Median Stream

You're given a list of n integers `arr[0..(n-1)]`. You must compute a list `output[0..(n-1)]` such that, for each index `i` (between `0` and `n-1`, inclusive), `output[i]` is equal to the median of the elements `arr[0..i]` (rounded down to the nearest integer).

The median of a list of integers is defined as follows. If the integers were to be sorted, then:

- If there are an odd number of integers, then the median is equal to the middle integer in the sorted order.
- Otherwise, if there are an even number of integers, then the median is equal to the average of the two middle-most integers in the sorted order.

Signature

```
int[] findMedian(int[] arr)
```

Input

n is in the range [1, 1,000,000].
Each value `arr[i]` is in the range [1, 1,000,000].

Output

Return a list of n integers `output[0..(n-1)]`, as described above.

Example 1

n = 4
`arr` = [5, 15, 1, 3]
`output` = [5, 10, 5, 4]

The median of `[5]` is 5, the median of `[5, 15]` is `(5 + 15) / 2 = 10`, the median of `[5, 15, 1]` is 5, and the median of `[5, 15, 1, 3]` is `(3 + 5) / 2 = 4`.

Example 2

n = 2
`arr` = [1, 2]
`output` = [1, 1]

The median of `[1]` is 1, the median of `[1, 2]` is `(1 + 2) / 2 = 1.5` (which should be rounded down to 1).