

Correction Examen 2021

Valeran MAYTIE

Exercice 1 – Rétro-ingénierie

1. **Ocaml** $(e_1 = e_2) = e_3$

Python : $(e_1 == e_2 == e_3)$ vérifie qu'ils soient tous égaux

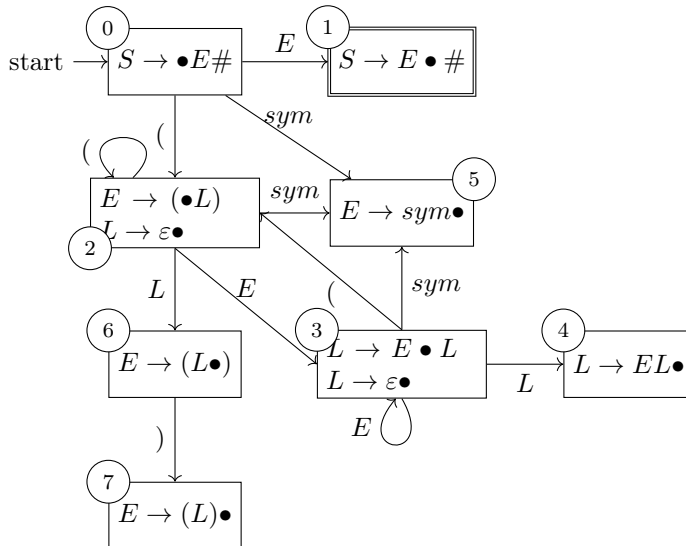
2. **Ocaml** : le type de e_1 et e_2 doivent être similaire et e_3 doit être de type *bool*

Python : à l'aire d'accepter n'importe quelle combinaison de types

Exercice 2 – 1. Grammaire G

S ::= E#
 E ::= (L)
 | sym
 L ::= ε
 | E L

2. Déterminisation :



3. Il peut y avoir des conflits (reduction/transition) pour les blocs avec la règle $L \rightarrow \varepsilon \bullet$

Si le prochain symbole est un symbole ou une parenthèse

Exemple : (L)

4. Table :

	Action			Sauts		
	#	()	sym	S	E	L
0		2	5		1	
1	OK					
2		2	5		3	6
		$L \prec \varepsilon$				
3		2	5		3	4
		$L \prec \varepsilon$				
4		$L \prec EL$				
5		$E \prec sym$				
6			7			
7		$E \prec (L)$				

	S	E	L
0.	∅	∅	∅
1.	∅	(, sym	∅
2.	(, sym	(, sym	(, sym
3.	(, sym	(, sym	(, sym

Premier

	S	E	L
0.	∅	∅	∅
1.	∅	#)
2.	∅	#, (, sym,))
3.	∅	#, (, sym,))

Suivant

Exercice 3 – Compilation vers de valeurs optionnelles :

1. Description :

registre	valeurs
\$a0	8
\$v0	0x10040008
\$t0	1

0x10040000	1
	2
0x10040008	1
	0x10040000

2. \$a0 = 0x10040000

Tas :

0x10040000	1
	0x10040008
0x10040008	1
	0x10040010
0x10040010	1
	3

3. récupère la valeur n de Somme (Somme n) dans le registre \$v0

```
lw $v0, 4($a0)
lw $v0, 4($v0)
```

4. code de la fonction f :

```
f:
    beqz $a0, N
    lw   $v0, 4($a0)
    addi $v0, $v0, 1
    jr   $ra
N:
    li   $v0, -1
    jr   $ra
```

5. Code :

```
bnez    $a0, some

bnez    $a1, incompatible
li      $v0, 1
b       fin
some:
    beqz $a1, incompatible

    lw   $a0, 4($a0)
    lw   $a1, 4($a0)
    seq  $v0, $a0, $a1
    b    fin
incompatible:
    li   $v0, 0
fin:
```

Exercice 4 – Typage et sémantique

Partie sur le sémantique pas vue en cours.