Easy Codewriting 2000

Note: Try to solve this task in `O(n)` time using `O(1)` additional space, where `n` is the number of elements in `ℓ`, since this is what you'll be asked to do during an interview.

Given a singly linked list of integers, determine whether or not it's a palindrome.

Note: in examples below and tests preview linked lists are presented as arrays just for simplicity of visualization: in real data you will be given a head node `ℓ` of the linked list

Example

- For `l = [0, 1, 0]`, the output should be
  `isListPalindrome(l) = true`;
- For `l = [1, 2, 2, 3]`, the output should be
  `isListPalindrome(l) = false`.

Input/Output

- **[execution time limit] 20 seconds (scala)**
- **[input] linkedlist.integer l**

  A singly linked list of integers.

  Guaranteed constraints:
  `0 ≤ list size ≤ 5 · 10^5`,
  `-10^9 ≤ element value ≤ 10^9`.

- **[output] boolean**

  Return `true` if `l` is a palindrome, otherwise return `false`.

[Scala] Syntax Tips

```
def helloWorld(name: String): String = {
    println("This prints to the console when you Run Tests")
    "Hello, " + name
}
```