IA01 — TD03

---

1 Boucles

Donner cinq fonctions différentes permettant d'imprimer tous les éléments de la liste suivante :
`(setq ll '( A 1 BB 2 CCC 3 DDD 4))`

2 Représentation et manipulation d'une page Web HTML

Préliminaires

Rappel de la syntaxe et de la sémantique HTML.

2.1 Représentation HTML

Construire une représentation sous forme de liste d'un contenu HTML tel que celui-ci :

```
<html>
<header>
    <title>Ma page</title>
</header>
<body>
    <h1>Un titre</h1>
    <p>Soror et aemula Romae</p>
</body>
</html>
```

2.2 Manipulation de la représentation

Écrire une fonction `make-html` permettant l'impression du code HTML correspondant à la représentation d'une page HTML telle que définie en 2.1. La page pourra être différente de celle donnée en exemple.

2.3 Utilisation des fichiers

Écrire une fonction permettant de générer le fichier .html correspondant interprétable par un navigateur Web.

Fonctions utiles pour réaliser le 2.3

```
;;Chargement d'un fichier dans un environnement Lisp
(load "my-file")
; load évalue séquentiellement chacune des formes du fichier


;; Création d'un fichier
(setq file (open "test" :if-does-not-exist :create :direction :output
                 :if-exists :overwrite))
```

Le stream (en français flux) file est associé au fichier "test" situé dans le répertoire courant. Les opérations d'entrée-sortie sont réalisées ensuite à l'aide de ce flux.
Notez l'association des mots-clés par paire :

- Le fichier est ouvert en écriture (:direction :output)
- si le fichier n'existe pas on le crée (:if-does-not-exist :create)
- s'il existe, le contenu de l'ancien fichier est perdu (:if-exists :overwrite)

Voir l'aide pour les autres possibilités.

```
;; écriture
(write "zoe" :stream file)

;; fermeture
(close file)

;; La macro with-open-file facilite la manipulation des fichiers.
Exemples :
(with-open-file (f "test")
  (print (read f)))

(with-open-file (file "test" :direction :output :if-exists :append)
  (print  "Albert" file))

;; On peut utiliser l'argument optionnel output-stream dans les fonctions
d'impression

(print "Albert" file)

;; Pour l'exercice, on pourra écrire par exemple
(with-open-file (file "test"
                      :if-does-not-exist :create
                      :if-exists :overwrite
                      :direction :output)
  (make-html *html* file))

; on passe le flux en paramètre à make-html. Il faut modifier les fonctions
d'impression en conséquence.
```