USTHB - Faculté d'Informatique
1ère Année I, M et MI 2022/2023 – Module OPM
Date : 18/06/2023 <08h30-10h>

Rattrapage

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

1) Montrer par récursion et par composition que f(x)=2x est primitive récursive.
2) Montrer que l'ensemble des entiers pairs est primitif récursif.
3) Montrer que $F_n(x_1,x_2,\dots,x_n)=(Fact(x_1)+Fact(x_2)+\dots+Fact(x_n))^2$ est primitive récursive pour tout entier n>0.

Indication : Les fonctions suivantes sont PR : ⊕, ⊗, Moins, Fact, Abs, Exp, Sg, $\overline{Sg}$, Pred, mod, div, les fonctions constantes d'arité 0 ou 1.

Exercice 2 (7 points = (((1,5+1,5)+1,5) +2,5) (Machines de Turing)

1) Soit la Machine de Turing MT=<E,S,I> avec :
   I={ (1) $\mathbf{q_0 1 0 q_1}$, (2) $\mathbf{q_1 0 D q_0}$, (3) $\mathbf{q_0 {*} 0 q_2}$, (4) $\mathbf{q_2 0 D q_3}$, (5) $\mathbf{q_3 1 D q_3}$, (6) $\mathbf{q_3 0 G q_4}$, (7) $\mathbf{q_4 1 G q_4}$, (8) $\mathbf{q_4 0 D q_f}$, (9) $\mathbf{q_3 {*}{*} q_6}$, (10) $\mathbf{q_6 {*} 0 q_5}$, (11) $\mathbf{q_5 0 D q_6}$, (12) $\mathbf{q_6 1 0 q_5}$, (13) $\mathbf{q_6 0 G q_7}$, (14) $\mathbf{q_7 0 G q_7}$, (15) $\mathbf{q_7 1 G q_4}$}
   a) Donner les configurations finales après exécution des instructions de MT en partant des configurations initiales suivantes :

Configuration Initiale	q_0 $\underline{4}$ * $\underline{2}$	q_0 $\underline{2}$ * $\underline{3}$ * $\underline{1}$
Configuration Finale	q_f $\underline{?}$	q_f $\underline{?}$

   b) Quelles sont toutes les fonctions calculées par MT ?
2) Définir les instructions d'une Machine de Turing qui calcule la fonction f suivante :
   f(x,y)= Sg(x*y)

Exercice 3 (7 points = (1+(1,5+0,5)+(1,5+0,5) +2) (Langage CAML)

1) On considère les instructions CAML suivantes :

```
let rec mystere1 e L = if L = [] then []
                        else if e = hd L then mystere1 e (tl L)
                             else hd L :: mystere1 e (tl L) ;;
let rec mystere2 L = if L = [] then [] else hd L :: mystere2 (mystere1 (hd L) L) ;;
```

   a) Donner, pour chacune des deux fonctions, le type inféré par CAML.
   b) Dérouler la fonction mystere1 pour e=5 et L= [3;5;7;8;3;5;4;5;7]. Que fait mystere1 ?
   c) Dérouler la fonction mystere2 pour L= [3;5;7;8;3;5;4;5;7]. Que fait mystere2 ?
2) Ecrire une fonction CAML **suppVoyMin** supprimant les voyelles minuscules non accentuées d'une chaine de caractères.

Exemples : suppVoyMin "" = "" ; suppVoyMin "USTHB" = "USTHB" ; suppVoyMin "Extraire" = "Extrr" ; suppVoyMin "excès" = "xcès"

Indication : les voyelles minuscules non accentuées sont tous les caractères de la chaine "aeiouy"

…Bon courage…

Corrigé Rattrapage OPM (18/6/2023)

Isli et Behloul

Exercice 1

1)i) par récursion 1

$f(0) = 0 = Z$: PR

$f(y+1) = 2y+2 = S \circ S \circ (f(y)) = S \circ S \circ P_1^2(f(y),y)$: PR

ii) par composition $f(x) = 2x = \oplus \circ (P_1^1, P_1^1)(x)$: PR 1

2) notons par E l'ensemble des entiers pairs

notons par $r2 = \lambda x.x \bmod 2$: PR

soit

$$Car_E(x) = Car_{pairs}(x) = \begin{cases} 1 \text{ si } x \text{ est pair} \\ 0 \text{ si } x \text{ est impair} \end{cases}$$ 2

$$= \begin{cases} 1 \text{ si } r2(x) = 0 \\ 0 \text{ si } r2(x) = 1 \end{cases} = \overline{Sg}(r2(x)) \text{ PR}$$

donc $Car_E = \overline{Sg} \circ r2$: PR

alors E est un ensemble PR

3) notons d'abord $carre(x) = x^2 = \otimes \circ (P_1^1, P_1^1)(x)$: PR

et $G_n(x_1, \ldots, x_n) = Fact(x_1) + \ldots + Fact(x_n)$ 1

on a $F_n(x_1, \ldots, x_n) = carre \circ G_n(x_1, \ldots, x_n)$

il suffit alors de montrer par récurrence sur n que G_n est PR pour tout entier $n > 0$.

-Pour $n = 1$, $G_1(x_1) = Fact(x_1)$: PR

-Supoosons que G_n est PR

on a $G_{n+1}(x_1, \ldots, x_n, x_{n+1}) = Fact(x_1) + \ldots + Fact(x_n) + Fact(x_{n+1})$ 1

$G_{n+1}(x_1, \ldots, x_n, x_{n+1}) = G_n(x_1, \ldots, x_n) + Fact(x_{n+1})$

$G_{n+1}(x_1, \ldots, x_n, x_{n+1}) = \oplus \circ (G_n \circ (P_1^{n+1}, \ldots, P_n^{n+1}), Fact \circ P_{n+1}^{n+1})(x_1, \ldots, x_n, x_{n+1})$: PR

-conclusion : G_n est PR pour tout entier $n > 0$.

