University of Jijel
Faculty of Exact Sciences and Computer Science
Computer Science Department

Master I – IA
Programming Language Semantics (SL)
2025/2026

Tutorials 03:
Lisp

Exercise 01: The goal is to calculate the sum of all elements in an array.

Procedural Approach

- Write a function that takes an array of numbers and computes the sum by iterating through the list, adding each element to an accumulator.

Functional Approach

- Write a recursive function that computes the sum of the elements of a list.

Exercise 02: Reverse the elements in a list.

Procedural Approach:

- Write a function that reverses an array using a loop (e.g., by iterating over the array and building a new reversed array).

Functional Approach:

- Write a recursive function to reverse a list.

Exercise 03: Apply a given function (e.g., squaring a number) to all elements of the list.

Procedural Approach:

- Write a function that iterates through the array and applies the function to each element.

Functional Approach:

- Write a recursive function that applies the given function to each element of the list.

Exercise 04 : Check if a list contains a specific element.

Procedural Approach:

- Write an algorithm that checks if a specific element exists in an array.

Functional Approach:

- Write recursive function that checks if a specific element exists in a list.

Exercise 05: Count the number of occurrences of an element in a list.

Procedural Approach:

- Write a function that iterates through the array, counting occurrences of the target element.

Functional Approach:

- Write a recursive function that checks each element of the list. If it matches the target, increment the count.

Exercise 06: Merge two sorted lists.

Procedural Approach:

- Write a function that takes two sorted lists and merges them into a single sorted list.

Functional Approach:

- Write a recursive function that compares the `car` of both lists.
- Place the smaller element in the result list, and recurse on the `cdr` of the list where the smaller element was chosen from.