

Programmation séquentielle

TP 6

05.10.2023

Le but de cet exercice est d'étudier numériquement des fonctions mathématiques.

Première partie

Le but est d'écrire une fonction capable de déterminer combien de fois une fonction change de signe dans un intervalle donnée. Soit la fonction $g(x) = 3x^3 + x^2 - 27$. Cette fonction change de signe une fois dans l'intervalle $[-1; 3]$.

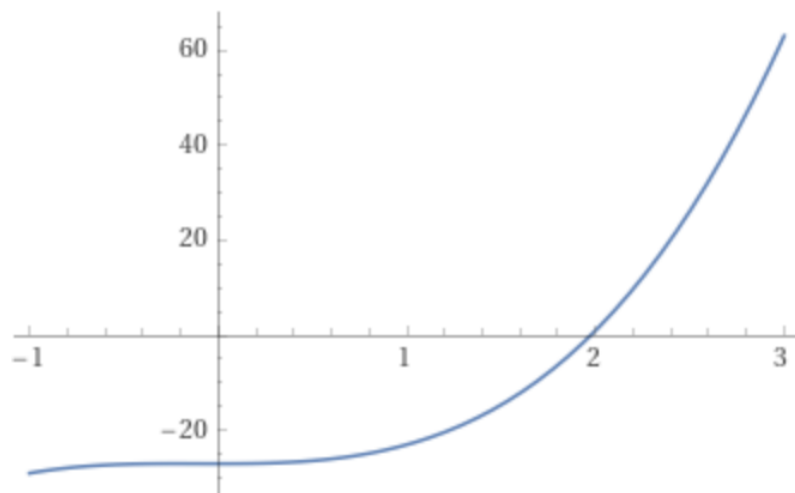


Figure 1: Fonction $g(x)$.

Pour cela, nous allons utiliser une technique naïve. Soit a la borne basse et b la borne haute, notre programme regardera sur chaque petit “tranche” de largeur ϵ entre ces deux bornes si la fonction a changé de signe.

Ainsi, le programme regardera d'abord si la fonction change de signe dans l'intervalle $[a; a + \epsilon]$, puis dans l'intervalle $[a + \epsilon; a + 2\epsilon]$, puis $[a + 2\epsilon; a + 3\epsilon]$, etc

jusqu'à ce que $a + n\epsilon \geq b$.

A chaque fois qu'un changement de signe est détecté (le signe de $a + n\epsilon$ est différent de celui de $a + (n + 1)\epsilon$), un compteur est incrémenté. Plus ϵ sera petit, moins les chances de “rater” un changement de signe seront grandes.

Pour cela, écrivez les fonctions :

```
double g(x);
```

qui retourne la valeur de la fonction g pour un x donné,

```
int sign(double x);
```

qui retourne 1 si le nombre passé en paramètre est positif et -1 si il est négatif (on considère que 0 est positif), et pour finir la fonction

```
int study_function(double epsilon, double low, double high);
```

qui retourne le nombre de fois que la fonction g change de signe dans l'intervalle allant de `low` à `high`, avec une précision de `epsilon`.

La dernière fonction doit obligatoirement utiliser les deux premières.

Deuxième partie

Changez la définition de $g(x)$ et assurez vous que `study_function` fonctionne toujours. Par exemple, définissez $g(x) = \sin(x)$.

Pour tracer des fonctions et vérifier que votre programme fonctionne correctement, vous pouvez utiliser <https://www.wolframalpha.com>.

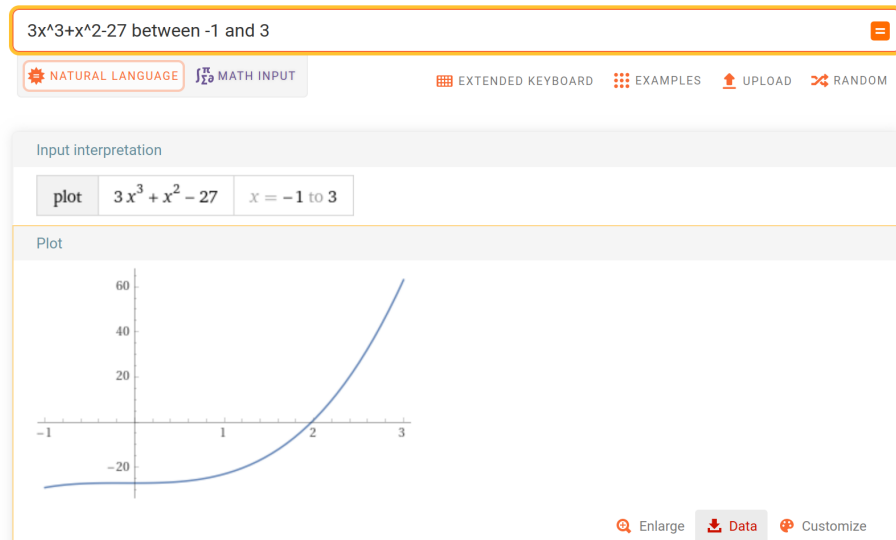


Figure 2: La fonction $g(x) = 3x^3 + x^2 - 27$ change de signe une fois entre -1 et 3.

