1. ATS 2016

51) *En résumé, la méthode des rectangles consiste à faire l'approximation :*

$$\int f(x)\,dx \approx \sum_i f(x_i)\,\Delta x$$

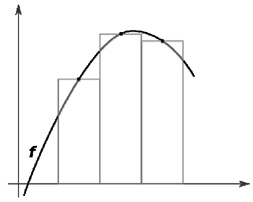


illustration succincte

D'après la procédure indiquée, l'intégrale S se calcule en ajoutant des termes de la forme $f\left(\frac{x_1+x_2}{2}\right)\times(x_2-x_1)$ *; f(x) est donc bien la fonction à intégrer,*

l'intervalle $[x_1,x_2]$ *variant entre* $\left[a,a+\frac{b-a}{n}\right]$*, pour* $i=1$*, et* $\left[a+(n-1)\frac{(b-a)}{n},b\right]$*, pour* $i=n-1$*.*

Sachant que $\Delta V = U_{AT} = \int_A^T \overline{E}.dr = \int_{R+z_0}^{R} -\frac{Q}{4\pi\varepsilon_0 r^2}.dr = \int_{r=R}^{r=R+z_0} \frac{Q}{4\pi\varepsilon_0 r^2}.dr$

on obtient par identification $\Delta V = \int_{x=a}^{x=b} f(x)\,dx$ *avec* $\boxed{a=R\ ;\ b=R+z_0\ ;\ f(x)=\frac{Q}{4\pi\varepsilon_0 x^2}}$

2. ATS 2017

8. *La dérivée partielle* $\frac{\partial V(y,z)}{\partial z}$ *est associée à la quantité* $\frac{V_{i,j+1}-V_{i,j}}{\Delta z}$ *:* $\boxed{\frac{\partial V(y,z)}{\partial z} \approx \frac{V_{i,j+1}-V_{i,j}}{\Delta z}}$

Remarque : $\Delta y = \Delta z = 1\,\mathrm{mm}$ *représente le pas de discrétisation de la méthode d'Euler, c'est ici un pas spatial, dans les deux directions y et z.*

9. *La dérivée seconde correspondant à la variation de la dérivée première* $\frac{\partial^2 V(y,z)}{\partial z^2} \approx \frac{\frac{V_{i,j+1}-V_{i,j}}{\Delta z} - \frac{V_{i,j}-V_{i,j-1}}{\Delta z}}{\Delta z}$

$\Rightarrow \quad \frac{\partial^2 V(y,z)}{\partial z^2} \approx \frac{V_{i,j+1}-V_{i,j}-V_{i,j}+V_{i,j-1}}{\Delta z^2}$, *soit* $\boxed{\frac{\partial^2 V(y,z)}{\partial z^2} \approx \frac{V_{i,j+1}-2V_{i,j}+V_{i,j-1}}{\Delta z^2}}$

10. $V_{i,j}$ *représente* $V(y,z)$*, qui satisfait l'équation de Laplace* $\frac{\partial^2 V(y,z)}{\partial y^2}+\frac{\partial^2 V(y,z)}{\partial z^2}=0$

Par analogie avec le résultat précédent, on peut écrire que $\frac{\partial^2 V(y,z)}{\partial y^2} \approx \frac{V_{i+1,j}-2V_{i,j}+V_{i-1,j}}{\Delta y^2}$

L'équation de Laplace correspond donc à $\frac{V_{i+1,j}-2V_{i,j}+V_{i-1,j}}{\Delta y^2}+\frac{V_{i,j+1}-2V_{i,j}+V_{i,j-1}}{\Delta z^2}=0$*, avec, rappelons-le,* $\Delta y = \Delta z$

On a donc $V_{i+1,j}-2V_{i,j}+V_{i-1,j}+V_{i,j+1}-2V_{i,j}+V_{i,j-1}=0$ *d'où* $\boxed{V_{i,j}=\frac{V_{i+1,j}+V_{i-1,j}+V_{i,j+1}+V_{i,j-1}}{4}}$

