Corrigé sujet CCINP (2022-2024-2023)

Partie I
Q1. $L_2(\mathcal{A}_{\{q_0\}}) = \{b^n aa\}$.
Q2. Soit un calcul réussi au seuil 2 dont on notera $q_{i_0}, \cdots, q_{i_n}$ les états successifs et $a_1 \cdots a_n$ les étiquettes. Par définition de l'automate, $a_1 \cdots a_n = b^p aab^q$ où p et q sont deux entiers. Si $p \geq 2$, $q_{i_0} = q_{i_1} = q_{i_2} = q_0$; ce calcul ne remplit pas la condition de seuil donc $p \leq 1$. De même, si $q \geq 2$, $q_{i_{n-2}} = q_{i_{n-1}} = q_{i_n} q_2$ donc ce calcul ne remplit pas la condition de seuil. On a donc $q \leq 1$. On a donc $a_1 \cdots a_n \in \{aa, aab, baa, baab\}$. Réciproquement, ces quatre mots remplissent bien la condition de seuil et sont des calculs réussis dans l'automate. On a donc $L_2(\mathcal{A}_{\{q_1\}}) = \{aa, baa, aab, baab\}$.
Q3. Soit n la taille d'un calcul réussi. Si $s \leq n$, il n'existe pas d'état $q \in \{q_0, \cdots, q_s\} \cap \emptyset$ donc ce calcul n'est pas réussi au seuil s. Si $s \leq n$, $\mathrm{Card}(L_s(\mathcal{A}_\emptyset)) = 0$. Si $s > n$, un calcul réussi de taille n est du type $b^p aab^q$ donc $p+2+q = n$. Il n'existe pas d'entiers $i \geq 0$ tels que $i+s \leq n$ donc ce calcul est réussi au seuil. On a donc $L_s(\mathcal{A}_\emptyset) = \{b^p aab^q \,/\, p+q+2 < s\}$. On a donc $\mathrm{Card}(L_s(\mathcal{A}_\emptyset)) = \mathrm{Card}(\{(p,q) \in \mathbb{N}^2 \,/\, p+q \leq s-3\}) = \sum\limits_{p=0}^{s-3} \mathrm{Card}(\{q \,/\, 0 \leq q \leq s-3-q\}) = \sum\limits_{p=0}^{s-3} (s-2-p) = \sum\limits_{k=1}^{s-2} k = \frac{(s-2)(s-1)}{2}$.
Q4. La taille d'un calcul réussi au seuil s ne passant par aucun état de $\mathcal{O}$ est au plus $s-1$.
Q5. $0 \leq i_0 \leq s$. $l - s \leq i_k \leq l$. Pour tout $j \in [\![0, k-1]\!]$, $i_j + 1 \leq i_{j+1} \leq i_j + 1 + s$ donc $1 \leq i_{j+1} - i_j \leq 1 + s$.
Q6. On note $A = (Q, \Sigma, I, \delta, F)$, $Q = \{q_0, \cdots, q_{p-1}\}$. On note $Q' = \{(q_i, k) \,/\, 0 \leq i \leq p-1, 0 \leq k \leq s\}$, $F' = \{(q_i, k) \,/\, q_i \in F\}$, $I' = \{(q_i, 0) \,/\, q_i \in I\}$. Pour tout $(q_i, k) \in Q'$, $a \in \Sigma$, on pose, si $k \leq s-1$, $\delta'((q_i, k), a) = \{(q_j, k+1) \,/\, q_j \in \delta(q_i, a) \cap (Q \backslash \mathcal{O})\} \cup \{(q_j, 0) \,/\, q_j \in \delta(q_i, a) \cap \mathcal{O}\}$. On note $\mathcal{A}'$ l'automate fini non déterministe ainsi créé.
Un calcul réussi dans cet automate commence à un état initial $(q_i, 0)$ où $q_i \in I$, se termine en un état final (q_n, k) tel que $q_n \in F$, enchaîne des transitions (q_i, k) vers $(q_j, k+1)$ ou $(q_j, 0)$ étiquetées par a tel que la transition de q_i vers q_j étiquetée par a existe dans l'automate a. Un tel calcul est donc réussi dans l'automate $\mathcal{A}$. En outre, s'il existe une succession $q_i, \cdots, q_{i+s}$ d'états non dans $\mathcal{O}$, alors le calcul correspondant dans $\mathcal{A}'$ par par les états $(q_{i+k}, k+1)$ donc par (q_{i+s-1}, s), état qui n'est l'origine d'aucune transition donc le calcul ne peut réussir. On en déduit qu'un calcul réussi dans l'automate $\mathcal{A}'$ est un calcul réussi au seuil s.
Réciproquement, un calcul réussi au seuil s dans l'automate $\mathcal{A}$ donne dans l'automate $\mathcal{A}'$ un calcul commençant à $(q_i, 0) \in I'$, se terminant à $(q_n, k) \in F'$ et, de par la condition de seuil et la définition de l'automate, un calcul n'aboutissant jamais en un état (q_j, s). Il donne donc un calcul réussi dans $\mathcal{A}'$.
L'automate $\mathcal{A}'$ reconnaît donc $L_s(\mathcal{A}_\mathcal{O})$ et possède $s(p+1)$ états.

