Operating System Problems (Task Scheduling)

As you all know, Operating System (OS) is a software that controls the execution of computer programs and may provide various services. Although modern operating systems are very easy to use and give us lots of services, their designing is not so easy. It needs lots of work and time to design a good OS. But no OS is perfect. Either they are not so usable or they are not so secure (got tons of viruses). Lingates, as an excellent programmer, wants perfection in all things. So he decided to not use this OSs and write his own OS. But he found out this work is not so easy to do alone even for a programmer like him. So he needs your help.

As a programmer with honor he won't let you do *equal or more* work in a turn then him (Means Lingates has to do more work then you on each turn). Each turn you both divide the works between yourself. But the works are related so it's better to take works which are adjacent. Like if there are **5** works **5, 2, 7, 1, 3** then Lingates would take 1st work and give you the 2nd work. But you can't take 2nd and 3rd work because that sum to **7+2 = 9** which is more than his work (**5**). Again you can't take 2nd and 4th work because they are not adjacent. After you both finish your work you both will again distribute your works. There is no limit how much work you both can take on a turn. Like Lingates can take 1st and 2nd work on first turn and give you *no work* on that turn or he can take 1st, 2nd and 3rd turn (total **5+2+7=14**) and give you only 4th (**1**) or both works (**1+3=4**). As you are helping him he let you to divide the works but it has to be that Lingates has to do more work on each turn. As you are also a programmer with honor you also like to take the turns and divide the work on each turn such that it ***maximizes your total works***.

Write a program which will take the list of work and give the total amount of work done by Lingates and you if you make the list optimally. Again Lingates will do any amount of work you will give him in any turn as long as your work on that turn is less than his.

Input

The first line will be number of test cases (**T<=500**) and each case will start with an integer n (**0<=n<=100**). In the flowing line n numbers will given as the amount of works (all will be **non negative integer <1000**)

Output

A single line for each test case first the total work of Lingates and second the work of your.

Example

Input:

```
4
3
1 2 3
5
5 2 7 1 3
5
6 6 6 6 6
7
4 9 5 7 6 5 1
```

Output:

```
6 0
12 6
18 12
20 17
```

Explanation:

In 1st case you have to give Lingates all 3 works because 1< (2+3) and (1+2) =3, so you can't take any work.

In 2nd case you can give Lingates 1st work (5) and you can take 2nd (2). Then you can give Lingates the 3rd (7) and you can take the rest (1+3).

In 3rd case you can give Lingates 1st to 3rd work (6+6+6=18). You can take the left two works (6+6=12).

In 4th case you can give Lingates 4 and 9 and you will take 5 and 7. Then in next turn you can give him 6 and you own take 5. Then you can give him 1.
So totaling (4+9+6+1) =20 and (5+7+5) =17.