Langages enchâssés, troisième année

EXAMEN : 2 HEURES

14 mars 2000

Tapez et tester votre solution sur machine. Remettez-le en double : sous forme électronique à `strandh@labri.u-bordeaux.fr` et sous forme papier au surveillant de l'examen. Respectez les règles d'indentation et de formatage.

Donnez des réponses brèves et claires ! Le résultat devrait tenir sur une page imprimée.

1. (10 points) On considère une version simplifiée de la fonction `format` de Lisp ou le `printf` de la bibliothèque C standard. La syntaxe générale est :

```
(printf <string> <arg1> <arg2> ... <argn>)
```

où `<string>` est une expression dont la valeur est une chaîne de caractères, et `<arg1>` ... `<arg`n`>` sont des expressions arbitraires. L'argument `<string>` contient de caractères arbitraires. Un caractère '`%`' dans cette chaine sera remplacer par l'argument `<arg`i`>` correspondant. Exemple :

```
> (printf "I have % things to do in only % days" (* 5 5) (+ 4 5))
I have 25 things to do in only 9 days
NIL
>
```

Souvent (mais pas toujours) le premier argument de `printf` est une chaîne de caractères constante, comme dans l'exemple ci-dessus. Dans ce cas, et si `printf` est une macro on peut générer du code qui évite l'interprétation de la chaîne à l'exécution. L'expansion de l'exemple ci-dessus sera donc :

```
(princ "I have ")
(print (* 5 5))
(princ " things to to in only ")
(print (+ 4 5))
(princ " days")
```

[on ignore le fait que print rajoute un caractère de fin de ligne]

Si l'argument `<string>` n'est pas une chaîne constante, l'expansion doit donner un appel à la fonction `printf-general` comme ceci :

```
> (macroexpand '(printf (f x) 2 3))
(printf-general (f x) 2 3)
```

Écrire la macro `printf`. En plus de l'expansion indiquée, elle doit vérifier (le plus tôt possible) que le nombre d'arguments (le 'n' dans `<arg`n`>`) est égal au nombre de caractères '`%`' dans la chaîne.

2. (10 points) L'exemple de `printf` ci-dessus donne une approche élémentaire au problème de formater selon un modèle des données générées à la volée.

Expliquer comment utiliser ces concepts pour créer des sites Web dynamiques, ou les pages sont fabriquées à la volée par un calcul arbitraire (qui peut faire un choix de langue, des accès à une base de données, etc.) Quel est l'influence de l'approche ci-dessus sur les performances du serveur ?

Un site sera alors décrit dans un langage enchâssé. Donner un aperçu de ce langage. Quel serait l'apport du système CLOS ?