

Devoir de Programmation Fonctionnelle

I) Sujet du devoir

On désire implémenter en Ocaml l'algorithme d'unification entre termes du premier ordre qui a été vu en cours. On implémentera bien l'algorithme du cours (*i.e.* la transformation d'un système d'équations entre termes du premier ordre par le moyen de quatre pas de calcul élémentaires jusqu'à obtention d'un système d'équations sous forme résolue ou d'un échec) et pas un autre.

Pour représenter les termes du premier ordre, on utilisera le type Ocaml suivant :

```
# type terme = X of int | A | B | F of terme | G of terme * terme | H of
terme * terme * terme ;;
```

Un terme est donc soit une variable X_i , soit la constante A, soit la constante B, soit le symbole de fonction F (resp. G, H) ayant pour argument(s) un (resp. deux, trois) terme(s). Une équation sera un couple de tels termes, et un système d'équations une liste de telles équations.

Parmi les nombreuses fonctions que vous ne manquerez pas de définir figureront les suivantes : `forme_resolue` qui prend en entrée un système d'équations, et renvoie un booléen indiquant si ce système est sous forme résolue ; `substitution` qui prend en entrée une variable X_i , un terme t_1 et un terme t_2 , et renvoie le terme t_2 où toutes les occurrences de X_i ont été substituées par t_1 ; `test_occurrence` qui prend en entrée une variable et un terme, et renvoie un booléen indiquant si la variable apparaît dans le terme ; etc.

II) Document attendu

Le devoir sera réalisé en binôme et donnera lieu à un rapport papier qui rendu le mardi 23 mai à 14h. Ce document comprendra :

- Une présentation du problème et de sa solution algorithmique en pseudo-code.
- Une discussion des différents choix d'implémentation ainsi que des problèmes rencontrés.
- Un listing commenté des types et fonctions Ocaml utilisés.
- Des jeux d'essais nombreux et pertinents.

La notation prendra en compte, à parts égales, le contenu (*i.e.* le code) et la forme (*i.e.* le rapport), ce devoir étant autant l'occasion d'apprendre à rédiger un document que celle de programmer en Ocaml.