Introduction à l'algorithmique **Hiver 2018**

IFT2125-6001 TA: Stéphanie Larocque

Démonstration 1

Corrigés légèrement modifiés de Maelle Zimmermann

1

Question: Implémenter en python un algorithme pour calculer le plus petit commun multiple de deux nombres a et b.

Solution: Le plus petit commun multiple (ppcm) de deux nombres a et b est donné par le produit $a \times b$ divisé par le plus grand commun diviseur (pgcd) de a et b. On peut implémenter les deux algorithmes suivants en python.

```
#Algorithme d'Euclide pour plus grand diviseur commun
def pgcd(a, b):
   while b:
      a, b = b, a % b
   return a

#Algorithme pour plus petit multiple commun
def ppcm(a, b):
   return a * b // pgcd(a, b)
```

2

Question: Déclarer une liste, un tuple et un set contenant les éléments 1,2,3,4 en python et énoncer les principales différences. Implémenter une méthode de tri d'une liste.

Solution: Un set est un ensemble non ordonné d'éléments. Une liste et un tuple sont une séquence d'éléments, donc chaque élément correspond à un indice. A noter qu'en python le premier élément est indexé par 0 et non 1. La principale différence entre un tuple et une liste est qu'un tuple ne peut être modifié une fois qu'il est déclaré.

```
list1=[1,2,3,4]
tup1=(1,2,3,4)
set1={1,2,3,4}

#On peut aussi declarer un tuple ou un set a partir d'une liste
tup2=tuple([1,2,3,4])
set2=set([1,2,3,4])

#Algorithme du tri par insertion
def insertionsort(alist)
   for i in range(1,len(alist))
      x = alist[i]
      j = i-1
      while j >= 0 and x < alist[j]
         alist[j+1] = alist[j]
         j = j-1
      alist[j+1] = x
   return alist
```

3

Question: Implémenter en python un algorithme pour calculer naïvement $\det(A)$.

Solution: La formule naïve pour calculer le déterminant d'une matrice A de taille $m \times m$ est:

$$\det(A) = \sum_{j=1}^{m} (-1)^{i+j}\ a_{ij}\ \det(A_{i,j})$$

où a_{ij} est le $j^{\text{ème}}$ élément de la $i^{\text{ème}}$ ligne de A, et $A_{i,j}$ est la matrice obtenue en effaçant la $i^{\text{ème}}$ ligne et la $j^{\text{ème}}$ colonne de A.

La ligne i dans la formule ci-dessus peut être choisie aléatoirement. Nous prenons par défaut la première ligne de A, ainsi nous fixons $i = 0$ dans l'algorithme python. Nous implémentons une fonction pour calculer la sous-matrice $A_{i,j}$, et une fonction qui calcule récursivement le déterminant de la matrice A.

```
def submatrix(A, i, j):
   return [[A[x][y] for y in range(len(A)) if y != j]
   for x in range(len(A)) if x != i]

#Algorithme naif du calcul de determinant
def det(A):
   i = 0  #indice de ligne fixe
```

```
s = 0
if len(A) == 1:
   return A[0][0]
else:
   for j in range(len(A)):
      s += (-1)**(i+j) * A[i][j] * det(submatrix(A, i, j))
      return s
```

4

Question: Prouver que:

1. $n^2 + n \in O(n^3)$
2. $n^2 \in \Omega(n \log(n))$
3. $2^{n+1} \in \Theta(2^n)$
4. $n^6 - n^5 + n^4 \in \Theta(n^6)$
5. $\log(n) \in O(\sqrt{n})$

Solution: Il y a souvent plusieurs façons de faire ces preuves. Nous allons en voir quelques unes. Notons d'abord que l'on peut utiliser les implications suivantes:

i. $\lim_{n\to\infty} \frac{f(n)}{g(n)} \in \mathbb{R}^+ \quad \Rightarrow \quad O(f(n)) = O(g(n)).$

ii. $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0 \quad \Rightarrow \quad O(f(n)) \subset O(g(n)).$

iii. $\lim_{n\to\infty} \frac{f(n)}{g(n)} = +\infty \quad \Rightarrow \quad \Omega(f(n)) \subset \Omega(g(n)).$

1. On peut simplement utiliser la définition de $O(f(n))$:

$$O(f(n)) = \{t : \mathbb{N} \to \mathbb{R}^* \mid \exists n_0 \in \mathbb{N}, \exists c \in \mathbb{R}^+ : \forall n \geq n_0,\ t(n) \leq cf(n)\}$$

$$\underbrace{n^2 + n \leq n^2 + n^2}_{\forall n \geq 1} = \underbrace{2n^2 \leq 2n^3}_{\forall n \geq 1}.$$

Ainsi $\exists n_0 = 1$ et $c = 2$ tel que $\forall n \geq n_0$, on a $n^2 + n \leq cn^3$, et donc $n^2 + n \in O(n^3)$.

2. On utilise la règle de l'Hôpital:

$$\lim_{n\to\infty} f(n)/g(n) = \lim_{x\to\infty} f'(x)/g'(x)$$

   On calcule la limite:

$$\lim_{n\to\infty} \frac{n^2}{n\log(n)} = \lim_{n\to\infty} \frac{n}{\log(n)} = \lim_{n\to\infty} \frac{1}{1/n} = +\infty.$$

   Ainsi $\Omega(n^2) \subset \Omega(n\log(n))$, donc $n^2 \in \Omega(n\log(n))$.

