

Support de travaux dirigés/pratiques - LOTOS

Mars 2008

Structure d'un processus

```
PROCESS nom [liste des actions ou portes] (<liste param>) : <fonctionnalité> :=  
    <expression de comportement>  
WHERE  
    <définitions de types>  
    <définitions de processus>  
ENDPROC
```

<fonctionnalité> est

- soit NOEXIT pour un comportement qui ne s'arrête pas;
- soit EXIT pour un comportement qui s'arrête.

<liste param> est de la forme act1 :TYPE1, act2 :TYPE2, act3 :TYPE3, ...

<expression de comportement> est une expression combinant les actions de l'alphabet <liste param> avec les opérateurs de LOTOS (voir tableau 1).

<définitions de types> est de la forme :

```
library Boolean endlib    (* un type prédéfini *)  
library Natural endlib  
library monType endlib   (* un nouveau type défini dans un fichier *)
```

<définitions de processus> est de la forme de l'exemple suivant :

```
process Correcteur [accr, libr, ecrire] : exit :=  
    accr ; ecrire ; libr ; Correcteur[accr, libr, ecrire]  
    [] i ; exit  
endproc
```

ou encore

```
process GestionnaireFich[acc_l, lib_l, acc_c, lib_c] (nl : Natural): noexit :=  
    ( [nl >= 0] ->  
        ( acc_l ; GestionnaireFich[acc_l, lib_l, acc_c, lib_c](nl+1)  
          [] (* ou *)  
            lib_l ; GestionnaireFich[acc_l, lib_l, acc_c, lib_c] (nl-1)  
          )  
        [] (* ou *)  
        [nl = 0] ->  
            ( acc_c ; lib_c ; GestionnaireFich[acc_l, lib_l, acc_c, lib_c] (nl)  
              []  
              i ; GestionnaireFich[acc_l, lib_l, acc_c, lib_c] (nl) )  
            )  
    )  
endproc
```

Syntaxe de Basic LOTOS

stop	Processus prédéfini : Inaction, <i>deadlock</i>
exit	Terminaison normale (succès)
;	Opérateur de préfixage d'un comportement (séquence)
i;P	action interne i préfixe d'un comportement P
g;P	action g observable, préfixe du comportement P
P₁ P₂	Choix non déterministe entre les comportements P ₁ et P ₂
P₁ >> P₂	Composition séquentielle des deux comportements P ₁ et P ₂
P₁ [> P₂	Interruption (préemption) du comportement P ₁ par P ₂
P₁ [Lp] P₂	Composition parallèle de P ₁ et P ₂ avec synchronisation sur les actions listées dans Lp
P₁ P₂	Composition parallèle avec entrelacement des actions de P ₁ et P ₂
P₁ P₂	Composition parallèle de P ₁ et P ₂ avec synchronisation sur toutes les actions de l'alphabet de P ₁ et P ₂
P[g₁, g₂, g₃, ...]	Instanciation de P par les actions ou portes (<i>gates</i>) g ₁ , g ₂ , g ₃ , ...

TAB. 1 – Table des opérateurs de LOTOS

Structure d'une spécification

<pre> SPECIFICATION nom [liste des actions ou portes] (<liste param>) : <fonctionnalité> := <définitions de types> BEHAVIOUR (* comportement du système global *) <expression de comportement> WHERE <définitions de types> <définitions de processus> ENDSPEC </pre>

Exemple de type de donnée

<pre> type BIT is sorts BIT opns 0 : -> BIT 1 : -> BIT not : BIT -> BIT endtype </pre>

et l'utilisation

```

process MEDIUM [SC, RC] : noexit :=
  SC ?B : BIT; (* reception d'un bit dans la variable B *)
  ( RC!(not(B)); ...suite du comportement
  )
endproc

```

Interaction, Communication et Synchronisation

Les processus LOTOS interagissent en communiquant (émissions avec !, réceptions avec ?) et en se synchronisant sur des portes `Processus [|porte1, ..., porten|] Processus`.