Coronavirus Spread | CodeChef

codechef.com/problems/COVID19

Coronavirus Spread Problem Code: COVID19 Submit

Read problem statements in Hindi, Bengali, Mandarin Chinese, Russian, and Vietnamese as well.

There are NN people on a street (numbered 11 through NN). For simplicity, we'll view them as points on a line. For each valid ii, the position of the ii-th person is XiXi.

It turns out that exactly one of these people is infected with the virus COVID-19, but we do not know which one. The virus will spread from an infected person to a non-infected person whenever the distance between them is at most 22. If we wait long enough, a specific set of people (depending on the person that was infected initially) will become infected; let's call the size of this set the *final number of infected people*.

Your task is to find the smallest and largest value of the final number of infected people, i.e. this number in the best and in the worst possible scenario.

Input

- The first line of the input contains a single integer TT denoting the number of test cases. The description of TT test cases follows.
- The first line of each test case contains a single integer NN.
- The second line contains NN space-seperated integers X1,X2,...,XNX1,X2,...,XN.

Output

For each test case, print a single line containing two space-separated integers — the minimum and maximum possible final number of infected people.

Constraints

- 1≤T≤2,0001≤T≤2,000
- 2≤N≤82≤N≤8
- 0≤Xi≤100≤Xi≤10 for each valid ii
- X1<X2<...<XNX1<X2<...<XN

Subtasks

Subtask #1 (10 points): N≤3N≤3

Subtask #2 (90 points): original constraints

Example Input

```
3
2
3 6
3
1 3 5
5
1 2 5 6 7
```

Example Output

```
1 1
3 3
2 3
```

Explanation:

Example case 1: The distance between the two people is 3, so the virus cannot spread and at the end, there will always be only one infected person.

Example case 2: The distance between each two adjacent people is 2, so all of them will eventually get infected.
