

Objectifs: Être capable d'écrire des programmes récursifs simples

Exercice 1 : Recherche d'une lettre dans une chaîne

On cherche dans cet exercice à concevoir une fonction déterminant la présence d'un caractère dans une chaîne de caractères.

1. Ecrivez dans un premier temps l'algorithme principal qui saisira auprès de l'utilisateur une chaîne de caractères puis un caractère et produira un affichage similaire à : Le caractère 'o' est présent dans la chaîne "totoro"..
2. Définissez maintenant la fonction récursive correspondante qui teste l'existence d'une lettre donnée dans une chaîne de caractères donnée.
3. Dessinez l'arbre d'évaluation pour l'appel : `estPrésentDans('o', "totoro")`.
4. Est-ce que cet algorithme est récursif terminal ? Justifiez.

Exercice 2 : Calcul des termes d'une suite numérique

On souhaite calculer le terme de rang n de la suite numérique : $U_{n+1} = 3 * U_n + 1$ avec $U_0 = 1$.

1. Calculez manuellement le terme U_3 .
2. Définissez une fonction récursive calculant le n ième terme de la suite U .
3. Dessinez l'arbre d'évaluation pour l'appel `u(3)`.
4. Est-ce que cet algorithme est récursif terminal ? Justifiez.

Exercice 3 : Palindrome

Un palindrome est un mot (ou une phrase) qui, lu de droite à gauche, contient la même suite de lettres que lorsqu'il est lu de gauche à droite. Par exemple: "été", "radar", "elu par cette crapule", "éric, notre valet, alla te laver ton ciré".

Pour faciliter un peu votre tâche, vous pourrez supposer que vous disposez de la fonction suivante: `boolean isLetter(char c)` qui retourne `true` si `c` est une lettre et `false` sinon.

1. Définissez une fonction récursive vérifiant si une chaîne donnée est un palindrome.
2. Est-ce que cet algorithme est récursif terminal ? Justifiez.

Exercice 4 : Calcul de la puissance entière d'un nombre

On s'intéresse dans cet exercice au calcul récursif de la fonction puissance.

Dans un second temps, vous essayerez d'améliorer votre fonction en observant les différences entre les puissances paires et impaires.

1. Sachant que $a^n = a^{n-1} * a$ et que $a^{-n} = 1/a^n$, définissez la fonction récursive double `pow(double base, int exp)` calculant l'élévation à la puissance entière `exp` du nombre réel `base`.
2. Dessinez l'arbre d'évaluation pour l'appel : `pow(-2.0, 3)`.
3. Est-ce que cet algorithme est récursif terminal ? Justifiez.
4. Sachant que $3^6 = 729$, calculez la valeur de $3^2 * 3^2 * 3^2$. Que peut-on en déduire ?
5. En observant le résultat de la question précédente et en vous interrogeant sur la différence d'élévation à la puissance lorsque la puissance est paire ou impaire, vous semble-t-il possible d'améliorer l'efficacité de votre algorithme ?
6. Dessinez l'arbre d'évaluation pour l'appel : `pow(2.0, 8)`.
7. Comment pourriez-vous mesurer la différence d'efficacité entre vos deux algorithmes ?