

I Faire tourner un algorithme « à la main »

Variables n, i, I, S du type nombre

Lire n

$0 \rightarrow I$

$0 \rightarrow S$

Pour i allant de 1 à n

début

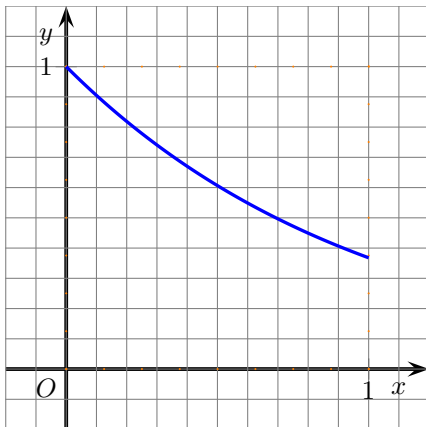
$I + \frac{1}{n} \times e^{-\frac{i}{n}} \rightarrow I$

$S + \frac{1}{n} \times e^{-\frac{i-1}{n}} \rightarrow S$

fin Pour

Afficher I

Afficher S



Ci-contre un algorithme écrit en langage libre.

Faire tourner cet algorithme « à la main » en remplissant le tableau suivant :

Choisir une valeur de n , par exemple $n = 5$.

| n | i | I | S |
|-----|-----|-----|-----|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Quel est le contenu des variables I et S à la fin de l'algorithme ?

II Coder un algorithme dans un langage

II.1 Avec « AlgoBox » puis « Python »

Coder l'algorithme du I. avec successivement AlgoBox et Python.

Pour Python : la fonction exponentielle s'écrit `exp()` après avoir insérer le module « math » de la manière suivante :

Saisir « `from math import*` »

II.2 Avec une boucle « Tant Que »

Reprendre l'algorithme précédent en utilisant une boucle « Tant Que ».

Vous l'écrirez tout d'abord en langage libre puis le coderez avec le l'outil de votre choix.

II.3 Exercice

Dans le I. , l'algorithme calcule pour un nombre n fixé par l'utilisateur (nombre de rectangles), les quantités S et I respectivement les sommes des rectangles « supérieurs » et des rectangles « inférieurs ».

La différence $S - I$ représente l'erreur maximale que l'on commet en utilisant ce procédé pour calculer l'aire exacte sous la courbe. Modifier l'algorithme précédent de la manière suivante :

- Ne plus demander le nombre d'itérations dans l'algorithme ;
- L'utilisateur saisit un entier p représentant l'erreur maximale qu'il autorise, en faisant varier le nombre n , calculer S et I jusqu'à ce que l'inégalité $S - I < 10^{-p}$ soit vérifiée (condition d'arrêt de la boucle)

Exemple 1 Pour $p = 2$, les calculs s'effectuent jusqu'à ce que $S - I < 10^{-2} \Leftrightarrow S - I < 0.01$

- Lorsque la boucle s'arrête, que représente la valeur de n et les quantités S et I .
- Structure possible de l'algorithme :

Variables n, i, I, S, p du type nombre

Lire p (valeur entière supérieure ou égale à 0)

$1 \rightarrow n$

$0 \rightarrow I$

$1 \rightarrow S$

Tant que $S - I \geq 10^{-p}$ faire

 début

$0 \rightarrow S$

$0 \rightarrow I$

 boucle du I. (boucle *pour*)

$n + 1 \rightarrow n$

 fin Tant que

Afficher $n - 1$

Afficher S

Afficher I