

Information, Calcul et Communication

Compléments de cours

Examen 1 2018 Q5

Quelle est la complexité de l'algorithme suivant :

algo5
entrée : <i>Entier positif n</i>
sortie : <i>valeur</i>
Si $n \leq 2$ Sortir : 3 Sinon Sortir : $4 \log(\text{algo5}(n-2)) + 1$

$\Theta(1)$ a
 $\Theta(1)$
 $\in \Theta(n)$

$\Theta(n)$ $\Theta(\lg n)$ $\Theta(n^2)$

3 méthodes :

1- Compter :

$$C(n) = a + C(n-2) + b$$

$$u_n = \lambda + u_{n-2} \rightarrow \text{arith.}$$

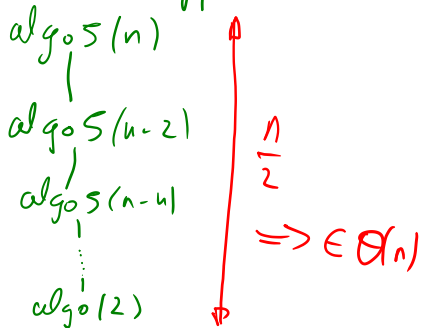
$$\Rightarrow C(n) \in \Theta(n)$$

Examen 1 2018 Q5

Quelle est la complexité de l'algorithme suivant :

algo5
entrée : <i>Entier positif n</i> sortie : <i>valeur</i>
Si $n \leq 2$ Sortir : 3 Sinon Sortir : $4 \log(\text{algo5}(n-2)) + 1$

2 - arbre des appels



3 - incrémenter et compter

$$\text{augment. algo5}(n+2) = 1 \Rightarrow \in \mathcal{O}\left(\int 1 du\right) = \mathcal{O}(n)$$

Examen 1 2018 Q5

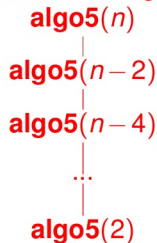
Quelle est la complexité de l'algorithme suivant :

algo5
entrée : <i>Entier positif n</i> sortie : <i>valeur</i>
Si $n \leq 2$ Sortir : 3 Sinon Sortir : $4 \log(\text{algo5}(n-2)) + 1$

👉 $\Theta(n)$

Rappel : vous avez trois moyens :

1. *Compter* les instructions :
 $C(n) = a + C(n-2) + b$
2. *dessiner* le graphe des appels :



3. « incrémenter et compter » :
 $\Delta C(n+2) = \text{cte}$

Examen 1 2018 Q15

Cette question est la suite de celle (Q13) posée la semaine passée que je reproduis donc ici :

On s'intéresse ici à raccourcir les répétitions de trois ou plus valeurs identiques successives ; par exemple à produire la liste (6,6,4,4,12,4,6) à partir de la liste (6,6,4,4,4,12,4,6) en supprimant le 4 en cinquième position car il est présent trois fois consécutives.

À noter que :

- ▶ les seules valeurs supprimées sont celles qui sont répétées **successivement** trois fois ou plus (l'une **à la suite** de l'autre) ; on ne garde alors que deux de ces valeurs (cf la valeur 4 ci-dessus) ;
- ▶ toute valeur présente une ou deux fois successivement est préservée, et l'on conserve l'ordre de la liste ;
- ▶ en sortie on ne peut donc pas avoir plus de deux valeurs identiques consécutives.

Écrivez un algorithme **récuratif** pour résoudre ce problème.

Indication : on pourra séparer le premier élément et le reste de la liste.

~~NON!~~

algo

Examen 1 2018 Q15 (solution)

Entrée : une liste L

Sortie : liste « simplifiée »

$n \leftarrow \text{taille}(L)$

Si $n \leq 2$ Sortir L

$L' \leftarrow \text{algo}(L[2], \dots, L[n])$

Si $L[1] = L'[1]$ et $L[1] = L'[2]$

Sortir L'

Sortir $L[1] \oplus L'$

algo autre

Entrée : $L, \underline{i}$

Sortie :