Partie II
Q23. Soit $p = |x| > 0$. Il n'existe pas d'entier $0 \leq i \leq |x| - p - 1 = -1$ donc $p = |x|$ est une période.
Q24.

```
let occurence x y = let n=String.length x and m=String.length y in
   let rec aux i j = if j+n>m then false else
      if i=n then true else ( (x.[i]=y.[j]) && aux (i+1) (j+1) ) || (aux 0 (j+1))) in
   aux 0 0;;
```

Q25. Dans le pire des cas, pour tout lettre de y, disons la j-ème lettre de y, on vérifie, pour chaque lettre de x, disons la i-ème, si la $j+i$-ème lettre de y est égal à la i-ème lettre de x. La complexité est donc $\sum\limits_{j=1}^{|y|} \sum\limits_{i=1}^{|x|} O(1) = O(|x|.|y|)$.
Q26. La période est 3.
Q27.

```
let periode x = let n=String.length x in
    let rec aux i p =
        if i+p>=n then p
        else if x.[i]=x.[i+p] then aux (i+1) p else aux 0 (p+1);;
```

Q28. Supposons que p est une période de x. Soit u le préfixe de taille p de x et v son suffixe de taille p de x. Il existe w_1, w_2 des mots tels que $x = uw_1 = w_2v$. Remarquons que $|w_1| = |x| - |u| = |x| - p = |x| - |v| = |w_2|$.

Soit p fixé. Montrons, par récurrence sur $|w_1| = |w_2|$, que, si $x = uw_1 = w_2v$ et $|u| = |v|$ est une période de x, alors $w_1 = w_2$.
Si $|w_1| = 0$, alors $w_1 = w_2 = \epsilon$.
Soit $k \geq 0$. Supposons le résultat valable pour $|w_1| = |w_2| = k$. Supposons que $|w_1| = |w_2| = k + 1$. Soit x_0 la première lettre de x. $x = w_2v$ donc la première lettre de w_2 est x ; $x = uw_1$ donc x_0 est la première de u ; par périodicité, x_0 est la première lettre de w_1. On peut noter $w_1 = x_0w_1'$ et $w_2 = x_0w_2'$, $u = x_0u'$. Comme $x = uw_1 = w_2v$, $x_0u'x_0w_1' = x_0w_2'v$ donc $u'x_0w_1' = w_2'v$. Comme p est une période de x, p est une période de $x' = u'x_0w_1' = w_2'v$, $p = |v| = |u'x_0|$ et $|w_1'| = |w_2'| = |w_1| - 1 = k$. De par l'hypothèse de récurrence, $w_1' = w_2'$ donc $w_1 = x_0w_1' = x_0w_2' = w_2$.
Par récurrence, $w_1 = w_2$ donc il existe des mots u, v et $w = w_1 = w_2$ tels que $x = uw = wv$ et $|u| = |v| = p$.
Réciproquement, supposons qu'il existe u, v, w tels que $x = uw = wv$ et $|u| = |v| = p$. Notons $w = w_0 \cdots w_{|w|-1}$. Remarquons que $|w| = |x| - p$.
Comme $x = wv$, pour tout $0 \leq i \leq |x| - p - 1$, $x_i = w_i$. Comme $x = uw$, pour tout $0 \leq i \leq |x| - p - 1$, $w_i = x_{i+p}$. En particulier, pour tout $0 \leq i \leq |x| - p - 1$, $x_i = w_i$ et $x_{i+p} = w_i$ donc $x_i = x_{i+p}$. Ainsi, p est une période de x.
Q29. Notons $y = Bord(x)$.
Supposons que $y = \epsilon$. Alors $n = 1$, le seul bord de x est ϵ. La suite des bords de x est donc réduite à ϵ.
Supposons que $|x| \geq 1$. Supposons le résultat à établir pour tout mot de taille strictement inférieure à $|x|$. On a $|y| < |x|$. $x = yu$ et $x = vy$. Un bord de x est z et $x = zu' = v'z$. Par maximalité de y, comme z est un préfixe de $x = yu$, z est un préfixe de y ; de même, z est un suffixe de y donc z est un bord de y. Par hypothèse de récurrence, z est l'un des $Bord^j(Bord(x))$ donc est du type $Bord^{j+1}(x)$. Réciproquement, tout élément $Bord^{j+1}(x) = Bord^j(Bord(x))$ est, par hypothèse de récurrence, l'un des éléments de la suite des bords de $Bord(x) = y$ donc, un tel élément z vérifie $y = zu'$ et $y = v'z$ pour certains mots u', v' donc $x = yu = zu'u$ et $x = vy = vv'z$ donc z est un bord de x.
Ainsi, les éléments $Bord^j(x)$ sont les bords de x. Pour tout j, $Bord^{j+1}(x)$ est un bord de $Bord^j(x)$ donc $Bord^{j+1}(x)$ est de longueur inférieur à $Bord^j(x)$ donc la suite $(Bord^j(x))$ est la suite des bords par ordre décroissant de longueur.
En outre, si z est un bord de x, $x = zu = vz$ et $|u| = |x| - |z| = |v|$ donc $|x| - |z|$ est une période de x donc $|x| - |Bord^j(x)|$ est une période de x pour un certain j. Réciproquement, si p est une période de x, $x = uw = wv$, $|u| = |v| = p$ donc w est un bord de x donc un élément du type $Bord^j(x)$ donc $p = |x| - |w| = |x| - |Bord^j(x)|$.
Q30. Un mot x est une occurrence d'un texte y dont on connaît les bords si et seulement si ce mot peut s'écrire $x = u'v'$ et, en écrivant pour un bord w, $y = uw = wv$, u' est suffixe de u et v' un préfixe de u.
Q31.

```
let t i c = match i,c with
   | 0,'a' -> 1
   | 0,_ -> 0
   | 1,'a' -> 2
   | 1,'b' -> 3
   | 1,'c' -> 0
   | 2,'a' -> 2
   | 2,'b' -> 3
   | 2,'c' -> 0
   | 3,'a' -> 4
   | 3,_ -> 0
   | 4,'a' -> 2
   | 4,'b' -> 3
   | 4,'c' -> 0 in
let automate={etats=[0;1;2;3;4] ; alphabet=['a','b','c'] ; initial=0 ; transition = t ; final = [2;4]};;
```

