Matthieu Boyer
Planche A

Automates et monoïdes

Un semigroupe est un couple $(E, \cdot)$ où E est un ensemble et $\cdot$ une loi de multiplication interne associative. Un monoïde est un semigroupe qui dispose d'un élément neutre e.

1. **Préliminaires**
   (a) Montrez qu'un monoïde possède un unique élément neutre.
   (b) Pour A un alphabet fini et $\cdot$ la concaténation, montrez que $(A^*, \cdot)$ est un monoïde.
2. **Monoïdes des transformations**
   Soit Q un ensemble fini. Montrez que l'ensemble des fonctions de $Q \to Q$ est un monoïde. Montrez que tout monoïde fini $M = (E, \cdot)$ est un sous-monoïde d'un monoïde des transformations (c'est-à-dire qu'il existe un morphisme injectif de M vers un monoïde de transformations).
3. **Monoïdes des transitions**
   Soit $\mathcal{A} = (Q, A, \delta, i, F)$ un automate déterministe fini. On note $q \cdot u$ l'état atteint depuis q par le mot $u \in A^*$. Les fonctions de transition sont les fonctions $f_u : Q \to Q$ définies par $f_u(q) = q \cdot u$.
   (a) Montrez que l'ensemble des fonctions de transition de $\mathcal{A}$ forme un monoïde fini.
   (b) Quelle est la complexité de vérifier qu'une fonction $f : Q \to Q$ est dans ce monoïde ?
   (c) Proposez un algorithme polynomial en $|Q|$ et N (où N est la taille du monoïde) pour énumérer ces fonctions.
   (d) Montrez que le monoïde des transitions reconnaît le langage $L(\mathcal{A})$.
4. **Reconnaissabilité et régularité**
   Montrez qu'un langage est reconnu par un monoïde fini si et seulement s'il est calculable par un automate fini.
5. **Monoïdes des relations**
   Soit Q un ensemble fini. Une relation sur Q est un sous-ensemble de $Q \times Q$.
   (a) Montrez que l'ensemble des relations sur Q, muni de la composition $\circ$, est un monoïde.
   (b) Donnez la taille de ce monoïde en fonction de $|Q|$.
6. **Automates non déterministes**
   Soit $\mathcal{A} = (Q, A, \delta, I, F)$ un NFA. On appelle relation de transition associée à $u \in A^*$ la relation $R_u = \{(q, q') \mid q' \in q \cdot u\}$.
   (a) Montrez que l'ensemble de ces relations forme un monoïde fini qui reconnaît $L(\mathcal{A})$.
   (b) Donnez la complexité du calcul de ce monoïde.
7. **Borne inférieure**
   Soit L un langage régulier dont le monoïde syntaxique est de taille n. Montrer que tout automate non déterministe calculant L est de taille au moins $\sqrt{\log_2 n}$.

