USTHB - Faculté d'Informatique
1ère Année MI 2021/2022 – Module OPM
Date : 29/05/2022 <08h30-10h00>

Examen final

Exercice 1 (7 points = 2+2+(1,5+1,5)) (Fonctions récursives)

1) Utiliser la règle de récursion pour montrer que la fonction $mult3$ est PR :
   $mult3 = \lambda xyz.x * y * z$
2) Utiliser la règle de composition pour montrer que la fonction SommeCarre est PR :
   SommeCarre = $\lambda xy.x^2+y^2$
3)
   a) Montrer par récurrence sur l'entier k≥1, que la fonction $S1_k$ est PR, pour tout k :
   $$S1_k = \boldsymbol{\lambda x_1 \dots x_k}.\sum_{i=1}^{k} x_i = \boldsymbol{\lambda x_1 \dots x_k}.x_1 + x_2 + \cdots + x_k$$
   b) En déduire, par récurrence sur l'entier k≥1, que la fonction $S2_k$ est PR, pour tout k :
   $$S2_k = \boldsymbol{\lambda x_1 \dots x_k}.\sum_{i=1}^{k} S1_i(x_1, \dots, x_i) = \boldsymbol{\lambda x_1 \dots x_k}.x_1 + (x_1 + x_2) + \cdots + (x_1 + \cdots + x_k)$$

Remarque : Les fonctions suivantes,vues en cours, sont considérées PR :
$\oplus$, $\otimes$, Moins, Fact, Abs, Exp, Sg, $\overline{Sg}$, Pred, mod, div, les fonctions constantes d'arité 0, les fonctions constantes d'arité 1.

Exercice 2 (6 points = 2+1+1+2) (Machines de Turing)

On considère la machine de Turing $MT = (S, E, I)$, avec :

$S = \{0, 1, *\}$, $E = \{q_0, q_1, q_2, q_3, q_4, q_f\}$,

$I = \{$

(1) $q_0 1\ Dq_0$,	(5) $q_1 0\ Dq_f$,	(9) $q_3 *\ 0q_2$,
(2) $q_0 *\ Dq_0$,	(6) $q_1 *\ 0q_2$,	(10) $q_3 0\ Dq_4$,
(3) $q_0 0\ Gq_1$,	(7) $q_2 0\ Gq_3$,	(11) $q_4 0\ Dq_4$,
(4) $q_1 1\ Gq_1$,	(8) $q_3 1\ 0q_2$,	(12) $q_4 1\ 1q_f$ $\}$

1- Dérouler cette machine pour : **a)** x=2, **b)** $(x_1, x_2) = (1{,}2)$, **c)** $(x_1, x_2, x_3) = (1{,}2{,}3)$.
2- Déduire toutes les fonctions calculables par cette machine.
3- Ecrire la machine de Turing MT_Sg qui calcule la fonction $Sg = \boldsymbol{\lambda x}.\begin{cases} 0 & si\ x = 0 \\ 1 & sinon \end{cases}$
4- En déduire la machine de Turing MT_f qui calcule la fonction $f = \boldsymbol{\lambda xy}.Sg(x) * y$

Exercice 3 (7 points = 1+2+(1,5+1,5+1)) (Langage CAML)

1- Ecrire une fonction **nbElem** calculant le nombre d'éléments d'une liste.
2- En utilisant la fonction nbElem, écrire une fonction récursive, notée **recupElem x n**, qui calcule la sous-liste de x composée des n premiers éléments de x, avec 0 ≤ n ≤ nbElm x.
Exemples : a) recupElem [1; 2; 6; 4] 3 = [1; 2; 6], **b)** recupElem [1; 2; 6; 4] 0 = []
3- On considère la fonction suivante, notée **verif**, utilisant les 2 fonctions précédentes :

```
let rec verif x y = if nbElem x < nbElem y then false
                    else if recupElem x (nbElem y)=y then true else verif (tl x) y ;;
```

a) Dérouler cette fonction pour : **i)** x = [1; 4;1;5;6] et y = [1;5;6], **ii)** x = [1; 4;1;5;6] et y = [1;6]
b) Que fait la fonction **verif** ?

Bon courage

Corrigé OPM 2021/2022

Behloul et Isli

Exo 1

1) $mult3(x, y, 0) = 0 = Z_1 \circ P_1^2(x, y)$ PR

$mult3(x, y, z+1) = xyz + xy$

$= \underbrace{\oplus \circ (P_3^4, \otimes \circ (P_1^4, P_2^4))}_{PR}(x, y, mult3(x,y,z), z)$

2) $SommeCarre(x, y) = x^2 + y^2$

$= \underbrace{\oplus \circ (\otimes \circ (P_1^2, P_1^2), \otimes \circ (P_2^2, P_2^2))}_{PR}(x, y)$

3) $S1_1(x_1) = x_1 = P_1^1(x_1)$ PR

Supposons que $S1_k$ est PR

$S1_{k+1}(x_1, \dots, x_k, x_{k+1}) = S1_k(x_1, \dots, x_k) + x_{k+1}$

$= \underbrace{\oplus \circ (S1_k \circ (P_1^{k+1}, \dots, P_k^{k+1}), P_{k+1}^{k+1})}_{PR}(x_1, \dots, x_k, x_{k+1})$

