

Série d'exercices #3 Implantation de la récursion

1. Traduisez, une fois à l'aide de la méthode "trampoline" et une fois à l'aide de la méthode des blocs d'activation explicites, le bout de programme suivant. Pour fins de simplification, ignorez la questions des variables capturées par les fonctions (il n'y en a pas, ici) et considérez que les opérations mathématiques sont faites à l'aide d'opérateurs arithmétiques (et non à l'aide d'appels à des fonctions de librairie).

```
(define (f x y)
  (if (= x 0)
      (- y 2)
      (g (- x 1) (+ (* 2 y) 1))))
(define (g a b)
  (if (= (modulo a 2) 1)
      (g (- a 1) (+ b 2))
      (f a (h b))))
(define (h t)
  (- (* 3 t) 1))
```

2. Faites la conversion en forme CPS de cette expression :

if #1 n then g(h(n)) else $\lambda x.n$

3. Donnez la forme générale de la conversion en forme CPS de l'expression suivante :

```
let x1 = if y1 then 1 else 2 in
let x2 = if y2 then 1 else 2 in
...
let xn = if yn then 1 else 2 in
3
```

Donnez la taille (asymptotiquement) de l'expression résultante.

4. Soit e une expression écrite suivant la syntaxe du langage utilisé dans la présentation en classe de la transformation en forme CPS. Supposez que \mathcal{K}_0 est une variable. Démontrez que $\text{CPS_main}[[e]]$ ne comporte aucun appel non-terminal.

Indice : démontrez d'abord que $\text{CPS}[[e]][[e']]$ ne comporte aucun appel non-terminal, en faisant la preuve par induction sur la taille de e et en supposant que e' ne comporte aucun appel non-terminal, et utilisez ensuite ce résultat pour démontrer l'énoncé principal.