

EA4 – Éléments d'algorithmique Interrogation n° 2 – Sujet A

Durée : 1h30

Aucun document autorisé, appareils électroniques éteints et rangés.

Le sujet est (trop) long, il en sera tenu compte dans la notation. Les exercices sont indépendants et ne sont absolument pas classés par ordre de difficulté.

*Sauf mention contraire, on s'intéresse à la complexité **dans le pire des cas**.*

Exercice 1 :

On considère les permutations $\sigma = 3\ 4\ 6\ 8\ 5\ 7\ 1\ 2$ et $\tau = 5\ 3\ 4\ 2\ 1\ 6\ 8\ 7$.

Donner les représentations en produit de cycles disjoints de σ et τ .

Calculer les inverses de σ et τ , ainsi que les produits $\sigma\tau$ et $\tau\sigma$ (sous la forme que vous préférez).

Donner deux décompositions différentes de σ en produit de transpositions.

Exercice 2 :

Cocher les assertions exactes.

		$f \in \Theta(g)$	$f \notin \Theta(g)$	$f \in \Omega(g)$	$f \in O(g)$
$f = n$	$g = n^2$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = \log n$	$g = \log(n^2)$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = \log(n^2)$	$g = (\log n)^2$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = 2^n$	$g = 2^{(n^2)}$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = n!$	$g = 2^n$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = n!$	$g = n^n$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = \log(n!)$	$g = \log(2^n)$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$f = \log(n!)$	$g = \log(n^n)$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Exercice 3 :

On s'intéresse au problème suivant : étant donné une liste L de n entiers, déterminer si L contient trois valeurs consécutives, *i.e.* s'il existe un entier i tel que L contient i , $i + 1$ et $i + 2$.

Décrire un algorithme naïf permettant de résoudre ce problème et donner sa complexité.

[illegible]

Proposer un algorithme de complexité en temps strictement meilleure. Justifier.

[illegible]

En déduire un algorithme `maximum(T)` de complexité optimale qui renvoie le plus grand élément de T . Justifier rapidement sa correction et sa complexité.

Exercice 5 :

Dans cet exercice, on manipule des ensembles d'entiers représentés par des listes triées sans doublon. Écrire un algorithme `est_inclus_dans(E, F)` aussi efficace que possible qui teste si E est inclus dans F , c'est-à-dire si *tous* les éléments de E appartiennent à F .

Quelle est la complexité (en temps et en espace) de cet algorithme ?

Exercice 6 :

Soit T le tableau suivant :

14	7	4	13	8	5	11	1
----	---	---	----	---	---	----	---

Appliquer l'algorithme de tri fusion (*MergeSort*) à **T**. Combien de comparaisons d'éléments sont effectuées (exactement)?

Appliquer l'algorithme de tri rapide (*QuickSort*) à T dans sa version simple (pas en place, avec T[0] comme pivot). Combien de comparaisons d'éléments sont effectuées (exactement) ?

This image shows a single sheet of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page. There are approximately 20 lines visible. The paper has a slight shadow on the right side, suggesting it's resting on a surface.

Exercice 7 :

Écrire un algorithme `petitsElements(L, k)`, inspiré de *QuickSort* et *QuickSelect*, retournant la liste des k plus petits éléments de la liste L (non nécessairement triée). Cet algorithme devra avoir une complexité en temps linéaire en moyenne (démonstration non demandée).

Exercice 8 :

Dessiner l'arbre binaire de recherche obtenu par insertion successive des éléments :

6, 14, 2, 5, 10, 7, 15, 12, 4, 9.

Proposer un autre ordre d'insertion qui aurait mené au même ABR.

Supprimer successivement les éléments 10 et 6 (dessiner l'ABR obtenu à chaque étape).

