

001-099

001. Length of a List

Given a list **L**, return length of it.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: 7

Example 2:

Input: L = []

Output: 0

002. Reverse a List

Given a list **L**, return a reversed list.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: [7, 6, 5, 4, 3, 2, 1]

Example 2:

Input: L = []

Output: []

Example 3:

Input: L = [element]

Output: [element]

003. Maximum Value

Given a number **A** and a number **B**, return a maximum value.

Example 1:

Input: A = 10, B = 3

Output: 10

Example 2:

Input: A = 1, B = 7

Output: 7

Example 3:

Input: A = 2, B = 2

Output: 2

004. Maximum Value in a List

Given a list **L** of numbers, return a maximum value.

Example 1:

Input: L = [1, 7, 2, -3, 5, 0]

Output: 7

Example 2:

Input: L = [4]

Output: 4

Example 3:

Input: L = [-1, -9, -4]

Output: -1

Constraints:

- 1 <= Length of L

005. Membership

Given an element **X** and a list **L**, return true if **X** is a member of **L**, false otherwise.

Example 1:

Input: X = alex, L = [bob, james, alan, alex, simon]

Output: true

Example 2:

Input: X = sam, L = [bob, james, alan, alex, simon]

Output: false

Example 3:

Input: X = 5, L = [1, 2, 3, 4, 5]

Output: true

Example 4:

Input: X = 0, L = [1, 2, 3, 4, 5]

Output: false

Example 5:

Input: X = 0, L = []

Output: false

006. Parity

Given an integer **N**, return atom **even** if the parity of **N** even, otherwise return atom **odd**.

Example 1:

Input: N = 5

Output: odd

Example 2:

Input: N = 8

Output: even

007. List Length Parity

Given a list **L**, return atom **even** if the list's length parity is even, otherwise return atom **odd**.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: odd

Example 2:

Input: L = [1, 2, 3, 4]

Output: even

008. Checking List Length Parity

Given a list **L**. Define two functions: **even_length** and **odd_length**, so that they return are true if their argument is a list of even or odd length respectively.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Call: even_length(L)

Output: false

Call: odd_length(L)

Output: true

Example 2:

Input: L = [1, 2, 3, 4]

Call: even_length(L)

Output: true

Call: odd_length(L)

Output: false

009. Sum of Elements in a List

Given a list **L** of numbers, return the sum of all elements in the list.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: 28

Example 2:

Input: L = []

Output: 0

Example 3:

Input: L = [12]

Output: 12

Example 4:

Input: L = [10, 0, -5]

Output: 5

010. Removing Last 3 Elements in a List

Given a list L, return a list without 3 last elements.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: [1, 2, 3, 4]

Example 2:

Input: L = []

Output: 0

Example 3:

Input: L = [sun, moon]

Output: []

Example 4:

Input: L = [jane, laura, jerry, katty]

Output: [jane]

011. Last Element

Given a list L, return the last element.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: 7

Example 2:

Input: L = [sun, moon]

Output: moon

Example 3:

Input: L = [1]

Output: 1

Example 4:

Input: L = [jane, laura, jerry, katty]

Output: katty

Constraints:

- 1 <= Length of L

012. Deleting an Item

Given an item **X** and a list **L**, return a list in which the first occurrence of item **X** has been removed.

Example 1:

Input: X = 2, L = [1, 2, 3, 4, 5, 6, 7]

Output: [1, 3, 4, 5, 6, 7]

Example 2:

Input: X = elisa, L = [bob, mark, elisa, greg]

Output: [bob, mark, greg]

Example 3:

Input: X = 1, L = [1]

Output: []

013. Ordered List

Given a list **L** of numbers, return **true** if the list is ordered, **false** otherwise.

Example 1:

Input: L = [1, 2, 3, 4, 5, 6, 7]

Output: true

Example 2:

Input: L = [1, 2, 7, 5, 9]

Output: false

Example 3:

Input: L = [10]

Output: true

Constraints:

- 1 <= Length of L