Matthieu Boyer

Solution: Automates et monoïdes

Ulm 2022 Sujet C1

1. **Préliminaires**
   (a) Soient e et e' deux éléments neutres. Par définition de e, $e \cdot e' = e'$. Par définition de e', $e \cdot e' = e$. Donc $e = e'$.
   (b) La concaténation est associative : $u \cdot (v \cdot w) = (u \cdot v) \cdot w$. Le mot vide ε est l'élément neutre car $\varepsilon \cdot u = u \cdot \varepsilon = u$.
2. **Transformations**
   L'ensemble Q^Q muni de la composition $\circ$ est un monoïde (l'identité id_Q est le neutre). Pour M, on définit $\varphi : M \to E^E$ par $\varphi(m) = f_m$ où $f_m(x) = m \cdot x$. - Morphisme : $f_{m \cdot n}(x) = (m \cdot n) \cdot x = m \cdot (n \cdot x) = f_m(f_n(x))$, donc $\varphi(m \cdot n) = \varphi(m) \circ \varphi(n)$. - Injectif : Si $f_m = f_n$, alors $f_m(e) = f_n(e) \Longrightarrow m \cdot e = n \cdot e \Longrightarrow m = n$.
3. **Transitions**
   (a) L'application $\psi : A^* \to Q^Q$ qui à u associe f_u est un morphisme de monoïdes. L'image $\psi(A^*)$ est donc un sous-monoïde de Q^Q. Comme Q est fini, Q^Q est fini ($|Q|^{|Q|}$), donc le monoïde est fini.
   (b) Vérifier si $f \in \psi(A^*)$ revient à un parcours de graphe (accessibilité) dans l'espace des fonctions Q^Q, ce qui peut être exponentiel.
   (c) **Algorithme par saturation** : On part de $S = \{id_Q, f_a \mid a \in A\}$. On calcule $S \cdot \{f_a \mid a \in A\}$ et on ajoute les nouvelles fonctions jusqu'à stabilisation. Complexité : $O(N \cdot |A| \cdot |Q|)$.
   (d) $u \in L(\mathcal{A}) \Longleftrightarrow f_u(i) \in F$. Soit $P = \{f \in M \mid f(i) \in F\}$. Alors $L = \psi^{-1}(P)$, donc M reconnaît L.
4. **Équivalence**
   ($\Leftarrow$) Fait en 3.d. ($\Rightarrow$) Soit $\varphi : A^* \to M$ reconnaissant L. On construit $\mathcal{A} = (M, A, \delta, e, \varphi(L))$ avec $\delta(m, a) = m \cdot \varphi(a)$. Cet automate fini calcule L.
5. **Relations**
   (a) La composition est associative et la relation identité $Id = \{(q, q) \mid q \in Q\}$ est le neutre.
   (b) Une relation est un sous-ensemble de $Q \times Q$, il y en a $2^{|Q|^2}$.
6. **NFA**
   (a) Similaire au cas déterministe en remplaçant les fonctions par des relations. $u \in L(\mathcal{A}) \Longleftrightarrow R_u \cap (I \times F) \neq \varnothing$.
   (b) Algorithme de saturation sur les matrices booléennes. Complexité : $O(2^{|Q|^2} \cdot |A| \cdot |Q|^3)$.
7. **Borne inférieure**
   Soit un NFA à N états reconnaissant L. Son monoïde de relations M_{rel} est de taille au plus 2^{N^2}. Par définition, le monoïde syntaxique est le plus petit monoïde reconnaissant L, donc $n \leq |M_{rel}| \leq 2^{N^2}$. En passant au log : $\log_2 n \leq N^2 \Longrightarrow N \geq \sqrt{\log_2 n}$.

Matthieu Boyer
Planche B

Cribles

On dit qu'un entier naturel est *sans carré* s'il n'est pas divisible par le carré d'un entier supérieur ou égal à 2. On note $\pi(n)$ le nombre de nombres premiers inférieurs ou égaux à n.

1. Donner le pseudo-code d'un algorithme permettant de calculer le nombre d'entiers naturels sans carré inférieurs ou égaux à n. Quelles sont ses complexités ?
2. Donner le pseudo-code d'un algorithme permettant de calculer $\pi(n)$. Quelles sont ses complexités ?
3. Décrire une structure de données permettant de représenter un sous-ensemble $E \subseteq [\![1, n]\!]$ supportant l'initialisation en $O(n)$, la suppression en $O(1)$ et l'énumération croissante en $O(|E|)$.
4. En déduire le pseudo-code d'un algorithme (Crible d'Euler) calculant $\pi(n)$ en temps $O(n)$.
5. Donner le pseudo-code d'un algorithme calculant tous les $\text{pgcd}(p, q)$ pour $p, q \in [\![1, n]\!]$ en temps $O(n^2)$.
6. Soit $\Phi(n)$ le nombre de paires $(p, q) \in [\![1, n]\!]^2$ telles que $\text{pgcd}(p, q) = 1$. Démontrer :

$$\Phi(n) = n^2 - \sum_{d=2}^{n} \Phi\left(\left\lfloor \frac{n}{d} \right\rfloor\right)$$

En déduire un algorithme calculant $\Phi(n)$ en temps sous-linéaire.

Solution: Cribles

1. **Nombre d'entiers sans carré** : On adapte le crible d'Ératosthène.

