



TD 4 : Unification

1 Unification

On considère les termes suivants :

1. $p(m\ 3\ X)\ 2$
2. $p(m\ 3\ X)\ Y$
3. $p\ X(m\ 3\ X)$
4. $p\ Y(m\ 3\ Z)$

On suppose que les symboles en minuscule sont des constantes et ceux en majuscule des variables. Parmi ces termes, lesquels sont unifiables ? Déterminez leur *mgu*.

2 Implantation de l'unification

On cherche à implanter l'algorithme d'unification vu en cours.

1. Définissez un type `term` qui correspond au type des termes vu en cours.
2. Représentez les termes de § 1. Utilisez ces termes pour vos tests par la suite.
3. Définissez une fonction `fv` (*free variables*) de type `term -> string list` qui calcule l'ensemble des variables libres d'un terme. (Les ensembles seront implantés comme des listes sans doublons.)
4. Définissez une fonction `subst_assoc` qui prend un terme et une substitution et applique la substitution au terme. Il est préférable de représenter des substitutions comme listes d'associations (voir TD 3).
5. Pour implanter l'algorithme d'unification, choisissez d'abord une représentation pour E et S dans l'algorithme d'unification.
6. Définissez les fonctions `subst_in_term_pair_list` (pour $E[x \leftarrow t]$) et `subst_in_subst` (pour $S[x \leftarrow t]$).
7. Définissez maintenant la fonction `unif` qui implante l'algorithme d'unification. Le *fail* de l'algorithme d'unification sera implanté par une exception de Caml (`failwith`).