Algorithmique III. L2 Informatique I41.

TD 3. Complexité et notations asymptotiques[1]

Dans tous les exercices, espace $\mathscr{F}$ désigne l'ensemble des fonctions de $\mathbb{R}_+^*$ dans $\mathbb{R}_+^*$ dont la variable est dénotée n.

EXERCICE 1. Calculez les fonctions de complexité des 4 algorithmes ci-dessous en fonction des paramètres n et m en entrée.

```
ALGORITHME A(n,m)
DONNEES
 · n,m:entiers
VARIABLES
 · i,j:entiers
DEBUT
 · i ← 0
 · j ← 0
 · TQ (i < n) ET (j < m) FAIRE
 ·   · i ← i + 1
 ·   · j ← j + 1
 · FTQ
FIN
```

```
ALGORITHME B(n,m)
DONNEES
 · n,m:entiers
VARIABLES
 · i,j:entiers
DEBUT
 · i ← 0
 · j ← 0
 · TQ (i < n) OU (j < m) FAIRE
 ·   · i ← i + 1
 ·   · j ← j + 1
 · FTQ
FIN
```

```
ALGORITHME C(n,m)
DONNEES
 · n,m:entiers
VARIABLES
 · i,j:entiers
DEBUT
 · i ← 0
 · j ← 0
 · TQ (j < m) FAIRE
 ·   · SI (i < n) ALORS
 ·   ·   · i ← i + 1
 ·   · SINON   ·
 ·   ·   · j ← j + 1
 ·   · FSI
 · FTQ
FIN
```

```
ALGORITHME D(n,m)
DONNEES
 · n,m:entiers
VARIABLES
 · i,j:entiers
DEBUT
 · i ← 0
 · j ← 0
 · TQ (j < m) FAIRE
 ·   · SI (i < n) ALORS
 ·   ·   · i ← i + 1
 ·   · SINON
 ·   ·   · j ← j + 1
 ·   ·   · i ← 0
 ·   · FSI
 · FTQ
 FIN
```

En toute rigueur, les fonctions de complexité de ces algorithmes devraient être exprimées avec quelles paramètres ?

1. Version du 5 juillet 2022, 12 : 15

EXERCICE 2. Donnez des exemples de fonctions qui sont en $\Omega(n^2)$ mais pas en $\Theta(n^2)$.

EXERCICE 3. Soit f et g deux fonctions de $\mathscr{F}$. Démontrez que

$$f = \Theta(g) \ \Leftrightarrow\ (f = O(g)) \ \wedge\ (f = \Omega(g)). \tag{1}$$

EXERCICE 4. Soit f et g deux fonctions de $\mathscr{F}$. Démontrez que

$$f = O(g) \Leftrightarrow g = \Omega(f). \tag{2}$$

EXERCICE 5. Soit $f \in \mathscr{F}$. Montrez que si f est définie par $f(n) := 3n^2+7$, alors $f = O(n^2)$ et que si f est définie par $f(n) := 3n^3 - 2n + 1$, alors $f = \Omega(n^3)$.

EXERCICE 6. Montrez que si f est une fonction polynomiale de degré d à coefficient dominant positif (i.e. le monôme de plus haut degré a un coefficient positif), alors

$$f = \Theta(n^d). \tag{3}$$

EXERCICE 7. Soit $g \in \mathscr{F}$. Montrez que si $k \in \mathbb{R}$ est une constante strictement positive, alors

$$k \times O(g) = O(g)$$

mais qu'en revanche

$$n \times O(g) = O(ng).$$

EXERCICE 8. Soit $n \in \mathbb{N}$. Calculez et/ou simplifiez :

(1) $\Theta(n) + \Theta(n)$
(2) $O(n) + \Theta(n)$
(3) $\Theta(n) + \Theta(1)$
(4) $O(n) + \Omega(n)$
(5) $O(n)\,O(n)$
(6) $n\,\Theta(1)$
(7) $O(n) + O(n^2)$
(8) $\Theta(\log_2(n) + n)$

EXERCICE 9. Démontrez que la fonction définie par $n \mapsto n^2$ est en $o(n^3)$.

EXERCICE 10. (1) Démontrez que la somme de la série de terme général 2^{-n} est égale à 2.
(2) Démontrez que la série harmonique de terme général $\frac{1}{n}$ n'est pas convergente en utilisant une minoration par une intégrale.
(3) En remarquant que

$$1+\frac{1}{2}+\underbrace{\frac{1}{3}+\frac{1}{4}}_{>\frac{1}{4}+\frac{1}{4}}+\underbrace{\frac{1}{5}+\frac{1}{6}+\frac{1}{7}+\frac{1}{8}}_{>\frac{1}{8}+\frac{1}{8}+\frac{1}{8}+\frac{1}{8}}+\cdots$$

démontrez d'une autre façon que cette série est divergente.

EXERCICE 11. Deux algorithmes A et B sur la machine RAM résolvent le même problème et ont respectivement pour complexité moyenne les fonctions $\overline{T}_A(n) := 1\,220n + 654$ et $\overline{T}_B(n) := \frac{1}{4}n^2 + 4n - 63$ où n désigne la taille des données à traiter. Quelle est la plus petite valeur de n pour laquelle l'algorithme A est plus efficace que l'algorithme B ?