

Session : 2

EPREUVE : Langages Formels et Compilation

Durée : 1 h 30

Documents papiers - sauf livres - autorisés ; appareils électroniques interdits – montres connectées comprises

Les exercices sont indépendants. Le barème est donné à titre indicatif.

Exercice 1 – 3 points

Soit l'automate à pile $P1 = (\{q\}, \{a, b\}, \{a, b, z_0\}, t, q, z_0, \emptyset)$

Avec t définie par :

- | | | |
|--------------------------------|--------------------------------|--------------------------------|
| 1) $t(q, a, z_0) = (q, a)$ | 4) $t(q, a, b) = (q, \lambda)$ | 7) $t(q, b, \lambda) = (q, b)$ |
| 2) $t(q, a, a) = (q, aa)$ | 5) $t(q, b, z_0) = (q, b)$ | 8) $t(q, b, a) = (q, \lambda)$ |
| 3) $t(q, a, \lambda) = (q, a)$ | 6) $t(q, b, b) = (q, bb)$ | |

- 1) Analysez le mot abbabaab avec cet automate à pile (étapes de l'analyse, conclusion, justification).
- 2) Au cours de l'analyse, on peut se rendre compte que plusieurs sous-mots appartiennent au langage reconnu par $P1$. Lesquels et comment le sait-on ?

Exercice 2 – 3.5 points

Soit la machine de Turing suivante :

$MT2 = (\{q_0, q_1, q_2, q_3, q_4\}, \{a, b, c\}, \{a, b, c, \square\}, t, q_0, \{q_4\})$

avec t définie par :

- | | | |
|------------------------------|------------------------------|-------------------------------------|
| 1) $t(q_0, a) = (q_1, a, D)$ | 5) $t(q_1, b) = (q_2, b, D)$ | 9) $t(q_3, b) = (q_1, b, D)$ |
| 2) $t(q_0, b) = (q_1, b, D)$ | 6) $t(q_1, c) = (q_2, c, D)$ | 10) $t(q_3, c) = (q_1, c, D)$ |
| 3) $t(q_0, c) = (q_1, c, D)$ | 7) $t(q_2, c) = (q_3, c, D)$ | 11) $t(q_3, \square) = (q_4, b, D)$ |
| 4) $t(q_1, a) = (q_2, a, D)$ | 8) $t(q_3, a) = (q_1, a, D)$ | |

- 1) Analysez le mot aacbbc avec $MT2$ (étapes de l'analyse, conclusion, justification).
- 2) Comment sont constitués les mots du langage reconnu par $MT2$? Donnez le maximum de précisions.

Exercice 3 – 3 points

Soit la grammaire

$G_3 = (\{S, B, W, X\}, \{a, b, c\}, S, \{S \rightarrow abc, S \rightarrow aSBc, cB \rightarrow WB, WB \rightarrow WX, WX \rightarrow BX, BX \rightarrow Bc, bB \rightarrow bb\})$

- 1) De quel type est G_3 ? Justifiez votre réponse.
- 2) Donnez deux mots du langage engendré par G_3 ainsi que les dérivations qui permettent de les obtenir.

Exercice 4 – 4.5 points

Soit la grammaire suivante :

$G4 = (\{S, X, A, B, T, U, Y\}, \{a, b\}, S, \{S \rightarrow AT|AU|AB, X \rightarrow BY|BA, A \rightarrow a, B \rightarrow b, T \rightarrow SB, U \rightarrow XB, Y \rightarrow XA\})$

- 1) A partir de $G4$, analysez le mot abbaab par l'algorithme de Cocke-Kasami-Younger (pyramide, conclusion, justification). Inutile d'expliquer la construction de la pyramide.
- 2) Donnez la liste de tous les sous-mots de abbaab.
- 3) En vous basant uniquement sur votre pyramide, quels sont les sous-mots de abbaab qui appartiennent au langage engendré par $G4$? Justifiez votre réponse.

Exercice 5 – Lex & Yacc – 6 points (à rendre sur une feuille à part)

Comme dans le projet, les fichiers que l'on veut analyser décrivent un environnement dans lequel sont placés des éléments. L'environnement et sa description sont toutefois simplifiés.

L'environnement est de taille fixe 5x5 cellules. La description de chaque cellule de cet environnement est un code couleur parmi R, V, B, ou _ correspondant à :

- rouge (code R) pour un acteur de la simulation
- vert (code V) pour une végétation
- bleu (code B) pour un point d'eau
- rien (code _) si la cellule est vide

Si toutes les cellules sont initialisées, on trouve exactement 25 symboles dans le fichier (cf exemple 1). On accepte aussi les descriptions incomplètes et donc comportant moins de 25 symboles (cf exemple 2).

- 1) Donnez le code de l'analyseur lexical qui analyse les unités lexicales de ce langage (voir exemples 1 et 2). Les définitions régulières et la gestion des erreurs sont **indispensables**. Vérifier que le nombre de symboles acceptés est **inférieur ou égal** au nombre de cellules de l'environnement. Les *include* du C ne sont pas nécessaires.
- 2) Donnez l'implémentation de l'analyse syntaxique avec Yacc en y incorporant les actions permettant de stocker les valeurs des cellules de l'environnement dans un tableau de taille 5x5.
- 3) Pour faciliter la configuration de l'environnement, on souhaite maintenant spécifier le nombre de répétition avant la valeur de la cellule (cf. exemple 3), modifier les 2 sources Lex et Yacc.

Exemple 1 : RRRRRBBBBB_R_B_B_RBB_R_RV

1^{er} ligne : 5 cellules rouges – 2^e ligne : 5 cellules bleues – 3^e ligne : vide, rouge, vide, bleu, vide – 4^e ligne : bleu, vide, rouge, bleu, bleu – 5^e ligne : vide, rouge, vide, rouge, vert.

Exemple 2 : RVB_RR_BB

1^{er} ligne : rouge, vert, bleu, vide, rouge – 2^e ligne : rouge, vide, bleu, bleu. Les autres cellules ne sont pas décrites.

Exemple 3 : 3R_VV6_

1^{er} ligne : rouge, rouge, rouge, vide, vert – 2^e ligne 2 vert, vide, vide, vide, vide – 3^e ligne : vide, vide. Les autres cellules ne sont pas décrites.