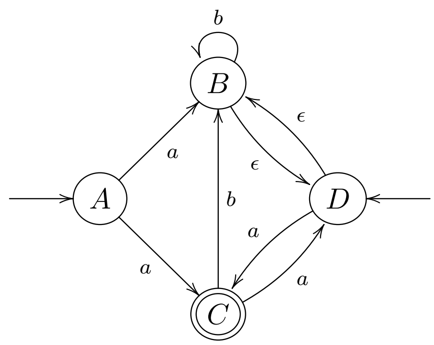
Automates et Théorie des Langages – Session 1 – 19/11/2020 – 1h30

Seul document autorisé : formulaire inclu dans le sujet

Calculatrice, Téléphone Portable, Ordinateur Interdits – Sauf dérogation

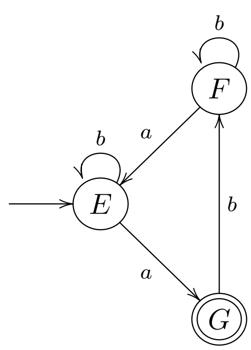
Les exercices sont indépendants les uns des autres. Lisez le sujet avant de commencer. Traitez les exercices dans votre ordre de préférence.
Si vous trouvez dans le sujet un élément qui vous semble erroné, signalez le sur votre copie, indiquez les hypothèses que vous faites, et poursuivez la composition.
Dans tous les exercices, vous veillerez à justifier vos résultats, par exemple en faisant apparaître le nom des règles utilisées, il en sera tenu compte dans la notation.

Exercice 1 Soit l'automate fini $\mathcal{E} = (\{A, B, C, D\}, \{a, b\}, \{A, D\}, \{C\}, \delta_{\mathcal{E}})$ dont la fonction de transition est définie par :



1. $\mathcal{E}$ est-il déterministe ? S'il ne l'est pas, donner toutes les causes d'indéterminisme, puis le déterminiser.
2. Donner la table de transition de l'automate déterministe.

Exercice 2 Soit l'automate fini $\mathcal{E} = (\{E, F, G\}, \{a, b\}, \{E\}, \{F\}, \delta_{\mathcal{E}})$ dont la fonction de transition est définie par :



1. Construire le système d'équations sur les expressions régulières associé à l'automate $\mathcal{E}$;
2. Résoudre ce système et donner une expression régulière décrivant le langage reconnu par l'automate $\mathcal{E}$.

Exercice 3 Calculer par la méthode des dérivées un automate reconnaissant le langage décrit par l'expression régulière $(a|\Lambda)\Big((ba)^\star \mid ba\Big)$.

Exercice 4 Soit la grammaire décrivant les nombres complexes dont les parties réelles et imaginaires sont des entiers naturels $G = (A, V, N, P)$ composée des non-terminaux $V = \{N, R, I\}$, de l'axiome N, des terminaux $A = \{c, i, p\}$ (c est un chiffre, p est l'addition binaire) et de l'ensemble P de règles suivantes :

1. $N \to R\, p\, I$
2. $N \to R$
3. $N \to I$
4. $I \to R\, i$
5. $R \to R\, c$
6. $R \to c$

1. Transformer G en une grammaire régulière (de la forme $X \to c\, Y$ ou $Z \to \Lambda$) décrivant le même langage en utilisant des transformations de grammaire préservant le langage reconnu (substitution, factorisation et élimination récursivité gauche) ;

2. Construire l'automate déterministe fini équivalent à la grammaire obtenu.

Exercice 5 Soit une grammaire simplifiée du format d'échange d'objets JSON (*JavaScript Object Notation*) $G_0 = (A, V, O, P)$ composée des non-terminaux $V = \{O, F, L\}$, de l'axiome O, des terminaux $A = \left\{\texttt{id}\ \texttt{num}\ \{\ \}\ ,\ :\right\}$ et de l'ensemble P de règles suivantes :

1. $O \to \{\ \}$
2. $O \to \{\ L\ \}$
3. $L \to L\ ,\ F$
4. $L \to F$
5. $F \to \texttt{id} : \texttt{num}$

1. Grammaire LL(1) :

   (a) Est ce que la grammaire G_0 est récursive à gauche ? Si oui, éliminer la récursivité à gauche et construire la grammaire G_1. Si G_0 n'est pas récursive à gauche, alors G_1 est par la suite égale à G_0.

   (b) Calculer les symboles directeurs associés aux différentes règles de production de G_1 ;

   (c) G_1 est-elle LL(1) ? Pourquoi ? Si G_1 n'est pas LL(1), transformer G_1 en G_2 pour la rendre LL(1) et calculer les symboles directeurs de G_2.

2. Questions de cours :

   (a) Quel est l'apport de la construction monadique dans la programmation des analyseurs descendants récursifs dans des langages fonctionnels purs (réponse en 3 lignes maximum) ?

   (b) Quels sont les rôles des analyseurs lexical et syntaxique (réponse en 10 lignes maximum en s'appuyant sur un schéma explicatif) ?