3. ATS 2018

48) *Il s'agit de transposer sans trop se poser de questions le script fourni.*

```
function y=f(Z)
    y=Z/(1+Z^2)^4
endfunction
Z=linspace(0,2,100)
plot(Z,f)
xlabel("Z"); ylabel("Z/(1+Z^2)^4")
```

4. ATS 2019

14) *Pour obtenir le tracé de* $\mathrm{U_p(t)}$*, il suffit de modifier une ligne dans la boucle définissant la deuxième fonction :*

on remplace

```
L($+1)=i(t)
```

par

```
L($+1)=R*i(t)
```

Pour modifier le nom du graphique ainsi que l'étiquette des ordonnées, on remplace, dans la section "tracé"

```
ylabel("i(A)")
title("i(t)")
```

par

```
ylabel("Up(V)")
title("Up(t)")
```

Remarque : on constate qu'il n'est pas nécessaire de comprendre le détail de ce programme pour répondre à la question.

Remarque 2 : si on se contente d'ajouter des lignes, comme suggéré par l'énoncé d'origine, on superpose les deux graphiques.

5. ATS 2020

21)

```
function tab_Ttheo = Temp(t,k)
    tab_Ttheo = 30*(1-exp(-k.*t));
endfunction
```

22)

```
function e = erreur(k)
    e = sum((tab_T - Temp(tab_temps,k))^2);
endfunction
```

23)

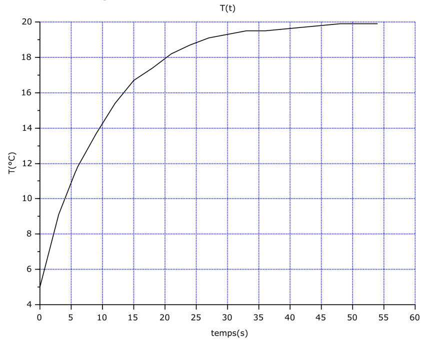
```
k = linspace(0.0001,0.01,100)'
tab_e = [ ]
for i=1:size(k,1)
    tab_e(i) = erreur(k(i));
end
```

6. ATS 2021

19) *Voici un exemple de code Scilab (avec options)*

```
M=csvRead("mesures.csv");
// construction des tableaux demandés
t = M(:,1);
T = M(:,2);
// affichage des tableaux demandés (facultatif)
disp("tableau des temps t d''acquisition",t)
disp("tableau des températures T",T)
// tracé
plot2d(t,T)
// grille (optionnel), titre et étiquette des axes
xgrid(2)
title("T(t)")
xlabel("temps(s)")
ylabel("T(°C)")
```

graphe obtenu (pour info) :



7. ATS 2022

33. *On a* $v = \dfrac{dz}{dt} \Rightarrow dz = v\ dt$. *Donc* $z(t+dt) = z(t) + dz = z(t) + v\ dt$.

C'est le principe de la méthode d'Euler, avec un pas noté `dt`*, pour la fonction* `z(t)` *liée à sa dérivée* `v`*.*

Transposé en langage informatique dans une boucle, on obtient :

```
z(i+1) = z(i) + v(i)*dt
```

34. *D'une manière analogue, on va lier la fonction* v (t) *à sa dérivée* $\dot{v}$ *:* $v(t+dt) = v(t) + \dot{v}\ dt$ *, avec* $\dot{v} = g - \dfrac{\alpha}{m} v^2$

On obtient donc :

```
v(i+1) = v(i) + (g-alpha*v(i)^2/m)*dt
```

8. ATS 2023

42 -

```
maxi = 0
imax = 0

for i=1:150
    if eta(i) >= maxi then
       maxi = eta(i) // permet de conserver la valeur maximale du rendement
       imax = i      // permet de conserver l'indice correspondant au rendement maximum
    end
end
```

43 -

```
disp(imax) // affichage dans la console
disp(maxi) // affichage dans la console
disp(p3(imax)) // affichage de la valeur de p3 correspondant au rendement maximum
disp(x4(imax)) // affichage de la valeur de x4 correspondant au rendement maximum
```