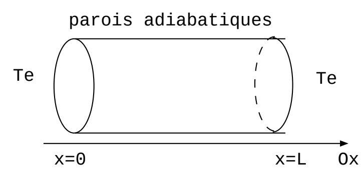
# TP modélisation : diffusion thermique

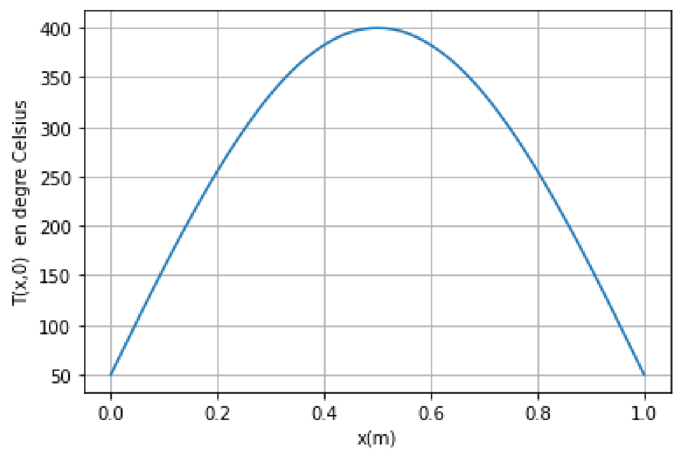
On étudie la diffusion thermique dans un cylindre de longueur $L$ et d'axe $Ox$. Les extrémités du cylindre sont maintenues à température constante $T(x=0,t) = T(x=L,t) = T_e$ (température extérieure). Les parois latérales sont adiabatiques.




La température initiale dans le cylindre n'est pas homogène, elle s'écrit $T(x,0) = T_e + (T_m - T_e)\sin(\frac{\pi x}{L})$. Le cylindre est donc le siège d'un phénomène de conduction thermique. On cherche à étudier $T(x,t)$ , la température dans le cylindre en fonction du temps.

**1.** On donne le code et son exécution:

```
import numpy as np
import matplotlib.pyplot as plt
Nx,L=100,....
def T0(Te,Tm,x):
——return Te+(Tm-Te)*np.sin(np.pi*x/L)
x=np.linspace(0,L,Nx+1)
plt.plot(x,T0(.....,....,.....))
plt.xlabel('x(m)')
plt.ylabel('T(x,0) en degre Celsius')
plt.grid()
plt.show()
```




Compléter le code à partir de la courbe obtenue. Prévoir qualitativement l'évolution de la température dans le cylindre au cours du temps.

**2.** On rappelle l'équation de diffusion thermique $\frac{\partial T}{\partial t} = D\Delta T$ où $D = \frac{\lambda}{\rho c}$ désigne le coefficient de diffusion. Simplifier cette équation dans le cas où $T = T(x,t)$. On note $\tau$ le temps caractéristique de diffusion dans le cylindre. Exprimer $\tau$ en fonction de $D$ et $L$.

Pour résoudre l'équation de diffusion, on utilise la méthode d'Euler et on introduit: $dt$ le pas de temps et $dx$ le pas d'espace.

Le temps est discrétisé et la variable $t$ se met sous la forme $t_j = j.dt$ avec $j = 0, 1, ...N_t$.

L'espace est discrétisé et la variable $x$ se met sous la forme $x_i = i.dx$ avec $i = 0, 1, ...N_x$.

La température s'écrit $T(x_i, t_j) = T(i.dx, j.dt)$. On définit un tableau 2D noté $tabT$, dont chaque terme $tabT[i,j]$ (terme sur la ligne $i$ et la colonne $j$) représente la température à l'abscisse $x_i = i.dx$ à l'instant $t_j = j.dt$.

On donne: $f(x+\epsilon) = f(x) + \epsilon f'(x) + \frac{\epsilon^2}{2} f''(x)$ pour $\epsilon$ voisin de 0.

Remarque: dans les sujets de concours, le DL est donné parfois sous cette forme: $f(y) = f(a) + (y-a)f'(a) + \frac{(y-a)^2}{2} f''(a)$ pour $y$ voisin de $a$.

