Aix-Marseille-Université

Test 3 UE Algorithmie et programmation 2026

Contrôle L1

11-02-2026

Les réponses sont à donner directement sur le sujet. Un espace est réservé pour chaque réponse.

Équivalences

Difficulté : ★ ☆ ☆ Question :

Quelles sont les égalités correctes ?

① une fonction en $\mathcal{O}(n)$ est aussi en $\Theta(n)$

② une fonction en $\mathcal{O}(n)$ est aussi en $\mathcal{O}(n^2)$

③ une fonction en $\mathcal{O}(1)$ est aussi en $\Theta(1)$

④ une fonction en $\mathcal{O}(n^2)$ est aussi en $\mathcal{O}(n)$

Difficulté : ★ ☆ ☆ Question :

Quelle est la complexité $C(n)$ qui correspond à l'équation :

$$C(n) = \begin{cases} \mathcal{O}(1) + C(n-3) & \text{si } n > 3 \\ \mathcal{O}(1) & \text{sinon} \end{cases}$$

① $C(n) = \mathcal{O}(n) + \mathcal{O}(1)$

② $C(n) = \mathcal{O}(n^2)$

③ $C(n) = \Theta(n^2)$

④ $C(n) = \Omega(n^2)$

⑤ $C(n) = \Theta(n)$

⑥ $C(n) = \Omega(1)$

Difficulté : ★ ★ ☆ Question :

Quelle est la complexité de l'algorithme suivant :

algorithme remplace(T: [entier], i: entier, j: entier):
  $c \coloneqq T[i]$
  pour chaque ($k \coloneqq$ entier) de [0..T.longueur[**:**
    si $T[k] == c$ **:**
      pour chaque ($l \coloneqq$ entier) de [0..T.longueur[**:**
        $T[l] \leftarrow T[j]$

① $C(n) = \mathcal{O}(n^2)$ avec $n = T$.longueur

② $C(n) = \mathcal{O}(n)$ avec $n = T$.longueur

③ $C(n) = \mathcal{O}(1)$ avec $n = T$.longueur

Méthode de Newton (GEI 2023)

Difficulté : ★ ★ ☆ Question :

On considère deux nombres réels a et b et une fonction réelle f continue sur l'intervalle $[a, b]$ avec comme contrainte que les deux valeurs $f(a)$ et $f(b)$ soient de signes contraires. On cherche à l'aide de l'algorithme de dichotomie comment trouver le zéro de la fonction f sur l'intervalle $[a, b]$, c'est-à-dire trouver x tel que $f(x) = 0$. On rappelle brièvement cet algorithme qui consiste à subdiviser en deux parties un intervalle et choisir celui dans lequel existe un zéro de la fonction. On répète le processus jusqu'à ce que l'erreur absolue soit inférieure à une valeur ϵ définie initialement. L'erreur absolue de la méthode de dichotomie, après n étapes, est au plus égale à :

① $\log_2((b-a)/\epsilon)$

② $(b-a)/2^n$

③ $(b-a)/2^{n+1}$

④ $(b-a)/(n+1)$

⑤ $(b-a)/n$

Difficulté : ★ ★ ☆ Question :

Écrivez un algorithme qui implémente la recherche dichotomique. Sa signature devra être `dichotomie(f: (réel) -> réel, a: réel, b: réel, epsilon: réel) -> réel`.

Difficulté : ★ ★ ★ Question :

Donnez la complexité de l'algorithme précédent. Explicitez bien les paramètres que vous utilisez pour la mesurer.