

# Programmation comparée – TP 3 :

## Programmation fonctionnelle

Université Paris Diderot – Master 2

(2014-2015)

**Exercice 1 (LISP)** *Faites les exercices de programmation suivants dans un LISP de votre choix.*

1. *Ecrire un programme qui affiche "Hello Lisp World!".*
2. *Implémenter une solution au problème des anagrammes.*
3. *Ecrire une fonction `pow n` qui produit la S-expr représentant la fonction qui à  $x$  associe  $x^n$ .*

□

**Exercice 2 (Production, consommation, croissance ( ? ))** *Faites les exercices suivants dans un langage fonctionnel de votre choix. Soit le programme `genrandom` qui produit sur sa sortie standard une liste finie de taille aléatoire d'entiers positifs aléatoires séparés par des retours chariots. On suppose qu'au bout de  $N$  lignes, `genrandom k` ne peut produire de nombre supérieur à  $k \cdot N$ .*

*On souhaite écrire un programme `sortprime` qui lit la sortie standard de `genrandom`, affiche le plus tôt possible le nombre produit si il est premier, et, une fois que `genrandom` a fini son travail, affiche la liste triée des nombres premiers rencontrés.*

1. *Écrire dans un langage impératif de votre choix mais en utilisant uniquement un tableau d'octets de taille 640Ko, un programme `sortprime` qui utilise l'algorithme du crible d'Eratosthène pour déterminer rapidement si un nombre est premier. Quelle est la consommation mémoire de votre programme ?*
2. *Réécrire ce même programme mais en s'autorisant des allocations dynamiques. Quelle est la consommation mémoire de votre programme ?*
3. *Réécrire ce même programme en utilisant les abstractions et le langage que vous souhaitez. Quelle est la consommation mémoire de votre programme ?*
4. *Dans un langage fonctionnel, réécrire ce programme de façon la plus pure possible. Quelle est la consommation mémoire de votre programme ?*
5. *Comparez les différentes versions de ce programme.*

□