**3.** Ecrire un DL à l'ordre 2 en $dx$ de $T(x+dx,t)$ et de $T(x-dx,t)$. En déduire $\frac{\partial^2 T}{dx^2}(x,t)$ en fonction de $T(x+dx,t)$, $T(x,t)$, $T(x-dx,t)$ et $dx$.

Ecrire un DL à l'ordre 1 de $T(x,t+dt)$ et en déduire $\frac{\partial T}{\partial t}(x,t)$ en fonction de $T(x,t)$, $T(x,t+dt)$ et $dt$.

Déduire de ces expressions et de l'équation de diffusion que l'on a $T(x,t+dt) = T(x,t) + r(T(x+dx,t) + T(x-dx,t) - 2T(x,t))$. Exprimer $r$ en fonction de $D$, $dt$ et $dx$.

**4.** On remplace $T(x_i,t_j)$ par $tabT[i,j]$. Par quoi remplace-t-on $T(x+dx,t)$? $T(x,t+dt)$? $T(x-dx,t)$? En déduire la relation de récurrence donnant $tabT[i,j+1]$ en fonction de termes de la forme $tabT[i,...]$.

**5.** Le tableau tabT est de la forme suivante:

$$\begin{array}{c|cccccc|c}
 & \texttt{j=0} & \texttt{j=1} & \texttt{j=2} & \texttt{j} & \texttt{j+1} & \texttt{j=Nt} & \\
\hline
\texttt{i=0} & & & & & & & \texttt{x=} \\
\texttt{i=1} & & & & & & & \texttt{x=} \\
\texttt{i=2} & & & & & & & \texttt{x=} \\
\texttt{i-1} & & & & & & & \texttt{x=} \\
\texttt{i} & & & & & & & \texttt{x=} \\
\texttt{i+1} & & & & & & & \texttt{x=} \\
\texttt{i=Nx} & & & & & & & \texttt{x=} \\
\hline
 & \texttt{t=} & \texttt{t=} & \texttt{t=} & \texttt{t=} & \texttt{t=} & \texttt{t=} &
\end{array}$$

Rappeler les conditions aux limites et préciser la position (ligne, colonne?) dans le tableau tabT des termes associés.

Rappeler les conditions initiales et préciser la position (ligne, colonne?) dans le tableau tabT des termes associés.

Repérer sur le tableau le terme $tabT[i,j+1]$ et repérer les termes nécessaires pour son calcul par récurrence. Conclure.

**6.** Compléter et analyser le code suivant:

```
12 rho,c,lamba,Te,Tm,Nx=2150,1000,1.65,50,400,100
13 D=................
14 tau=.....
15 dx=L/Nx
16 Nt=100000
17 dt=tau/Nt
18 r=................................... # on doit vérifier que r < 1/2 : c'est le critère de convergence
19 tabT=np.zeros((Nx+1,Nt+1)) # tabT est un tableau de Nx+1 lignes et Nt+1 colonnes ne comportant que des zéros
20 tabT[:,0]=...................................# désigne la première colonne du tableau
21 tabT[0,:]=......................# désigne la première ligne du tableau
22 tabT[Nx,:]=................# désigne la dernière ligne du tableau
23 for i in range(1,Nx):
24 ——tabT[i,1]=tabT[i,0]+r*(tabT[i-1,0]-2*tabT[i,0]+tabT[i+1,0])
```

Compléter les lignes 13, 14 et 18 avec les valeurs littérales.

Faire les applications numériques de $D$, $\tau$, $dt$, $dx$ et $r$. Le critère de convergence est-il vérifié?

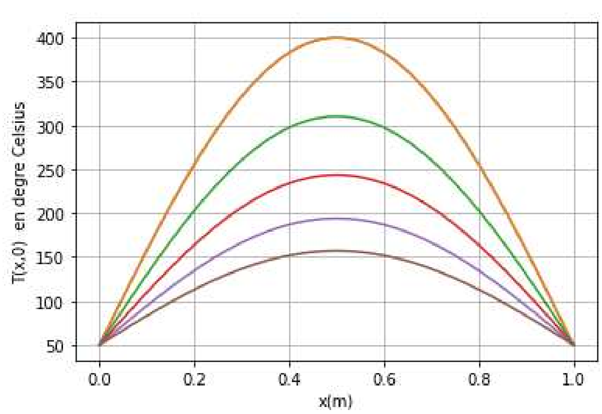
A quels instants correspondent $j = 1000$ et $j = 5000$? A quelles positions correspondent $i = 15$ et $i = 60$?