   (c) Quel est le rôle des actions dans les analyseurs syntaxiques (réponse en 5 lignes) ?

Equivalence entre expressions régulières : L'opérateur de concatenation/juxtaposition . est implicite pour ne pas surcharger l'écriture des expressions : $e_1.e_2$ est notée $e_1\, e_2$.

$$\begin{array}{ll}
\emptyset\, e = e\, \emptyset = \emptyset & \Lambda\, e = e\, \Lambda = e \\
e \mid \emptyset = \emptyset \mid e = e & e \mid e = e \\
e_1\,(e_2\, e_3) = (e_1\, e_2)\, e_3 & e_1 \mid (e_2 \mid e_3) = (e_1 \mid e_2) \mid e_3 \\
e_1\,(e_2 \mid e_3) = (e_1\, e_2) \mid (e_1\, e_3) & (e_1 \mid e_2)\, e_3 = (e_1\, e_3) \mid (e_2\, e_3) \\
e_1 \mid e_2 = e_2 \mid e_1 & \emptyset^\star = \Lambda^\star = \Lambda \\
e^\star = \Lambda \mid e^+ & e^+ = e\, e^\star = e^\star\, e \\
e^\star\, e^\star = e^\star & e^{\star\star} = e^\star \\
e = e^\star \Leftrightarrow e = e\, e & e\, e^\star = e^\star \Leftrightarrow \Lambda \in L(e) \\
(e_1^\star\, e_2^\star)^\star = (e_1 \mid e_2)^\star = (e_1^\star \mid e_2^\star)^\star & \\
(e_1^\star\, e_2)^\star\,(e_1^\star) = (e_1 \mid e_2)^\star = e_1^\star\,(e_2\,(e_1^\star))^\star &
\end{array}$$

ϵ-fermeture : L'ϵ-fermeture d'un ensemble d'états E est la fermeture réflexive et transitive de la relation de transition sur ϵ. Il s'agit de l'union de E et de tous les états accessibles depuis les états de E en suivant un nombre quelconque de transitions sur ϵ.

Théorème de Arden : Soient x une variable, e_1 et e_2 des expressions régulières, l'équation $x = e_1\, x \mid e_2$ admet au moins une solution : $x = e_1^\star\, e_2$

Dérivation des expressions régulières :

$$\begin{array}{ll}
D_a(a) = \Lambda & D_a(b) = \emptyset \\
D_a(\emptyset) = \emptyset & D_a(\Lambda) = \emptyset \\
D_a(e^\star) = D_a(e)\, e^\star & D_a(e_1 \mid e_2) = D_a(e_1) \mid D_a(e_2) \\
D_a(e_1\, e_2) = D_a(e_1)\, e_2 \mid \delta(e_1)\, D_a(e_2) & \\
\delta(\emptyset) = \emptyset & \delta(\Lambda) = \Lambda \\
\delta(a) = \emptyset \text{ si } a \in A & \delta(e^\star) = \Lambda \\
\delta(e_1\; e_2) = \delta(e_1)\; \delta(e_2) & \delta(e_1 \mid e_2) = \delta(e_1) \mid \delta(e_2)
\end{array}$$

Soit la grammaire non contextuelle $G = (A, V, S, P)$:

Calcul des Premiers :

$$\begin{array}{l}
\text{Premiers}(\Lambda) = \{\Lambda\} \\
\text{Premiers}(a\,\alpha) = \{a\} \text{ avec } a \in A \text{ et } \alpha \in (A \cup V)^\star \\
\text{Premiers}(X) = \displaystyle\bigcup_{X \to \gamma \in P} \text{Premiers}(\gamma) \text{ avec } X \in V \text{ et } \gamma \in (A \cup V)^\star \\
\text{Premiers}(X\,\alpha) = \text{Premiers}(X)\, \underbrace{\setminus\{\Lambda\} \cup \text{ Premiers}(\alpha)}_{\text{si } \Lambda \in \text{Premiers}(X)} \text{ avec } X \in V \text{ et } \alpha \in (A \cup V)^\star
\end{array}$$

Calcul des Suivants :

$$\text{Suivants}(X) = \bigcup_{Y \to \alpha\, X\, \beta \in P} \text{Premiers}(\beta)\, \underbrace{\setminus\{\Lambda\} \cup \text{Suivants}(Y)}_{\text{si } \Lambda \in \text{Premiers}(\beta)}\, \underbrace{\cup\{\$\}}_{\text{si } X = S} \text{ avec } X, Y \in V \text{ et } \alpha, \beta \in (A \cup V)^\star$$

Calcul des Symboles Directeurs :

$$\text{Directeurs}(X \to \alpha) = \text{Premiers}(\alpha)\, \underbrace{\setminus\{\Lambda\} \cup \text{Suivants}(X)}_{\text{si } \Lambda \in \text{Premiers}(\alpha)} \text{ avec } X \in V \text{ et } \alpha \in (A \cup V)^\star$$