TP 10

Intégration numérique

Sources :

Savoirs et compétences :

- *AN.S3 : Méthodes des rectangles et des trapèzes pour le calcul approché d'une intégrale sur un segment.*
- *SN.C1 : Réaliser un programme complet structuré*
- *SN.C2 : Étudier l'effet d'une variation des paramètres sur le temps de calcul, sur la précision des résultats, sur la forme des solutions pour des programmes d'ingénierie numérique choisis, tout en contextualisant l'observation du temps de calcul par rapport à la complexité algorithmique de ces programmes*
- *SN.C3 : Utiliser les bibliothèques de calcul standard*
- *SN.C5 : Tenir compte des aspects pratiques comme l'impact des erreurs d'arrondi sur les résultats, le temps de calcul ou le stockage en mémoire.*
- *SN.S1 : Bibliothèques logicielles*

Activité 1 : Fractions rationnelles

Le but est d'obtenir un encadrement de $I = 4\int_0^1 \frac{\mathrm{d}x}{1+x^2}$

Q 1 : Compléter cet algorithme et le coder en python afin d'obtenir une valeur approchée de I par la méthode des rectangles à gauche en utilisant les champs suivants : $\boxed{\text{x+h}}$ $\boxed{\text{(b-a)/n}}$ $\boxed{\text{h}}$ $\boxed{\text{somme+f(x)}}$ $\boxed{\text{f(a)}}$.

```
a=0
b=1
n=100
h=
x=a
somme=
for k in range(1,n) :
    x=
    somme=
print(somme*
```

Q 2 : Modifier cet algorithme pour que la méthode soit celle des rectangles à droite.

Q 3 : Modifier cet algorithme afin d'afficher les résultats des deux méthodes.

Q 4 : Augmenter le nombre de subdivisions.

Q 5 : Justifier que la méthode des rectangles à droite donne un minorant de I et que la méthode des rectangles à gauche donne un majorant.

Pour tout $n \in \mathbb{N}$ soit l'intégrale :

$$I_n = \int_0^1 \frac{x}{1+x^n}\mathrm{d}x$$

Q 6 : Écrire une fonction d'argument n qui renvoie une valeur approchée de I_n par une des méthodes des rectangles (avec une subdivision de 100 intervalles).

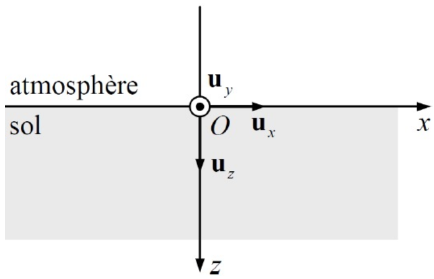
Q 7 : Afficher quelques valeurs de cette suite afin de conjecturer sa monotonie et son comportement asymptotique.

Activité 2 : Mise hors gel des canalisations d'eau

La température dans le sol terrestre étant initialement constante, égale à 5°C , on cherche à déterminer à quelle profondeur minimale il est nécessaire d'enterrer une canalisation d'eau pour qu'une brusque chute de la température de sa surface à -15°C n'entraine pas le gel de cette canalisation après 10 jours.

Les hypothèses sont les suivantes :

- la température en un point quelconque du sol et de sa surface à tout instant $t < 0$ est constante et égale à $T_0 = 278\ K$ ($\theta_0 = 5^o C$) ;
- la température à la surface du sol, confondue avec le plan d'équation $z = 0$, passe brutalement à l'instant $t = 0$, de $T_0 = 278\ K$ à $T_1 = 258\ K$ ($\theta_1 = -15^o C$) et se maintient à cette valeur pendant t_f =10 jours.



On peut montrer que la température $T(z,t)$ à la profondeur z et à l'instant t est donnée par la relation suivante :

$$T(z,t) = T_1 + (T_0 - T_1)\mathrm{erf}\left(\frac{z}{2\sqrt{D\,t}}\right)$$

où erf(x) désigne la fonction définie par :

$$\mathrm{erf}(x) = \frac{2}{\pi}\int_0^x e^{-u^2}\mathrm{d}u$$

Données numériques : $D = 2,8 \cdot 10^{-7}\ m^2 \cdot s^{-1}$ (diffusivité thermique du sol terrestre).

Q 8 : Écrire une fonction Python, appelée erf, prenant en paramètre un nombre réel positif ou nul x et retournant la valeur de erf(x).

Q 9 : Écrire une fonction Python, appelée Temperature, prenant en paramètre la profondeur z (exprimée en m) et le temps t (exprimé en s) et retournant la valeur de la température $T(z,t)$.

Q 10 : Écrire un programme Python permettant de créer une liste, nommée ListeErreur, contenant les valeurs de la fonction erf(x) pour x variant par pas de 0,05 dans l'intervalle $[0;2]$.

Q 11 : En déduire, à 1 cm près, à quelle profondeur minimale z_{min} il est nécessaire d'enterrer une canalisation d'eau pour qu'une brusque chute de la température de la surface du sol de 5°C à -15 °C n'entraine pas le gel de cette canalisation au bout de 10 jours.