Exo 2 :

1/ a/ $q_0 \underline{4} * \underline{2} = q_0 |||||*||| \xrightarrow{[(1) \text{ puis } (2)]^+} q_0 *||| \xrightarrow{(3) \text{ puis } (4)} q_3 |||$ (1,5)

$\xrightarrow{[(5)]^+} |||q_3 0 \xrightarrow{(6) \text{ puis } [(7)]^+} q_4 0||| \xrightarrow{(8)} q_f ||| = q_f \underline{2}$

$q_0 \underline{2} * \underline{3} * \underline{1} = q_0 |||*||||*|| \xrightarrow{[(1) \text{ puis } (2)]^+} q_0 *||||*||$

$\xrightarrow{(3) \text{ puis } (4)} q_3 ||||*||| \xrightarrow{[(5)]^+} ||||q_3 *||| \xrightarrow{(9) \text{ puis } (10)} ||||q_5 0|||$

$\xrightarrow{[(11) \text{ puis } (12)]^+} |||0000 q_6 0 \xrightarrow{(13) \text{ puis } [(14)]^+} |||q_7 |$

$\xrightarrow{(15) \text{ puis } [(7)]^+} q_4 0|||| \xrightarrow{(8)} q_f |||| = q_f \underline{3}$ (1,5)

b) Les fonctions calculés par MT sont P^n_2 (1), $n \geq 2$. La MT n'est pas définie pour $n = 1$ (0,5)

2/ Instructions d'une machine de Turing calculant

$f = \lambda xy.\ Sg(x * y)$

(1) $q_0 1 D q_1 \xrightarrow{x=0} q_1 * D q_8$ (14) $\rightarrow q_8 1 D q_8$ (16) $\rightarrow q_8 0 G q_9$ (17) $\rightarrow q_9 1 0 q_8$ (18)

$\rightarrow q_9 * 0 q_8$ (19)

$\rightarrow q_9 0 1 q_f$ (20)

(2) $q_1 1 D q_2$

(3) $q_2 1 D q_2$

(4) $q_2 * D q_3$

(5) $q_3 1 D q_4 \xrightarrow{y=0} q_4 0 0 q_8$ (15) $\rightarrow q_8 0 G q_9$

(6) $q_4 1 D q_5$

$\rightarrow q_6 * 0 q_5$ (9)

(7) $q_5 1 D q_5 \rightarrow q_5 0 G q_6$ (8) $\rightarrow q_6 1 0 q_5$ (10) $\rightarrow q_6 0 1 q_7$ (11) $\rightarrow q_7 1 G q_7$ (12) $\rightarrow q_7 0 1 q_f$ (13)

(2,5)

Exo 3 :

1) a) mystere1 : 'a → 'a list → 'a list = <fun> (0,5)

mystere2 : 'a list → 'a list = <fun> (0,5)

b) mystere1 5 [3;5;7;8;3;5;4;5;7] =

3 :: mystere1 5 [5;7;8;3;5;4;5;7] =

3 :: mystere1 5 [7;8;3;5;4;5;7] =

3 :: 7 :: mystere1 5 [8;3;5;4;5;7] =

3 :: 7 :: 8 :: mystere1 5 [3;5;4;5;7] =

3 :: 7 :: 8 :: 3 :: mystere1 5 [5;4;5;7] =

3 :: 7 :: 8 :: 3 :: mystere1 5 [4;5;7] =

3 :: 7 :: 8 :: 3 :: 4 :: mystere1 5 [5;7] =

3 :: 7 :: 8 :: 3 :: 4 :: mystere1 5 [7] =

3 :: 7 :: 8 :: 3 :: 4 :: 7 :: mystere1 5 [] =

3 :: 7 :: 8 :: 3 :: 4 :: 7 :: [] =

[3; 7; 8; 3; 4; 7] (1,5)

(0,5)

La fonction mystere1 supprime toutes les occurrences d'un élément e dans une liste L.

c) mystere2 [3;5;7;8;3;5;4;5;7] =

3 :: mystere2 (mystere1 3 [3;5;7;8;3;5;4;5;7]) =

3 :: mystere2 [5;7;8;5;4;5;7] =

3 :: 5 :: mystere2 (mystere1 5 [5;7;8;5;4;5;7])

= 3 :: 5 :: mystere2 [7;8;4;7]

= 3 :: 5 :: 7 :: mystere2 [8;4]

= 3 :: 5 :: 7 :: 8 :: mystere2 [4]

= ______ 4 :: mystere2 []

= ______ [] = [3;5;7;8;4]

(1,5)

Exo 3 (suite) :

La fonction mystere2 supprime les redondances dans une liste. ~~Pour~~ chaque élément de la liste en entrée, seulement la 1ère occurrence est gardée. 0,5

2)

```
let rec occurre c cc =
    if cc = "" then false
    else if c = cc.[0] then true
    else occurre c (substring cc 1
                     (string_length cc - 1));;
occurre : char -> string -> bool = <fun>
```

1

```
let rec suppVoy cc = if cc = "" then "" else
  let sc = substring cc 1 (string_length cc - 1) in
  if occurre cc.[0] "aeiouy" then
        suppVoy sc
  else substring cc 0 1 ^ suppVoy sc;;
suppVoy : string -> string = <fun>
```

1