Peak Index in a Mountain Array

An array arr a **mountain** if the following properties hold:

- arr.length >= 3
- There exists some i with 0 < i < arr.length - 1 such that:
    - arr[0] < arr[1] < ... < arr[i - 1] < arr[i]
    - arr[i] > arr[i + 1] > ... > arr[arr.length - 1]

Given a mountain array arr, return the index i such that arr[0] < arr[1] < ... < arr[i - 1] < arr[i] > arr[i + 1] > ... > arr[arr.length - 1].

You must solve it in O(log(arr.length)) time complexity.

Example 1:

Input: arr = [0,1,0]

Output: 1

Example 2:

Input: arr = [0,2,1,0]

Output: 1

Example 3:

Input: arr = [0,10,5,2]

Output: 1

Constraints:

- 3 <= arr.length <= 10^5
- 0 <= arr[i] <= 10^6
- arr is **guaranteed** to be a mountain array.