Q32. L'état 0 est initial donc accessible, co-accessible par exemple via le mot aa. L'état 1 est accessible via le mot a et co-accessible via le mot a. L'état 2 est accessible via le mot aa et final donc co-accessible. L'état 3 est accessible via le mot ab et co-accessible via le mot a. L'état 4 est accessible via le mot aba et final donc co-accessible. Chaque état étant accessible et co-accessible (donc utile), l'automate est émondé.
Q33. On ajoute un unique état initial et un unique état final. Après élimination de l'état 0, on obtient l'automate :

Q34. Cette liste contient les entiers i tels que $\lambda_0 \cdots \lambda_i$ est reconnu par l'automate donc possède le motif X comme suffixe. Cette liste contient donc les entiers i tels que λ_i est la fin d'une occurrence d'un motif X dans le texte. Cette liste

est la liste de tous les indices d'occurrences du motif X dans le texte.
Pour l'automate M et $t = cabaac$, on obtient $lst = [4; 3]$.

Q35.

```
let cherche m t = let n=String.length t in
   let rec aux i l etat = match i with
       | n -> l
       if List.mem etat (m.final) then (n-1)::l else l
       | _ -> let e=m.transition etat t.[i] in
       if List.mem e (m.final) then aux (i+1) (i::l) e
       else aux (i+1) l e
    in aux 0 [] (m.initial);;
```

Q36. Notons $t = \lambda_0 \cdots \lambda_{|t|-1}$ et e_0 l'état initial. Par récurrence, au i-ème passage de boucle, e prend la valeur $\delta(e_0, \lambda_0 \cdots \lambda_{i-1})$. Par définition de l'algorithme, on ajoute $i-1$ à la liste si et seulement si $\lambda_0 \cdots \lambda_{i-1}$ est reconnu par l'automate donc si et seulement si $\lambda_0 \cdots \lambda_{i-1} \in A^\star X$ si et seulement si une occurence de X s'achève en l'indice $i-1$. La liste renvoyée contient donc toutes les occurrences de X dans le texte t.

Partie III

Q17. $\delta^\star(q_0, \epsilon_1 \cdots \epsilon_n) = (n \mod 3, \mathrm{Card}(\{i = 1 \mod 3 \,/\, \epsilon_i = 1\}) \mod 2)$.

Q18. $\delta^\star(q_0, \epsilon_1 \cdots \epsilon_n) = (n \mod p, \mathrm{Card}(\{i = k \mod p \,/\, \epsilon_i = 1\})$.

