Bricks Game

HackerRank

You and your friend decide to play a game using a stack consisting of N bricks. In this game, you can alternatively remove 1, 2 or 3 bricks from the top, and the numbers etched on the removed bricks are added to your score. You have to play so that you obtain the maximum possible score. It is given that your friend will also play optimally and you make the first move.

As an example, bricks are numbered $arr = [1, 2, 3, 4, 5]$. You can remove either $[1] = 1$, $[1, 2] = 3$ or $[1, 2, 3] = 6$. For your friend, your moves would leave the options of 1 to 3 elements from $[2, 3, 4] = 9$ leaving 5 for you (total score = 6), $[3, 4, 5] = 12$ or $[4, 5] = 9$. In this case, it will never be optimal for your friend to take fewer than the maximum available number of elements. Your maximum possible score is 6, achievable two ways: 1 first move and 5 the second, or $[1, 2, 3]$ in your first move.

Function Description

Complete the *bricksGame* function in the editor below. It should return an integer that represents your maximum possible score.

bricksGame has the following parameter(s):

- *arr*: an array of integers

Input Format

The first line will contain an integer t, the number of test cases.

Each of the next t pairs of lines are in the following format:
The first line contains an integer n, the number of bricks in arr.
The next line contains n space-separated integers $arr[i].

Constraints

$1 \leq t \leq 5$
$1 \leq n \leq 10^5$
$0 \leq arr[i] \leq 10^9$

Output Format

For each test case, print a single line containing your maximum score.

Sample Input

```
2
5
999 1 1 1 0
5
0 1 1 1 999
```

Sample Output

```
1001
999
```

Explanation

In first test case, you will pick 999,1,1. If you play in any other way, you will not get a score of 1001.
In second case, best option will be to pick up the first brick (with 0 score) at first. Then your friend will choose the next three blocks, and you will get the last brick.