Maîtrise Informatique 1999-2000 — Bordeaux I

TFP

TD 9 - Common Lisp

Encore des macros

1. Observation de l'expansion des macros d'usage courant comme `cond`, `defstruct`, `do`, `setf`, `multiple-value-bind`, `and`, `with-hash-table-iterator`):

   (a) Utiliser dans un premier temps `macroexpand-1`.

   (b) Trouver un cas où le résultat de `macroexpand` est différent de celui de `macroexpand-1`.

2. (a) Ecrire l'opérateur `circ (l)` qui rend la liste `l` circulaire.

   (b) Ecrire l'opérateur `def-sequencer` qui définit la fonction `sequencer` qui énumère ad-infinitum les éléments de la liste `l` rendue circulaire. Par exemple, l'appel `(def-sequencer '(do mi sol))` donnera une fonction `sequencer` telle que

   ```
   (list (sequencer)(sequencer)(sequencer)(sequencer)(sequencer))
                                  =>  (do mi sol do mi)
   ```

   (c) Ecrire `with-sequencer` en s'inspirant des `with` déjà rencontrés ; on aura par exemple :

   ```
   (with-sequencer pom-pom '(sol sol sol mi fa fa fa re)
     (defun la-cinquieme () (pom-pom)))
   ```

   S'inspirer des macros `with-...` pour ecrire l'opérateur `def-sequencer` tel qu'i l soit capable de prendre en argument le nom du `sequencer` (on s'inspirera de `with-hash-table-iterator`).