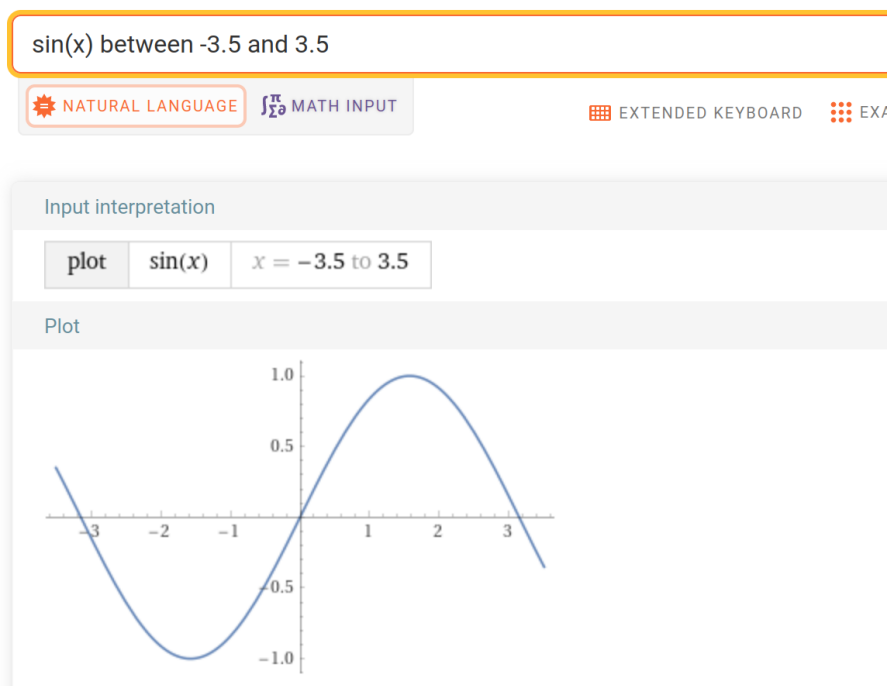


Figure 3: La fonction $g(x) = \sin(x)$ change de signe trois fois entre -3.5 et 3.5.

Troisième partie

Ne commencez cette partie que si la précédente est terminée.

Nous allons ajouter la possibilité de calculer numériquement l'intégrale de la fonction dans un intervalle donné. Pour rappel, une intégrale peut être définie informellement comme “l'aire sous une courbe entre deux points” [https://fr.wikipedia.org/wiki/Intégration_\(mathématiques\)](https://fr.wikipedia.org/wiki/Int%C3%A9gration_(math%C3%A9matiques))

Pour cela, on va calculer la somme des aires de chaque “petits rectangles” de largeur ϵ et de hauteur $g(x)$ pour $x \in a, a + \epsilon, a + 2\epsilon, a + 3\epsilon, \dots, b$ [https://fr.wikipedia.org/wiki/Intégration_\(mathématiques\)#Calcul_numérique_d'une_intégrale](https://fr.wikipedia.org/wiki/Int%C3%A9gration_(math%C3%A9matiques)#Calcul_num%C3%A9rique_d'une_int%C3%A9grale).

Normalement, vous devriez juste ajouter dans la boucle se trouvant dans `study_function` une somme qui calcule l'aire de chacun de ces petits rectangles.

Modifiez votre fonction pour que sa signature soit la suivante

```
int study_function(double epsilon, double low, double high, double *integral);
```

Le comportement de la fonction sera le même que précédemment, et en plus elle stockera dans `integral` la valeur estimée de l'intégrale dans l'intervalle donné.

Vous pouvez utiliser wolfram alpha pour connaître et visualiser le calcul d'une intégrale.

Essayez de changer la valeur d' ϵ (par exemple avec $\epsilon = 0.1$ et $\epsilon = 0.001$) et regardez comment cela impacte le calcul de l'approximation de l'intégrale.

integral of $3x^3+x^2-27$ between -1 and 3



NATURAL LANGUAGE



MATH INPUT



EXTENDED

Definite integral

$$\int_{-1}^3 (3x^3 + x^2 - 27) dx = -\frac{116}{3} \approx -38.667$$

Visual representation of the integral

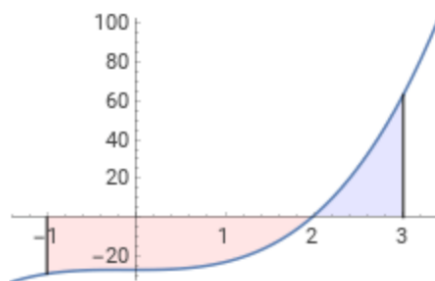


Figure 4: Exemple de calcul d'une intégrale.