

Kink

Valvassori Moise

6 octobre 2000

Résumé

Kink est un langage de la grande famille des LISPs. Celui-ci a la particularité d'être à évaluation paresseuse et d'avoir un environnement à portée statique. On a aussi la possibilité de choisir la langue des messages d'erreurs.

Table des matières

1	Introduction	2
2	Installation	2
2.1	Compilation	2
2.2	Internationalisation	2
3	Liste des fonctions	2
3.1	Primitives essentielles	2
3.2	Fonctions sur les listes	3
3.3	Fonctions arithmétiques	3
3.3.1	Calcul	3
3.3.2	Prédicats	4
3.4	Fonctions diverses	4
4	Performances et limites	4
4.1	Bugs connus	4
5	Conclusion	4

1 Introduction

Kink est un Lisp. Il a été réalisé pour l'UV «implémentation de micro langage». Il se veut minimaliste et propre du point de vue fonctionnel. Il ne possède donc pas de fonction à effet de bord comme `set !` ou `setq`.

2 Installation

2.1 Compilation

Dans la plupart des cas, il suffit de taper la séquence suivante :

```
./configure
make
make install
```

Néanmoins, il se peut que dans certain cela ne soit pas suffisant. Alors, si vous voulez changer le répertoire d'installation, tapez :

```
./configure --prefix=\texttt{Mon répertoire}
```

Si cela ne convient toujours pas, lisez le fichier `INSTALL` ou tapez :

```
./configure --help
```

2.2 Internationalisation

Par défaut, la langue de `kink` est l'anglais. Si vous voulez un autre langue (seulement le français pour l'instant), il vous faut compiler `kink` avec la librairie `gettext`. C'est ce qui est fait par défaut. Si vous n'avez pas installé `gettext`, vous pouvez utiliser l'option suivante :

```
configure --with-included-gettext
```

ou bien si vous ne voulez pas l'internationalisation essayez :

```
configure --disable-nls
```

Ensuite, il faut installer le programme. Pour cela, allez voir à la section [2.1](#).

Un fois que tout est installé, il faut choisir sa langue. On fait cela en modifiant la variable d'environnement `LANG` :

```
export LANG=fr
```

Voilà, `kink` parle français maintenant.

3 Liste des fonctions

3.1 Primitives essentielles

`(lambda (formal) body)`

Évalue le corps de `lambda` dans l'environnement de sa première évaluation étendu par les arguments. Si lors de l'évaluation un des paramètres formels est rencontré et si c'est la première fois qu'on le rencontre alors ce dernier est évalué dans l'environnement d'appel, sinon on prend la valeur de sa dernière évaluation. Le résultat de cette évaluation remplace l'ancienne valeur du paramètre.

```
(define a (lambda (a) (+ a 1)))  
(a (a 1)) => 3
```

(define *symbol* *body*)

Associe *body* au symbole *symbol* dans l'environnement courant.

body n'est pas évalué. `define` retourne le *symbol*

(quote *expr*)

(' *expr*)

Retourne *expr* sans l'évaluer

(if *condition* *expr-t* *expr-f*)

Évalue la *condition*. Si celle-ci renvoie #*f* alors on évalue *expr-f* sinon on évalue *expr-t*

3.2 Fonctions sur les listes

(cons *atom* *list*)

Construit la liste *list* avec le *atom* comme premier élément.

(car *list*)

Retourne le premier élément de *list*

(cdr *list*)

Retourne la liste *list* sans le premier élément

(null? *list*)

Retourne #*t* si *list* est vide ; sinon retourne #*f*

(scons *atom* *list*)

Construit un flux.

3.3 Fonctions arithmétiques

3.3.1 Calcul

(+ *a* *b*)

Additionne deux nombres

(- *a* *b*)

Soustrait deux nombres

(* *a* *b*)

Multiplie deux nombres

(/ *a* *b*)

Divise deux nombres. (*a* / *b*)

3.3.2 Prédicats

(= a b)

Renvoie #t si a et b ont la même valeur, sinon renvoie #f

3.4 Fonctions diverses

(quit)

Sort du programme

(version)

Affiche la version courante de Kink

4 Performances et limites

Il faut que j'améliore les points suivants :

- Les environnements sont implémentés en utilisant de simples listes chaînées. Il est clair que c'est moins efficace qu'une table de hachage
- L'arbre syntaxique est très minimal. Il pourrait être utile de le prétraiter. D'autant plus qu'il est stocké tel quel lors d'un `define`
- Il serait beaucoup plus propre de gérer les erreurs par la continuation.

4.1 Bugs connus

Parser

les espaces en début de ligne ne sont pas éliminés

parser

l'opérateur «-» est considéré comme l'entier «-0»

cons

Parcours de `scons` récursif très incertain

5 Conclusion

Cet évaluateur est loin d'être terminé. Notement, je compte lui rajouter un système de bibliothèque dynamique de primitive. De plus, Kink peut relativement facilement être parallélisé.