$\vdots$

algo
entrée L

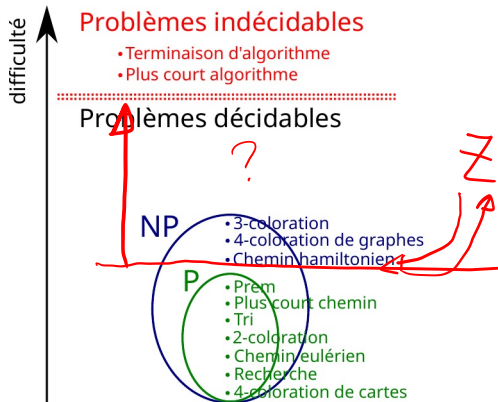
Sortir algo-autre(
 $L, \text{taille}(L)$)

Examen 1 2018 Q16

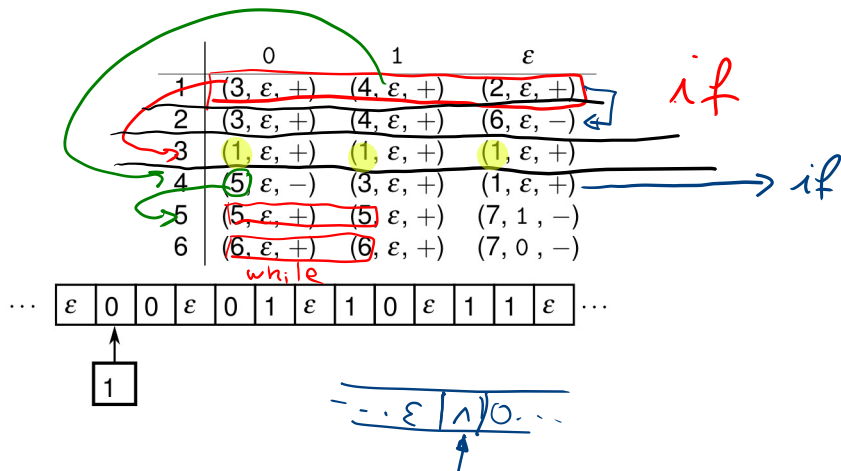
Déterminez la complexité de votre algorithme (question précédente).
Justifiez votre réponse.

Leçon I.3 (calculabilité) – Points clés

- ▶ qu'est-ce que *une* machine de Turing et **la** machine de Turing universelle
- ▶ problème = une question (portant sur ses entrées) et (une infinités d') instances
- ▶ il y a beaucoup de problèmes non calculables
exemple : problème de l'arrêt
- ▶ P et NP
- ▶ on ne sait pas si NP est inclu dans P
- ▶ la 2- et la 4-coloration de graphes **planaires** sont dans P
la 3-coloration de graphes est dans NP
- ▶ et d'autres...



Leçon I.3 (calculabilité) – Machines de Turing



Leçon I.3 (calculabilité) – P et NP

Si un algorithme résoud un problème X (dont la taille de l'entrée est n) en $\Theta(n^2 \log n)$, que peut on dire ?

Si un algorithme permet de vérifier en $\Theta(n^2 \log n)$ qu'une donnée de taille n est bien une instance positive (« solution ») à un problème Y , moyennant une donnée supplémentaire (certificat) de taille en $\Theta(\log n)$, que peut on dire ?

Un tel problème est-il dans P ?

Peut-il être dans P ?

1. X est dans P
2. Y est dans NP
3. pas forcément
4. oui c'est possible

Leçon I.3 (calculabilité) – Complexité des problèmes

Un algorithme possible pour résoudre un problème Z est le suivant :

1. transformer les données de Z en un graphe
2. trouver un chemin hamiltonien dans ce graphe
3. retransformer ce chemin dans les données de Z , puis conclure

Que peut-on dire de Z ?

1. rien ➡ **en effet pas grand chose au niveau du cours ICC**
2. qu'il est dans P ➡ **on ne sait pas, en tout cas pas avec cet algorithme car le problème du chemin hamiltonien est dans NP**
3. qu'il est dans NP ➡ **on ne sait pas, rien ne dit qu'on puisse en vérifier une solution (instance positive) en un temps polynomial**
4. qu'il n'est pas dans P ➡ **ce serait bien présomptueux (savez vous si NP est dans P ou non ?)**
5. (qu'il est NP -difficile) ➡ **c'est en fait la définition de NP -difficile, HORS COURS**