Corrigés des exercices

Tableau des capacités exigibles par exercice

Capacité exigible	5 minutes chrono et QCM	Exercices résolus	Exercices rapides	Appliquer	S'entraîner	Objectif Première
Exploiter la loi des mailles et la loi des nœuds dans un circuit électrique comportant au plus deux mailles.	1, 2, 3, 4	41	12, 13, 14, 34	17, 18, 19, 21, 32, 39, 42	45, 50, 51	52, 53, 54
Mesurer une tension et une intensité.				16, 32	47, 49	
Exploiter la caractéristique d'un dipôle électrique : point de fonctionnement, modélisation par la relation $U = f(I)$ ou $I = g(U)$.		10, 41		27, 28, 29, 32, 40, 42	47	53
Utiliser la loi d'Ohm.	5, 6, 7, 8, 9	11, 41	22, 23, 24, 25	26, 29, 30, 32, 33, 37, 39, 40, 42	46, 47, 48, 49, 50, 51	52, 53, 54
Représenter et exploiter la caractéristique d'un dipôle.			34	31, 32, 33	47, 49	
√NUMÉRIQUE Représenter un nuage de points associé à la caractéristique d'un dipôle et modéliser la caractéristique de ce dipôle à l'aide d'un langage de programmation.				33	49	
√MATHS Identifier une situation de proportionnalité.				29, 33	43, 49	52, 53, 54
Citer des exemples de capteurs présents dans les objets de la vie quotidienne.	8, 9	41	35	36, 42, 43	43, 44	52, 53, 54
Mesurer une grandeur physique à l'aide d'un capteur électrique résistif. Produire et utiliser une courbe d'étalonnage utilisant la résistance d'un système avec une grandeur d'intérêt (température, pression, intensité lumineuse, etc.).	8	41	34	37, 42, 43	43, 50	
Utiliser un dispositif avec microcontrôleur et capteur.		41	34	42	50, 51	53, 54
√MESURE ET INCERTITUDES Exploiter une série de mesures, discuter de l'influence du protocole et/ou évaluer une incertitude-type pour comparer des résultats.				38		

Exercices 1 à 9

Corrigés dans le manuel.

10 Exploiter la caractéristique d'un dipôle

APPLICATION

Graphiquement si $I = 5{,}0$ mA, $U = 2{,}0$ V.

11 Utiliser la loi d'Ohm APPLICATION

D'après la loi d'Ohm, $U = R \times I$.

Ainsi : $R = \dfrac{U}{I}$.

$R = \dfrac{1{,}0 \text{ V}}{0{,}40 \text{ mA}}$ $R = \dfrac{1{,}0 \text{ V}}{4{,}0 \times 10^{-4} \text{ A}} = 2{,}5 \times 10^3\,\Omega = 2{,}5 \text{ k}\Omega$.

12 ORAL

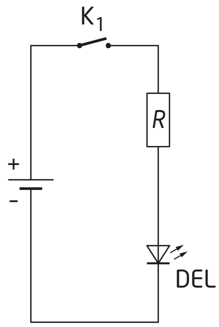
Vérifier que les notions de nœuds et de mailles sont clairement définies sur un ou deux schémas, que des flèches courants et tensions sont représentées sur ces schémas et qu'à partir de là, les lois des nœuds et des mailles sont clairement écrites sous forme de relation littérales et également sous forme d'énoncés.

13 La loi des nœuds donne ici : $I_1 = I_2 + I_3$ soit $I_3 = I_1 - I_2$.

A.N. : $I_3 = 3{,}5 \text{ mA} - 2{,}0 \text{ mA} = 1{,}5 \text{ mA}$.

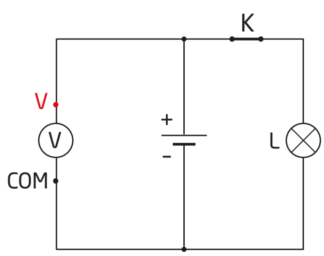
14 La loi des mailles donne ici : $9 \text{ V} - 0 \text{ V} - U - 6 \text{ V} = 0$ soit $U = 9 \text{ V} - 6 \text{ V} = 3 \text{ V}$.

