



**Universidade Federal de Goiás**  
**Instituto de Informática**

**LPP – LINGUAGENS E PARADIGMAS DE PROGRAMAÇÃO**

**Laboratório (29/10/2025) – individual**

**Funções em Common Lisp e Clojure**

Faça um arquivo no formato .ipynb que rode códigos-fontes em Common Lisp e Clojure e grave os exercícios em formato .clj e .lisp e o .sh da instalação do Clojure. Copie todos os códigos-fontes desta tarefa e capture o output ao rodá-los no próprio arquivo .ipynb. Envie suas respostas em um único arquivo compactado no SIGAA. Para cada exercício executado imprima o seu nome e matrícula, inclua estes dados nos códigos-fontes.

Matrícula: \_\_\_\_\_ Nome: \_\_\_\_\_

**Exercício 1:**

a) O que faz o seguinte código-fonte em Common Lisp?

```
(format t "lista: ~a~%" (sort '(1 3 2 5 4) #'<))
```

```
(format t "lista: ~a~%" (sort '(1 3 2 5 4) #'>))
```

b) Os símbolos #'< e #'> do código-fonte são uma forma abreviada de quais comandos.

c) Inclua no código-fonte o seu nome e matrícula e os imprima.

d) Capture o output ao rodar o código-fonte em uma célula do .ipynb.

**Exercício 2:** Passe o código-fonte do exercício 1 para o Clojure. Inclua no código-fonte o seu nome e matrícula e os imprima e capture o output ao rodar o código-fonte em uma célula do .ipynb.

**Exercício 3:**

a) O que faz o seguinte código-fonte em Common Lisp?

```
(defparameter numbers '(1 2 3))
```

```
(defparameter squares (mapcar (lambda (x) (* x x)) numbers))
```

```
(format t "Soma: ~a~%" (reduce #' + squares))
```

b) O que significa lambda e mapcar e reduce do código-fonte em Common Lisp.

c) Inclua no código-fonte o seu nome e matrícula e os imprima.

d) Capture o output ao rodar o código-fonte em uma célula do .ipynb.

#### Exercício 4:

a) O que faz o seguinte código-fonte em Clojure?

```
(def n 5)

(if (> n 0)
  (println "n é positivo")
  (println "n não é positivo"))

(cond
  (= n 0) (println "n é zero")
  (> n 0) (println "n é maior que zero")
  :else   (println "n é negativo"))

(loop []
  (when (< @contador 5)
    (swap! contador inc)
    (println "Contador:" @contador))
  (recur)))
```

b) O que significa def, swap e recur do código-fonte do Clojure.

c) Inclua no código-fonte o seu nome e matrícula e os imprima.

d) Capture o output ao rodar o código-fonte em uma célula do .ipynb

**Exercício 5:** Passe o código-fonte do exercício 4 para o Common Lisp. Inclua no código-fonte o seu nome e matrícula e os imprima e capture o output ao rodar o código-fonte em uma célula do .ipynb.

**Exercício 6:** Implemente em Common Lisp ou em Clojure. Inclua no código-fonte o seu nome e matrícula e os imprima e capture o output ao rodar o código-fonte em uma célula do .ipynb.

a) O que é uma função mônada?

b) Implemente um exemplo de função da mônada Maybe. Gere o arquivo em formato .clj ou .lisp, rode o programa e inclua no código-fonte o seu nome e matrícula e os imprima.

c) Implemente um exemplo de funções bind (ou >>=) e return (ou unit). Gere o arquivo em formato .clj ou .lisp, rode o programa e inclua no código-fonte o seu nome e matrícula e os imprima.