3. On calcule la limite:

$$\lim_{n\to\infty} \frac{2^{n+1}}{2^n} = \lim_{n\to\infty} \frac{2}{1} = 2.$$

   Ainsi $O(2^{n+1}) = O(2^n)$, donc $2^{n+1} \in \Theta(2^n)$.

4. Alternativement au calcul de limite qui nécessiterait d'appliquer plusieurs fois la règle de l'Hôpital, on peut faire:

$$\begin{aligned} O(n^6 - n^5 + n^4) &= O(\frac{1}{2}n^6 + (\frac{1}{2}n^6 - n^5) + n^4) \\ &= \underbrace{O(\max\{\frac{1}{2}n^6, \frac{1}{2}n^6 - n^5, n^4\})}_{\text{car } 1/2n^6 - n^5 \geq 0,\ \forall n \geq 2} \\ &= O(\frac{1}{2}n^6) = O(n^6). \end{aligned}$$

   Ainsi $n^6 - n^5 + n^4 \in \Theta(n^6)$.

5. On calcule la limite en utilisant la règle de l'Hôpital:

$$\lim_{n\to\infty} \frac{\log(n)}{\sqrt{n}} = \lim_{n\to\infty} \frac{1/n}{\frac{1}{2}n^{-\frac{1}{2}}} = \lim_{n\to\infty} \frac{2}{n^{\frac{1}{2}}} = 0.$$

   Ainsi $O(\log(n)) \subset O(\sqrt{n})$ et donc $\log(n) \in O(\sqrt{n})$.

5

Question: Prouver par induction que les permutations (12) et $(12\ldots m)$ engendrent S_m, l'ensemble des permutations de $\{1, 2, \ldots m\}$.

Rappelons tout d'abord (par exemple) que la permutation $\sigma = (1\ 3\ 4)$ envoie $\sigma(1) = 3, \sigma(3) = 4$ et $\sigma(4) = 1$ et laisse tous les autres éléments inchangés/fixés. On a d'ailleurs que (1 3 4) = (3 4 1) = (4 1 3), mais n'est pas équivalent à (4 3 1) (qui correspond plutôt à l'inverse de σ). Aussi, notons que la convention utilisée dans le cours pour la composition de permutation est de gauche à droite et n'est pas commutative:

$$(1\ 3\ 4)(4\ 5\ 6) = (1\ 3\ 5\ 6\ 4) \neq (4\ 5\ 6\ 1\ 3) = (4\ 5\ 6)(1\ 3\ 4)$$

Solution: Nous prouvons la proposition en deux parties. D'abord, nous montrons que (12) et $(12\dots m)$ génèrent l'ensemble des transpositions (permutations qui échangent deux éléments et préservent tous les autres), puis nous prouvons par induction sur m que l'ensemble de ces transpositons engendrent S_m.

Dans un premier temps, nous prouvons que toutes les transpositions de $\{1, 2, \dots m\}$ sont engendrées par (12) et $(12\dots m)$. En effet, nous constatons d'abord que toute transposition du type $(k\ k{+}1)$ où $1 \leq k \leq m{-}1$ est engendrée par (12) et $\gamma = (12\dots m)$:

$$\begin{cases} \gamma^{-1}(12)\gamma = (23) \\ \gamma^{-1}(23)\gamma = (34) \\ \vdots \\ \gamma^{-1}(m-2\ m-1)\gamma = (m-1\ m). \end{cases}$$

Puis nous prouvons que toute transposition de la forme $(1\ k)$ pour $2 \leq k \leq m$ s'écrit comme produit de transpositions du type précédent. En effet, pour $k \geq 3$ nous avons

$$(1\ k) = (k\ k-1)(1\ k-1)(k\ k-1).$$

Finalement il reste à constater que toute transposition $(x\ y)$ peut s'écrire comme le produit $(1\ x)(1\ y)(1\ x)$ pour conclure que toute transposition de $\{1, 2, \dots m\}$ peut être engendrée par (12) et $(12\dots m)$.

Maintenant, montrons que l'ensemble des transpositions engendre S_m par induciton sur m:

<u>Cas de base: m=2:</u> L'ensemble S_2 ne contient que la permutation identité et la permutation (12). Comme Id = (12)(12), c'est vrai.

<u>Etape d'induction:</u> Soit $m > 2$. Supposons que la proposition est vraie pour S_m. Soit une permutation $\sigma \in S_{m+1}$.

- Si σ laisse $m+1$ fixe, alors la restriction de σ à $\{12\dots m\}$ est engendrée par des transpositions (qui laissent $m+1$ fixe) par hypothèse d'induction.
- Sinon, $\sigma(m+1) = y \neq m+1$. Soit la transposition $\tau = (m+1\ y)$, alors la permutation $\sigma\tau$ fixe $m+1$ et comme précédemment elle est engendrée par des transpositions. Comme $\sigma = \sigma\tau\tau^{-1} = (\sigma\tau)\tau$, on conclut que σ est engendrée par des transpositions.