Easy Codewriting 2000

You're given `2` huge integers represented by linked lists. Each linked list element is a number from `0` to `9999` that represents a number with exactly `4` digits. The represented number might have leading zeros. Your task is to add up these huge integers and return the result in the same format.

Example

- For `a = [9876, 5432, 1999]` and `b = [1, 8001]`, the output should be `addTwoHugeNumbers(a, b) = [9876, 5434, 0]`.

  Explanation: `987654321999 + 18001 = 987654340000`.

- For `a = [123, 4, 5]` and `b = [100, 100, 100]`, the output should be `addTwoHugeNumbers(a, b) = [223, 104, 105]`.

  Explanation: `12300040005 + 10001000100 = 22301040105`.

Input/Output

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

  The first number, without its leading zeros.

  Guaranteed constraints:

  $0 \le a\ size \le 10^4$,

  $0 \le element\ value \le 9999$.

- **[input] linkedlist.integer b**

  The second number, without its leading zeros.

  Guaranteed constraints:

  $0 \le b\ size \le 10^4$,

  $0 \le element\ value \le 9999$.

- **[output] linkedlist.integer**

  The result of adding `a` and `b` together, returned without leading zeros in the same format.

[Scala] Syntax Tips

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