Conclusion $S1_k$ est PR pour tout $k \geq 1$

3) b) $S2_1(x_1) = x_1 = P_1^1(x_1)$ PR

Supposons que $S2_k$ est PR

$$S2_{k+1}(x_1, \dots, x_k, x_{k+1}) = S2_k(x_1, \dots, x_k) + S1_{k+1}(x_1, \dots, x_k, x_{k+1})$$

$$= \underbrace{\oplus \circ \left(S2_k \circ (P_1^{k+1}, \dots, P_k^{k+1}), S1_{k+1}\right)}_{PR}(x_1, \dots, x_k, x_{k+1})$$

Conclusion : $S2_k$ est PR pour tout $k \geq 1$

Exo 2 :

1- Déroulements

a) $x = 2$:

$$q_0 \underline{2} = q_0 1^3 \xrightarrow{3\times(1)} 1^3 q_0 0 \xrightarrow{(3)} 1^2 q_1 1 \xrightarrow{3\times(4)} q_1 0 1^3$$

$$\xrightarrow{(5)} q_f 1^3 = q_f \underline{2}$$

b) $(x_1, x_2) = (1, 2)$:

$$q_0 \underline{(1,2)} = q_0 1^2 * 1^3 \xrightarrow{2\times(1)} 1^2 q_0 * 1^3 \xrightarrow{(2)} 1^2 * q_0 1^3$$

$$\xrightarrow{3\times(1)} 1^2 * 1^3 q_0 0 \xrightarrow{(3)} 1^2 * 1^2 q_1 1 \xrightarrow{3\times(4)} 1^2 q_1 * 1^3$$

$$\xrightarrow{(6)} 1^2 q_2 0 1^3 \xrightarrow{2\times[(7)\text{ puis }(8)]} q_2 0^3 1^3 \xrightarrow{(7)} q_3 0^4 1^3$$

$$\xrightarrow{(10)} q_4 0^3 1^3 \xrightarrow{3\times(11)} q_4 1^3 \xrightarrow{(12)} q_f 1^3 = q_f \underline{2}$$

c) $(x_1, x_2, x_3) = (1, 2, 3)$:

$$q_0 \underline{(1,2,3)} = q_0 1^2 * 1^3 * 1^4 \xrightarrow{11\times[(1)\text{ ou }(2)]} 1^2 * 1^3 * 1^4 q_0 0$$

$$\xrightarrow{(3)} 1^2 * 1^3 * 1^3 q_1 1 \xrightarrow{4\times(4)} 1^2 * 1^3 q_1 * 1^4 \xrightarrow{(6)} 1^2 * 1^3 q_2 0 1^4$$

$$\xrightarrow{3\times[(7)\text{ puis }(8)]} 1^2 * q_2 0^4 1^4 \xrightarrow{(7)} 1^2 q_3 * 0^4 1^4 \xrightarrow{(9)} 1^2 q_2 0^5 1^4$$

$$\xrightarrow{2\times[(7)\text{ puis }(8)]} q_2 0^7 1^4 \xrightarrow{(7)} q_3 0^8 1^4 \xrightarrow{(10)} q_4 0^7 1^4$$

$$\xrightarrow{7\times(11)} q_4 1^4 \xrightarrow{(12)} q_f 1^4 = q_f \underline{3}$$

2- La machine de Turing calcule toutes les fonctions P_n^n, avec $n \geq 1$.

3- $MT\_Sg = (S, E, I)$

$S = \{0, 1\}$

$E = \{q_0, q_1, q_2, q_3, q_4, q_5, q_f\}$

$$\begin{array}{l} \qquad\ (1) \qquad\qquad (2) \qquad (x=0) \qquad (3) \\ I = \{ q_0 1 0 q_1 - q_1 0 D q_2 \longrightarrow q_2 0 1 q_f \end{array}$$

$(x \geq 1)$ | (4) $q_2 1 0 q_3$ — (5) $q_3 0 D q_4$ — (6) $q_4 1 0 q_3$

| (7) $q_4 0 1 q_5$

| (8) $q_5 1 G q_5$ — (9) $q_5 0 1 q_f \}$

4- $MT\_f = (S, E, I)$

$S = \{0, 1, *\}$

$E = \{q_0, q_1, \dots, q_7, q_f\}$

(5) $q_4 0 1 q_f$

|

$$\begin{array}{l} \qquad\ (1) \qquad\qquad (2) \qquad (x=0) \qquad (3) \qquad\qquad (4) \\ I = \{ q_0 1 0 q_1 - q_1 0 D q_2 \longrightarrow q_2 * 0 q_3 - q_3 0 D q_4 \end{array}$$

| (6) $q_4 1 0 q_3$

$(x \geq 1)$ | (7) $q_2 1 0 q_5$

| (8) $q_5 0 D q_6$ — (9) $q_6 * 0 q_7$ — (10) $q_7 0 D q_f$

| (11) $q_6 1 0 q_5 \}$

4/5

Exo 3 :

1-

let rec **nbElem** l = if l= [] then 0 else 1+**nbElem** (tl l);;

nbElem : 'a list -> int = <fun>

2-

let rec **recupElem** x n = if n=0 then [] else hd x :: **recupElem** (tl x) (n-1);;

recupElem : 'a list -> int -> 'a list = <fun>

3-

a)

i) **verif** [1;4;1;5;6] [1;5;6] = **verif** [4;1;5;6] [1;5;6] = **verif** [1;5;6] [1;5;6] = true

ii) **verif** [1;4;1;5;6] [1;6] = **verif** [4;1;5;6] [1;6] = **verif** [1;5;6] [1;6] = **verif** [5;6] [1;6] = **verif** [6] [1;6] = false

b)

La function **verif** vérifie si y est sous-liste de x.

5/5