Q19. Un mot est reconnu par $A_{0,2}$ si et seulement si son nombre de 1 en position paire est impair. Un mot est reconnu par $A_{1,2}$ si et seulement si son nombre de 1 en position impaire est impair. Soit un mot reconnu par $A_{k,p}$. Supposons que ce mot n'est reconnu ni par $A_{0,2}$ ni par $A_{1,2}$. Alors son nombre de 1 en position paire est pair et son nombre de 1 en position impaire est pair. Son nombre de 1 est donc paire. Le nombre de 1 en position congrue à k modulo p est impair. En outre, les ensembles $\{m \in [\![0, |w|-1]\!] \,/\, m = l \mod p\,,\, w_m = 1\}$, pour $0 \le l \le p-1$, est une partition de $\{m \in [\![0, |w|-1]\!] \,/\, w_m\}$. $\mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, w_m\}$ est impair ; $\mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, w_m\} = \sum_{l=0}^{p-1} \mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = l \mod p\,,\, w_m = 1\}$ donc $\sum_{l=0}^{p-1} \mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = l \mod p\,,\, w_m = 1\} = 0 \mod 2$. Comme w est reconnu par $A_{k,p}$, $\mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = k \mod p\,,\, w_m = 1\} = 1 \mod 2$ donc il existe $l \neq k$ tel que $\mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = l \mod p\,,\, w_m = 1\} = 1 \mod 2$ donc w est reconnu par $A_{l,p}$.

Q20. Pour tout $w \in \Sigma^\star$, $L_{k,p}(w) = \mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = k \mod p\,,\, w_m = 1\} = 1 \mod 2$. En outre, $(u \oplus v)_m = u_m + v_m = 1$ si et seulement si ($u_m = 0$ et $v_m = 1$) ou ($u_m = 1$ et $v_m = 0$). On a donc $\mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, (u \oplus v)_m = 1\} = \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, u_m = 1\,,\, v_m = 0\} + \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, u_m = 0\,,\, v_m = 1\} = \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, u_m = 1\} - \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, u_m = 1\,,\, v_m = 1\} + \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, v_m = 1\} - \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, v_m = 1\,,\, u_m = 1\} = L_{k,p}(u) + L_{k,p}(v) - 2\,\mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, v_m = 1\,,\, u_m = 1\} \mod 2 = L_{k,p}(u) + L_{k,p}(v) \mod 2$ donc $L_{k,p}(u \oplus v) = \mathrm{Card}\{m \in [\![0, |u|-1]\!] \,/\, m = k \mod p\,,\, (u \oplus v)_m = 1\} \mod 2 = L_{k,p}(u) + L_{k,p}(v)$.

Q21.

a. Par contraposition de la question 19, $A_{0,p}$ ne reconnaît pas w donc $L_{0,p}(w) = 0$.

b. $L_{0,2}(w) = 0$ donc le nombre de 1 en position paire de w est pair donc le nombre de 1 en position impaire de $0w$ est pair donc $L_{1,2}(w) = 0$. De même, le nombre de 1 en position impaire de w est pair donc le nombre de 1 en position paire de $0w$ est pair donc $L_{0,2}(w) = 0$. Pour tout $0 \le k \le p-1$, $\mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = k \mod p\,,\, w_m = 1\}$ est pair. Soit $k \in [\![1, p-1]\!]$. Soit $w' = 0w$. $\mathrm{Card}\{m \in [\![0, |w'|-1]\!] \,/\, m = k \mod p\,,\, w'_m = 1\} = \mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = k-1 \mod p\,,\, w_m = 1\}$ donc ce nombre est pair donc $L_{k,p}(w') = 0$.

Soit $n \in \mathbb{N}$. Supposons que $L_{0,2}(0^n w) = L_{1,2}(0^n w) = L_{k,p}(0^n w) = 0$ pour tout $k \in [\![1, p-1]\!]$. D'après la question a, $L_{0,p}(0^n w) = 0$. D'après le raisonnement précédent appliqué au mot $w_n = 0^n w$, on obtient que $L_{0,2}(0w_n) = L_{1,2}(0w_n) = L_{k,p}(0w_n) = 0$ pour tout $k \in [\![1, p-1]\!]$ donc $L_{0,2}(0^{n+1}w) = L_{1,2}(0^{n+1}w) = L_{k,p}(0^{n+1}w)$ pour tout $k \in [\![1, p-1]\!]$. Par récurrence, pour tout $w' \in 0^\star w$, $L_{0,2}(w') = L_{1,2}(w') = L_{k,p}(w') = 0$ pour tout $k \in [\![1, p-1]\!]$.

Q22. Soient $w \in \Sigma^\star$, $n \in \mathbb{N}$ et $w' = w0^n$. Pour tout $m \ge |w|$, $w'_m = 0$ donc $\mathrm{Card}\{m \in [\![0, |w'|-1]\!] \,/\, m = k \mod p\,,\, w'_m = 1\} = \mathrm{Card}\{m \in [\![0, |w|-1]\!] \,/\, m = k \mod p\,,\, w_m = 1\}$ donc $L_{k,p}(w) = L_{k,p}(w')$.