 10^5$

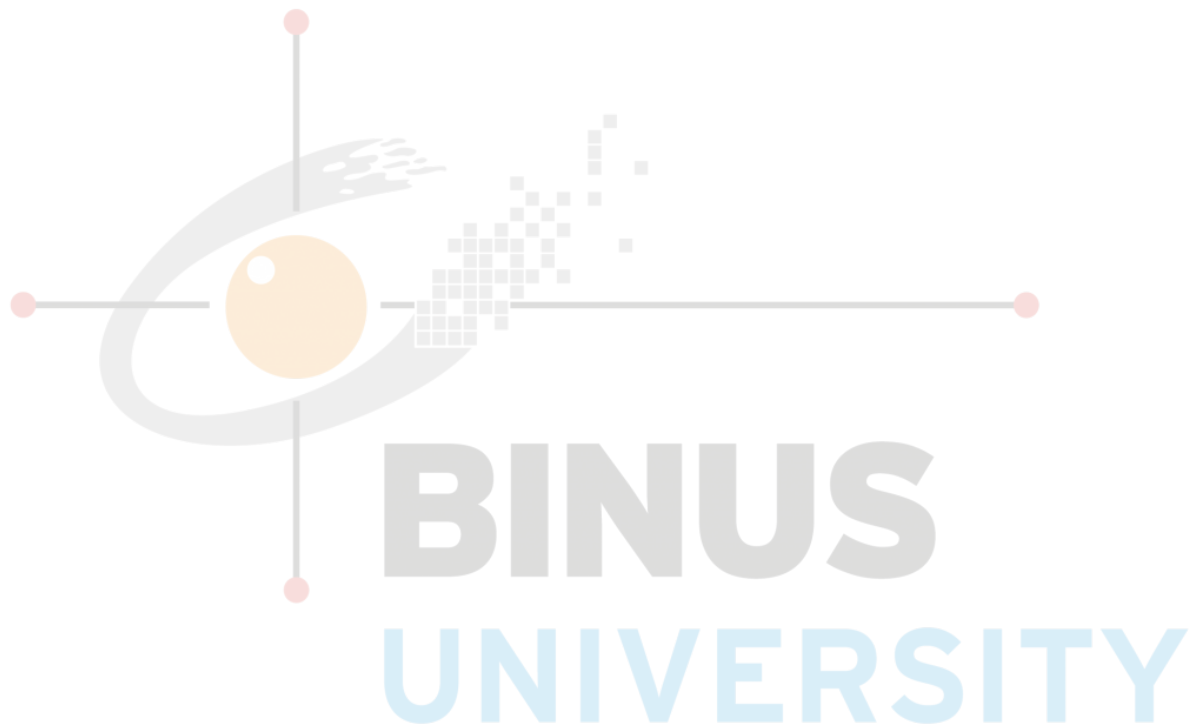
### Sample Input (standard input)

```
3
8
1 1 2 2 3 4 5 5
8
5 5 4 3 2 2 1 1
4
1 1 1 3
```

---

## Sample Output (standard output)

```
Case #1: 2
1 2 5
Case #2: 2
1 2 5
Case #3: 3
1
```



## Bibi's Array

Bibi mempunyai sebuah array yang berisi  $N$  buah elemen. Seperti Lili, Bibi ingin tahu elemen dengan frekuensi(banyak kemunculan) tertinggi dan semua elemen yang memiliki frekuensi tersebut.

### Format Input

Baris pertama berisi sebuah bilangan bulat  $T$  yang menyatakan banyaknya kasus uji. Untuk setiap kasus uji, baris pertama berisi sebuah bilangan bulat  $N$  yang menyatakan banyak elemen pada array. Pada baris selanjutnya, terdapat  $N$  bilangan bulat  $X_i$  ( $1 \leq i \leq N$ ) yang menyatakan nilai elemen ke- $i$  pada array.

### Format Output

Terdiri dari  $T$  baris yang setiap barisnya memiliki format “Case # $X$ :  $Y$ ”, dimana  $X$  adalah nomor kasus uji mulai dari 1 dan  $Y$  adalah frekuensi tertinggi. Baris selanjutnya berisi semua elemen dengan frekuensi tertinggi terurut menaik.

### Constraints

- $1 \leq T \leq 20$
- $2 \leq N \leq 20.000$
- $1 \leq X_i \leq 2 \times 10^5$

### Sample Input (standard input)

```
3
8
1 1 2 2 3 4 5 5
8
5 5 4 3 2 2 1 1
4
1 1 1 3
```

---

## Sample Output (standard output)

```
Case #1: 2
1 2 5
Case #2: 2
1 2 5
Case #3: 3
1
```

