Informatique - MPI

Question 1. Pour établir l'ambiguité de la grammaire, il suffit de trouver un mot qui admet deux arbres syntaxiques différents. Par exemple, le mot $babab \in \mathcal{L}(G)$.

Commentaire – Le jury attend du candidat qu'il exhibe un mot montrant l'ambiguïté, en justifiant cette dernière via deux arbres syntaxiques différents ou bien deux dérivations gauches (ou droites) différentes.

Question 2. On constate que $\mathcal{L}(G)$ est rationnel car dénoté par l'expression rationnelle $(ba)^*b$.
Commentaire – Il est attendu que le candidat exhibe une expression régulière pour ce langage (qui est donc régulier). Plusieurs sont possibles et toutes sont acceptées. Aucune justification n'est nécessaire à ce stade, sauf si l'expression proposée est démesurément complexe, auquel cas le jury invite le candidat à lui expliquer. Certains candidats n'ont pas compris ce qui était attendu par la plus petite classe de langages*, le jury a alors précisé ses attentes sans pénaliser pour autant ces candidats. Au moins un candidat a répondu à cette question en affirmant que $\mathcal{L}(G)$ est en fait local. Il a été invité à poursuivre l'exercice avec cette réponse.*

Question 3. On établit le résultat par double inclusion.

Montrons tout d'abord que $L \subset \mathcal{L}(G)$. Pour $n \in \mathbb{N}^*$, on définit la propriété $H(n)$ suivante : si $u \in L$ est un mot de taille n alors u est engendré par la grammaire G.

- C'est le cas pour $n = 1$ puisque le seul mot de L de cette taille est b, engendré par la deuxième règle de G.
- Soit $n \in \mathbb{N}^*$ et u un mot de L de taille $n+1$. Comme $|u| \geqslant 2$, ba est nécessairement préfixe de u et il existe donc $v \in (ba)^*b = L$ tel que $u = bav$. Par hypothèse, ce mot v est engendré par G : il existe donc une dérivation telle que $S \Rightarrow^* v$.

On en déduit que :

$$S \Rightarrow SaS \Rightarrow baS \Rightarrow^* bav = u$$

est une dérivation licite et donc que $u \in \mathcal{L}(G)$.

Montrons réciproquement que $\mathcal{L}(G) \subset L$. Pour $n \in \mathbb{N}^*$, on définit la propriété $H(n)$ suivante : si $u \in \Sigma^*$ se dérive de S en n dérivations alors $u \in L$.

- La propriété est vérifiée pour $n = 1$: le seul mot de Σ^* qu'on peut obtenir en une dérivation est $b \in L$.
- Soit $n \in \mathbb{N}^*$ et u un mot dans $\mathcal{L}(G)$ tel que $S \Rightarrow^{n+1} u$. Comme ce mot est obtenu en au moins 2 dérivations, les règles de G nous informent que la première est nécessairement $S \to SaS$ (sans quoi ce serait $S \to b$ et dans ce cas u serait obtenu en une seule dérivation). Donc la dérivation permettant d'obtenir u se décompose en :

  $$S \Rightarrow SaS \Rightarrow^n u$$

  Ainsi, il existe $v, w \in \Sigma^*$, $k_1, k_2 \in [\![1, n]\!]$ tels que $u = vaw$, $S \Rightarrow^{k_1} v$, $S \Rightarrow^{k_2} w$ et $k_1 + k_2 = n$. L'hypothèse de récurrence (forte) s'applique à v et w et on en déduit que ces deux mots appartiennent au langage dénoté par $(ba)^*b$ donc qu'il existe $r_1, r_2 \in \mathbb{N}$ tels que $v = (ba)^{r_1}b$ et $w = (ba)^{r_2}b$. Par conséquent, $u = (ba)^{r_1}ba(ba)^{r_2}b$ soit $u = (ba)^{r_1+r_2+1}b \in L$.

Commentaire – Cette question attend une preuve précise et rigoureuse, par exemple par double inclusion. Les explications vagues et les arguments d'évidence ne satisfont pas le jury.

Question 4. Sachant que $\mathcal{L}(G) = (ba)^*b$, il convient désormais de trouver une grammaire non ambiguë engendrant ce langage. On peut proposer par exemple la grammaire G' dont les règles sont :

$$S \to Tb \qquad T \to baT \mid \varepsilon$$

les règles sur T permettant de générer le facteur dans $(ba)^*$ et la première de rajouter le b final.

Cette grammaire est non ambiguë car pour tout mot dans $\mathcal{L}(G')$, il existe une unique dérivation permettant de le construire (donc un seul arbre syntaxique). L'unicité découle du fait que dans cette grammaire, un non terminal se dérive toujours en un mot qui contient au plus un seul non terminal.

Commentaire – Comme pour la deuxième question, plusieurs grammaires peuvent être proposées. Le jury n'attend pas une preuve rigoureuse de non ambiguïté.

Question 5. La question demande de montrer que tout langage rationnel peut être engendré par une grammaire non ambiguë. Soit donc L un langage rationnel. Par le théorème de Kleene, il existe un automate fini déterministe $\mathcal{A} = (\Sigma, Q, \{q_0\}, F, \delta)$ qui reconnaît L.

Considérons alors la grammaire dont les non terminaux sont $\{V_q \mid q \in Q\}$, l'axiome est V_{q_0}, les terminaux sont les lettres de Σ et dont les règles sont données par :

- pour toute transition $q \xrightarrow{a} q'$ dans A, on ajoute la règle $V_q \to aN_{q'}$;
- pour tout $q \in F$, on ajoute la règle $V_q \to \varepsilon$.

Cette grammaire engendre L de manière non ambiguë en raison au déterminisme de $\mathcal{A}$.
Commentaire – Proposer une construction correcte d'une grammaire non ambiguë pour un langage régulier, à partir de l'automate associé, suffit à obtenir tous les points. Le jury n'hésitait pas à aiguiller le candidat si nécessaire pour cette question.