

Examen

Le langage Caml

Exercice 1. On considère la fonction `merge` suivante :

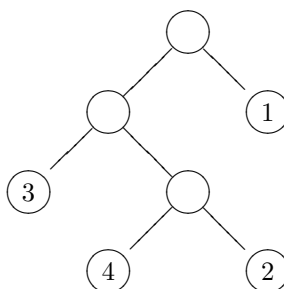
```
merge.ml
1 : let rec merge comparaison = fonction
2 :   ([],l2) -> l2
3 :   | (l1,[]) -> l1
4 :   | (t1::q1,t2::q2) ->
5 :     if (comparaison t1 t2) then
6 :       t1::(merge comparaison (q1,t2::q2))
7 :     else
8 :       t2::(merge comparaison (t1::q1,q2))
9 :   ;;
```

1. Écrire le résultat de l'évaluation des expressions suivantes :
`merge (<=) ([2;3;6;4], [1;3;5;6])`;;
`merge (>) ([2;3;6;4], [1;3;5;6])`;;
2. Expliquer ce que calcule cette fonction, en général.
3. Écrire le type de cette fonction.

Exercice 2. On considère le type récursif `arbre` suivant :

```
type arbre = Feuille of int | Noeud of arbre*arbre;;
```

1. Écrire une expression Caml, ayant pour type `arbre`, qui représente l'arbre ci-dessous :



Produire une représentation graphique de l'arbre correspondant à l'expression Caml

```
Noeud(Feuille 3, Noeud(Feuille 4, Noeud(Feuille 1,Feuille 2))));;
```

2. Écrire trois fonctions Caml qui calculent :
 - (a) le nombre de noeuds internes¹ d'un arbre passé en paramètre.
 - (b) la somme des entiers qui étiquettent les feuilles de l'arbre passé en paramètre,
 - (c) la profondeur² d'un arbre passé en paramètre.

¹Les noeuds qui ne sont pas des feuilles.

²La profondeur d'un arbre est la longueur maximum d'une branche de l'arbre.

Unification et résolution

Exercice 3. Appliquer l'algorithme d'unification, tel que présenté en cours, pour résoudre les problèmes d'unification suivants :

1. $(f(x), y), (g(y), z), (y, f(w)),$
2. $(g(x), g(f(y))), (z, h(y, x)), (y, z),$
3. $(f(x), f(g(y))), (x, h(y)).$

Pour chaque exemple, on s'efforcera de clarifier le déroulement de l'algorithme.

Exercice 4.

1. On peut appliquer la règle de factorisation à droite à la clause suivante :

$$Q(x, y) \Rightarrow P(f(x), y), P(x, g(z)), P(f(w), y)$$

Combien y a-t-il de façons d'appliquer cette règle? Justifier votre réponse et calculer les résultats.

2. On peut appliquer la règle de résolution au couple de clauses suivant :

$$P(x) \Rightarrow Q(f(x)) \quad Q(y) \Rightarrow P(g(y))$$

Combien y a-t-il de façons d'appliquer cette règle? Justifier votre réponse et calculer les résultats.

Exercice 5. La sémantique du prédicat $P(x, y)$ est : « le nombre entier x est strictement plus petit que le nombre entier y ». À partir de la théorie composée par les deux clauses

$$P(0, s(0)), \quad P(x, y) \Rightarrow P(x, s(y)),$$

utiliser le calcul de la résolution pour en déduire que 0 est strictement plus petit que le nombre entier 3, qui est représenté par le terme $s(s(s(0)))$.

Exercice 6. Proposer une représentation des termes et des substitutions en Caml. Écrire une fonction `substituer`, prenant en paramètre un terme et une substitution, qui calcule le résultat de l'application de la substitution au terme.

Sémantique

Exercice 7.

1. Énoncer le théorème de point fixe de Tarski.
2. Définir la sémantique dénotationnelle d'un commande de la forme

`while b do c`

appartenant à la classe syntaxique `Com` du langage IML.