Compléter les lignes 20, 21 et 22 avec les conditions initiales et les conditions aux limites.

Expliquer ce que réalisent les lignes 23 et 24.

**7.** On finit de compléter le tableau tabT avec les lignes de code suivantes:

```
25 for j in range(1,Nt):
26 ——for i in range(1,Nx):
27 ————tabT[i,j+1]=........................
28 plt.plot(x,tabT[:,0])
29 for j in range(5):
30 ——plt.plot(x,tabT[:,j*3000])
31 plt.xlabel('x(m)')
32 plt.ylabel('T(x,0) en degre Celsius')
33 plt.grid()
34 plt.show()
```



Compléter la ligne 27.

Donner les valeurs numériques des instants pour lesquelles on a tracé les 5 courbes de températures dans le cylindre.

Commenter l'évolution de la température dans le cylindre au cours du temps.

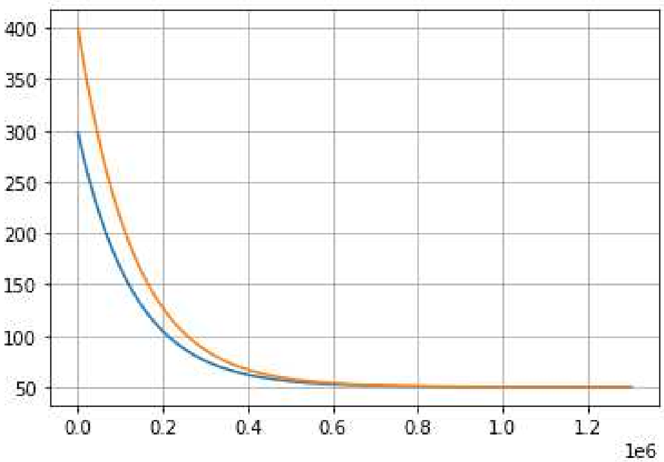
**8.** On ajoute les courbes suivantes, déduire du code ce qu'elles représentent:

35 plt.plot(np.linspace(0,tau,Nt+1),tabT[25,:])

36 plt.plot(np.linspace(0,tau,Nt+1),tabT[50,:])

37 plt.grid()

38 plt.show()



Document réponse:

```python
import numpy as np
import matplotlib.pyplot as plt
Nx,L=100,....
def T0(Te,Tm,x):
    return Te+(Tm-Te)*np.sin(np.pi*x/L)
x=np.linspace(0,L,Nx+1)
plt.plot(x,T0(....,....,....))
plt.xlabel('x(m)')
plt.ylabel('T(x,0) en degre Celsius')
plt.grid()
plt.show()
rho,c,lamba,Te,Tm,Nx=2150,1000,1.65,50,400,100
D=........
tau.............
dx=L/Nx
Nt=100000
dt=tau/Nt
r=..........
tabT=np.zeros((Nx+1,Nt+1))
tabT[:,0]=........
tabT[0,:]=.......
tabT[Nx,:]=......
for i in range(1,Nx):
    tabT[i,1]=tabT[i,0]+r*(tabT[i-1,0]-2*tabT[i,0]+tabT[i+1,0])
for j in range(1,Nt):
        for i in range(1,Nx):
                tabT[i,j+1]=...........
plt.plot(x,tabT[:,0])
for j in range(5):
    plt.plot(x,tabT[:,j*3000])
    plt.xlabel('x(m)')
    plt.ylabel('T(x,0)  en degre Celsius')
    plt.grid()
plt.show()
plt.plot(np.linspace(0,tau,Nt+1),tabT[25,:])
plt.plot(np.linspace(0,tau,Nt+1),tabT[50,:])
plt.grid()
plt.show()
```