Examen de remplacement

[Interrogation de remplacement : Exo 1 1){3pts} 2){3pts} ; Exo 2 1a){1,5pt} 1b){1,5pt} 4){5pt} ; Exo 3 1){1pt+1pt} 2){4pt}]

Exercice 1 **(6 points = 2+2+2) (Fonctions récursives)**

1) Utiliser la règle de récursion pour montrer que la fonction f est PR :
   $f = \lambda xyz.x * y + Exp(x,z)$
2) Utiliser la règle de composition pour montrer que la fonction g est PR :
   g = $\lambda xy.x^2$+Exp(x,y+1)
3) Montrer par récurrence sur l'entier k≥1, que la fonction S_k est PR, pour tout k :

$$S_k = \boldsymbol{\lambda x_1 \ldots x_k}.\sum_{i=1}^{k} i * x_i = \boldsymbol{\lambda x_1 \ldots x_k}.x_1 + 2x_2 + \cdots + kx_k$$

Remarque : Les fonctions suivantes, vues en cours, sont considérées PR :
⊕, ⊗, Moins, Fact, Abs, Exp, Sg, $\overline{Sg}$, Pred, mod, div, les fonctions constantes C_k=λ.k d'arité 0, les fonctions constantes D_k=λn.k d'arité 1.

Exercice 2 **(8 points = (0,5+0,5+1)+1,5+2+2,5) (Machines de Turing)**

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

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

$I = \{$

(1) $\boldsymbol{q_0 1 \;\; D q_0}$,	(5) $\boldsymbol{q_1 0 \;\; D q_f}$,	(9) $\boldsymbol{q_3 * \;\; G q_3}$,
(2) $\boldsymbol{q_0 * \;\; D q_0}$,	(6) $\boldsymbol{q_1 * \;\; 1 q_2}$,	(10) $\boldsymbol{q_3 0 \;\; 1 q_f}\}$
(3) $\boldsymbol{q_0 0 \;\; G q_1}$,	(7) $\boldsymbol{q_2 1 \;\; G q_3}$,	
(4) $\boldsymbol{q_1 1 \;\; G q_1}$,	(8) $\boldsymbol{q_3 1 \;\; G q_3}$,	

1) Dérouler cette machine pour : **a)** x=1, **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_f qui calcule la fonction $f = \boldsymbol{\lambda x}.\begin{cases} x+1 & si\ x\ pair, \\ 1 & sinon \end{cases}$
4) Ecrire la machine de Turing MT_g qui calcule la fonction $g = \boldsymbol{\lambda xy}.\begin{cases} x\ si\ y\ pair, \\ y \quad sinon \end{cases}$

Exercice 3 **(6 points = (0,5+0,5)+2+1+2) (Langage CAML)**

1) Donner les types inférés par CAML pour chacune des deux fonctions suivantes :
   let sc x i j = sub_string x i j ;;
   let lg x = string_length x ;;
2) Utiliser les deux fonctions **sc** et **lg** pour écrire une fonction récursive, notée **sousChaine x y**, qui teste si la chaine de caractères x est sous-chaine de la chaine de caractères y.
   Exemples : a) sousChaine '''' ''ab'' = true, **b)** sousChaine ''bc'' ''abcd'' = true, **c)** sousChaine ''bd'' ''abcde'' = false , **d)** sousChaine ''abc'' ''ab'' = false
3) Ecrire une fonction **suppElem** supprimant un élément à la fin d'une liste non vide.
4) Ecrire une fonction récursive, notée **nbEP x**, qui compte le nombre d'entiers pairs dans une liste x d'entiers.

Bon courage

USTHB - Faculté d'Informatique
1ère Année MI 2021/2022 Section 6 – Module OPM
Date : 09/06/2022 <13h00-14h30>
Responsable du module : Mr Isli

Corrigé

Exercice 1

1) f d'arité 3 : trouver deux fonctions G et H d'arités 2 et 4, respectivement, telles que, pour tous x, y, z :
   - f(x,y,0)=G(x,y)
   - f(x,y,z+1)=H(x,y,f(x,y,z),z)
   - G et H PR