   $S \leftarrow$ tableau de taille $n+1$ contenant Vrai
   for x allant de 2 à $\lfloor\sqrt{n}\rfloor$ **do**
   for xx de x^2 à n par pas de x^2 **do**
   $S[xx] \leftarrow$ Faux
   return nombre de Vrai dans $S[1 \dots n]$

   Complexité : Temps $O(n\sum \frac{1}{x^2}) = O(n)$, Espace $O(n)$.

2. **Crible d'Ératosthène ($\pi(n)$)** :

   $P \leftarrow$ tableau de taille $n+1$ contenant Vrai
   $cnt \leftarrow 0$
   for p de 2 à n **do**
   if $P[p]$ **then**
   $cnt \leftarrow cnt + 1$
   for pp de $2p$ à n par pas de p **do**
   $P[pp] \leftarrow$ Faux
   return cnt

   Complexité : Temps $O(n \ln\ln n)$, Espace $O(n)$.

3. **Structure de données** : On utilise une **liste doublement chaînée** représentée par deux tableaux `suiv` et `prec` de taille $n+2$. L'élément i a pour voisin de droite `suiv`[i] et de gauche `prec`[i]. La suppression de i se fait en $O(1)$ par : `suiv[prec[i]]` $\leftarrow$ `suiv[i]` et `prec[suiv[i]]` $\leftarrow$ `prec[i]`.

4. **Crible d'Euler (Temps $O(n)$)** : On élimine chaque composé exactement une fois par son plus petit facteur premier.

   Init structure question 3 pour $E = [\![2, n]\!]$
   $p \leftarrow 2$, $cnt \leftarrow 0$
   while $p \leq n$ **do**
   $cnt \leftarrow cnt + 1$, $x \leftarrow p$
   while $x \cdot p \leq n$ **do**
   Supprimer$(x \cdot p)$
   $x \leftarrow$ suiv$[x]$
   $p \leftarrow$ suiv$[p]$

5. **PGCD par mémoïsation** :

   $res \leftarrow$ matrice $n \times n$ init à -1
   function PGCD(p, q)
   if $p = 0$ ou $q = 0$ **then return** $p + q$
   if $res[p][q] = -1$ **then**
   if $p < q$ **then** $res[p][q] \leftarrow \text{pgcd}(p, q-p)$
   else$res[p][q] \leftarrow \text{pgcd}(p-q, q)$
   return $res[p][q]$

   Chaque couple (p, q) n'est calculé qu'une fois. Temps $O(n^2)$.

6. **Calcul de $\Phi(n)$** : **Preuve** : L'ensemble $[\![1, n]\!]^2$ de cardinal n^2 est la réunion disjointe des ensembles $A_d = \{(p, q) \mid \text{pgcd}(p, q) = d\}$ pour $1 \leq d \leq n$. Or $(p, q) \in A_d \Longleftrightarrow p = dp', q = dq'$ avec $\text{pgcd}(p', q') = 1$ et $1 \leq p', q' \leq \lfloor n/d \rfloor$. Donc $|A_d| = \Phi(\lfloor n/d \rfloor)$. D'où $n^2 = \sum_{d=1}^{n} \Phi(\lfloor n/d \rfloor)$.

   Algorithme sous-linéaire : On utilise la formule par blocs et un dictionnaire pour la mémoïsation.

   function Phi(n)
   if n dans dico **then return** dico$[n]$
   $res \leftarrow n^2$
   for $d = 2$ à $\lfloor\sqrt{n}\rfloor$ **do** $res \leftarrow res - \text{Phi}(\lfloor n/d \rfloor)$

for $v = 1$ à $\lceil \sqrt{n} \rceil - 1$ **do**
$res \leftarrow res - (\lfloor n/v \rfloor - \lfloor n/(v+1) \rfloor) \cdot \text{Phi}(v)$
dico$[n] \leftarrow res$ **return** res

Complexité : Temps $O(n^{3/4})$, Espace $O(\sqrt{n})$ (nombre de valeurs distinctes de $\lfloor n/d \rfloor$).