

# Théorie des Langages et Compilation (TLANG) – DS

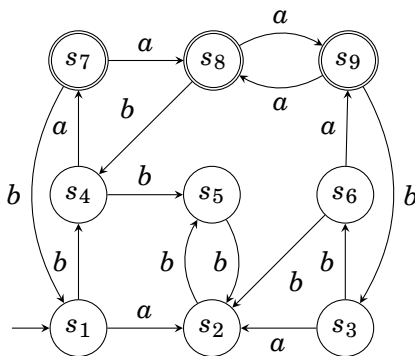
École Centrale de Nantes  
Option Informatique

Documents autorisés: une feuille A4 recto-verso de notes personnelles, manuscrites, et non photocopiées.

25 janvier 2019  
Durée: 2h

## 1 Partie 1 : analyse lexicale (30 min) 2 Partie 2 : analyse syntaxique (35 min)

**Question 1.** Minimisez l'automate suivant. *Donnez le détail de la construction.*



**Question 2.** Démontrez, avec le lemme de l'étoile, que le langage sur l'alphabet  $\Sigma = \{a, b, c\}$ ,  $L = \{wcw | w \in \{a, b\}^*\}$  n'est pas régulier ;

**Question 3.** On considère une définition alternative du langage accepté par un automate dans laquelle un mot est accepté s'il existe un chemin correspondant à ce mot dans l'automate et pour lequel l'un des états (pas forcément le dernier) est accepteur. Montrez que cette définition est équivalente à la définition classique : pour tout langage obtenu avec l'une des deux définitions, on peut l'obtenir aussi avec l'autre définition (pas forcément avec le même automate).

Soit la grammaire  $G$  suivante ( $S$  est l'axiome,  $A, B, C$  les non-terminaux,  $a, b, c$  les terminaux) :

$$\begin{aligned} S &\rightarrow BaB | CCA \\ A &\rightarrow Aa | bb \\ B &\rightarrow cAS | a \\ C &\rightarrow a | bc \end{aligned}$$

**Question 4.** Donnez l'arbre de dérivation du mot  $w = cbbcbcbbaa$  ;

**Question 5.** Calculez les ensembles PREMIER et SUIVANT pour les non-terminaux  $S, A, B$  et  $C$ . *Détaillez.*

**Question 6.** Calculez l'état initial de l'automate des items  $LR(0)$  de la grammaire  $G$  ainsi que ses successeurs directs (par une seule transition).

**Question 7.** Éliminez les récursivités à gauche dans la grammaire  $G$ . La grammaire obtenue est-elle  $LL(1)$ ? *Justifiez.*

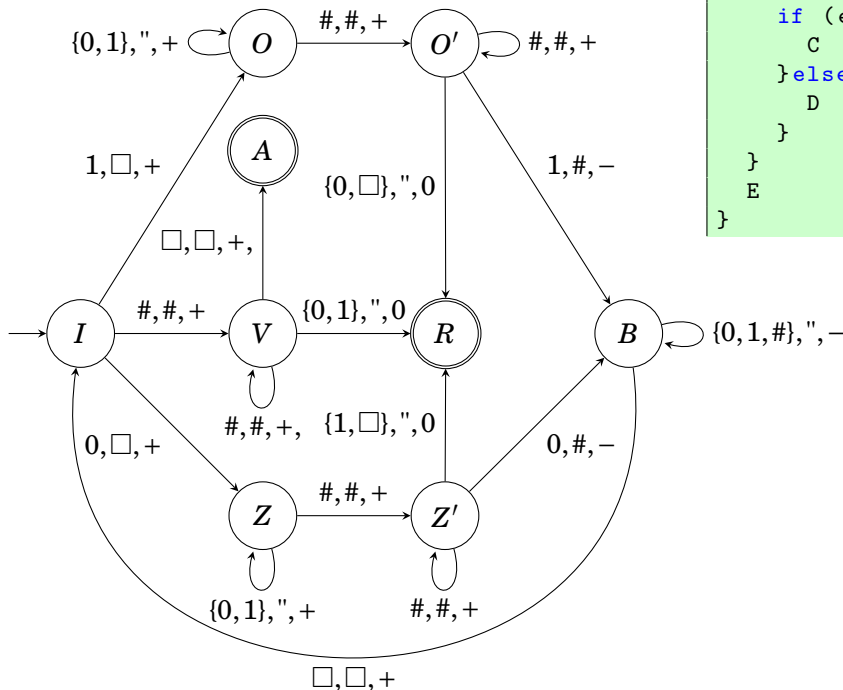
### 3 Partie 3 : Machines de Turing (30 min)

Soit la machine de Turing sur  $\{0,1,\#\}$  décrite par le schéma ci-dessous.  $I$  est l'état init,  $A$  l'état accept, et  $R$  l'état reject.  $\square$  est le caractère blanc (vide). Un ensemble en lecture (p. ex.  $\{0,1\}$ ) dénote que la transition est possible pour chacun des éléments de l'ensemble. Dans ce cas, le caractère " en écriture signifie que la machine laisse la bande inchangée, quel que soit le caractère lu.

**Question 8.** Quelles sont les entrées avec au plus un caractère '#' acceptées par la machine? *Justifiez*

**Question 9.** Au moment de l'acceptation, quel est, en fonction de l'entrée, le contenu de la bande? *Justifiez*

**Question 10.** Donnez des évaluations asymptotiques, quand la taille  $n$  de l'entrée tend vers l'infini, des complexités temporelles et spatiales au pire cas pour cette machine. Justifiez



A, B, C, D et E sont des listes d'instructions séquentielles, et  $\text{expr1}$ ,  $\text{expr2}$  et  $\text{expr3}$  sont des conditions booléennes simples. «  $\&\&$  » évalue ses opérandes en court-circuit (c'est-à-dire n'évalue pas l'opérande de droite si celui de gauche est faux).

**Question 11.** Donner sous forme de bloc le code engendré par la fonction suivante :

```
function f1 () {
  A
  if (expr1 && expr2) {
    B
  } else if (expr3) {
    C
  } else {
    D
  }
  E
}
```

**Question 12.** Donner sous forme de bloc le code engendré par la fonction suivante :

```
function f2 () {
  A
  while (expr1 && expr2) {
    B
    if (expr3) {
      C
    } else {
      D
    }
  }
  E
}
```