   f(x,y,0)=x*y+Exp(x,0)=xy+1=S(xy)=[S°⊗](x,y)
   f(x,y,z+1)=xy+x*Exp(x,z)=[⊕°(⊗°(P_1^4,P_2^4),⊗°(P_1^4,Exp°(P_1^4,P_4^4)))](x,y,f(x,y,z),z)
   Dérivation PR triviale.
2) g(x,y)=[⊕°(⊗°(P_1^2,P_1^2),Exp°(P_1^2,$S°P_2^2$))](x,y)
   Dérivation PR triviale.
3)
   a. Vérification pour k=1 : $S_1=P_1^1$ PR car fonction de base.
   b. Hypothèse de récurrence : propriété vraie jusqu'à l'ordre n≥1 : pour tout k≤n, S_k est PR.
   c. Montrons que la propriété reste alors vraie pour k=n+1 :
   $S_{n+1}(x_1,\ldots,x_{n+1})=S_n(x_1,\ldots,x_n)+(n+1)x_{n+1}$
   $=[\oplus°(S_n°(P_1^{n+1},\ldots,P_n^{n+1}),\otimes°(D_{n+1}°P_1^{n+1},P_{n+1}^{n+1}))](x_1,\ldots,x_{n+1})$
   La propriété reste bien vraie pour k=n+1 (règle de composition : dérivation PR triviale) : elle est donc vraie pour tout k≥1.

Exercice 2

1) Déroulements triviaux (laissés au lecteur).
2) La machine de Turing calcule la fonction P_1^1, la fonction $\lambda xy.x+y+3$, et toutes les fonctions f_n suivantes de IN^n dans IN^{n-1}, avec n≥3 : $f_n(x_1, \ldots ,x_n) = (y_1, \ldots ,y_{n-1})$, avec :
   - $y_1=x_1+1$
   - $y_i=x_i$, pour tout i dans {2, …, n-2}
   - $y_{n-1}=x_{n-1}+x_n+2$
3) $S=\{0,1\}$, $E=\{q_0,q_1,\ldots,q_6,q_f\}$,
   $I=\{$

(1) $q_0 1\ Dq_1$,	(5) $q_3 1\ Gq_3$,	(9) $q_5 0\ Gq_4$,
(2) $q_1 1\ Dq_2$,	(6) $q_3 0\ Dq_f$,	(10) $q_4 0\ 1q_5$,
(3) $q_2 1\ Dq_1$,	(7) $q_2 0\ Gq_4$,	(11) $q_5 1\ Gq_6$,
(4) $q_1 0\ 1q_3$,	(8) $q_4 1\ 0q_5$,	(12) $q_6 0\ 1q_f$ }

Les instructions (1), (2) et (3) testent la parité de l'entrée x. Les instructions (4) à (6) traitent le cas où x est paire (sortie=x+1). Les instructions (7) à (12) traitent le cas où x est impaire (sortie=1).

4) $S = \{0, 1, *\}, E = \{q_0, q_1, \dots, q_{10}, q_f\}$,
$I = \{$

$(1)\ \boldsymbol{q_0 1\ Dq_0}$,
$(2)\boldsymbol{q_0 *\ Dq_1}$,
$(3)\ \boldsymbol{q_1 1\ Dq_2}$,
$(4)\ \boldsymbol{q_2 1\ Dq_1}$,
$(5)\ \boldsymbol{q_2 0\ Gq_3}$,
$(6)\ \boldsymbol{q_3 1\ 0q_4}$,
$(7)\ \boldsymbol{q_4 0\ Gq_3}$,
$(8)\boldsymbol{q_3 *\ 0q_5}$,
$(9)\ \boldsymbol{q_5 0\ Gq_6}$,
$(10)\ \boldsymbol{q_6 1\ Gq_6}$,
$(11)\ \boldsymbol{q_6 0\ Dq_f}$,
$(12)\ \boldsymbol{q_1 0\ Gq_7}$,
$(13)\ \boldsymbol{q_7 1\ Gq_7}$,
$(14)\ \boldsymbol{q_7 *\ 0q_8}$,
$(15)\ \boldsymbol{q_8\ 0\ Gq_9}$,
$(16)\ \boldsymbol{q_9 1\ 0q_8\}}$
$(17)\ \boldsymbol{q_9 0\ Dq_{10}}$,
$(18)\ \boldsymbol{q_{10}\ 0\ Dq_{10}}$,
$(19)\ \boldsymbol{q_{10} 1\ 1q_f\}}$

Les instructions (1) à (4) testent la parité de y. Les instructions (5) à (11) traitent le cas où y est pair (sortie=x). Les instructions (12) à (19) traitent le cas où y est impair (sortie=y).

Exercice 3

1) sc : string -> int -> int -> string = <fun>
   lg : string -> int = <fun>
2) let rec sousChaine x y = if lg x > lg y then false else if x= sc y 0 (lg x) then true else sousChaine x (sc y 1 (lg y-1)) ;;
   sousChaine : string -> string -> bool =<fun>
3) let rec suppElem x = if tl x = [] then [] else hd x :: suppElem (tl x) ;;
   suppElem : 'a list -> 'a list = <fun>
4) let rec nbEP x = if x = [] then 0 else if (hd x) mod 2 = 0 then 1+nbEP (tl x) else nbEP (tl x) ;;
   nbEP : int list -> int = <fun>