15 Représenter un schéma électrique

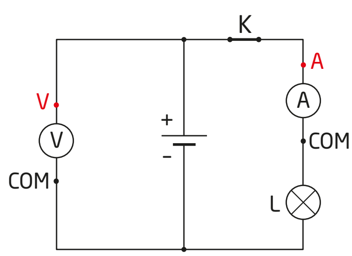


16 Schématiser un circuit électrique

a.



b. Il suffit d'intégrer un ampèremètre dans la branche qui comporte la lampe, comme le montre le schéma ci-dessous.



17 Exploiter la loi des nœuds

Corrigé dans le manuel.

18 Exploiter la loi des mailles

Corrigé dans le manuel.

19 Déterminer la valeur d'une tension inconnue

La loi des mailles donne ici : $U_{AB} + U_{DA} + U_{CD} + U_{BC} = 0$.

a. $U_{DA} = -U_{AB} - U_{CD} - U_{BC}$.

A.N. : $U_{DA} = -2{,}0\ \text{V} - (-4{,}0)\ \text{V} - 1{,}2\ \text{V} = 0{,}8\ \text{V}$.

b. $U_{CD} = -U_{AB} - U_{DA} - U_{BC} = -U_{AB} + U_{AD} - U_{BC}$.

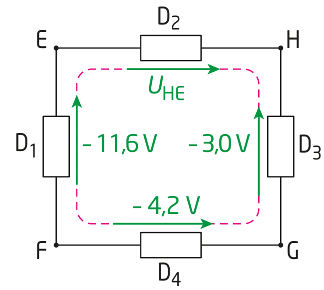
A.N. : $U_{CD} = -4{,}5\ \text{V} + 2{,}5\ \text{V} - 3{,}2\ \text{V} = -5{,}2\ \text{V}$.

c. $U_{DA} = -U_{AB} - U_{CD} - U_{BC}$.

A.N. : $U_{CD} = -1{,}5\ \text{V} - 513\ \text{mV} - 0{,}75\ \text{V} = -1{,}5\ \text{V} - 0{,}513\ \text{V} - 0{,}75\ \text{V} = -2{,}7\ \text{V}$.

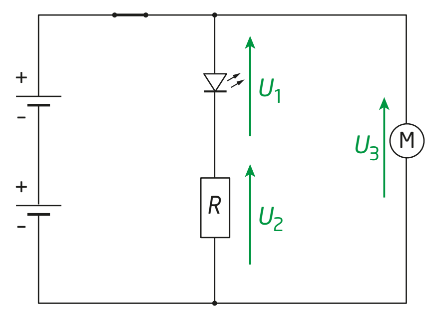
En général, dans une somme, le nombre de chiffres significatifs du résultat est celui de la donnée la moins précise de l'énoncé.

20 Schématiser un montage électrique

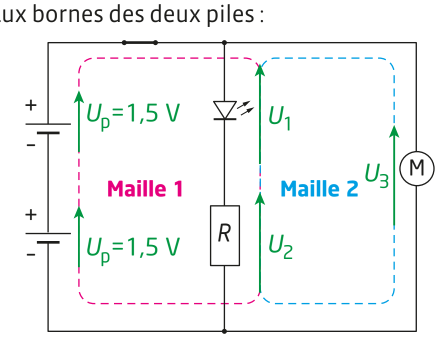


21 Appliquer la loi des mailles

a.



b. Pour pouvoir calculer la tension U_2, il faut représenter les tensions aux bornes des deux piles :



Dans la maille 1, on a : $Up + Up - U_1 - U_2 = 0$ soit $U_2 = + Up + Up - U_1$.
A.N. : $U_2 = +1{,}5\ \text{V} + 1{,}5\ \text{V} - 1{,}2\ \text{V} = +1{,}8\ \text{V}$.
Dans la maille 2, on a : $U_1 + U_2 - U_3 = 0$ soit $U_3 = U_1 + U_2$.
A.N. : $U_3 = +1{,}2\ \text{V} + 1{,}8\ \text{V} = +3{,}0\ \text{V}$.

22 ORAL

Vérifier que la présentation propose une représentation graphique de type droite linéaire avec des axes comportant la tension U aux bornes du conducteur ohmique en ordonnées et I l'intensité du courant électrique le traversant.

23

On applique la loi d'Ohm : $U = R \times I$.

A.N. : $U = 46\ \Omega \times 1{,}0 \times 10^{-3}\ \text{A} = 46 \times 10^{-3}\ \text{V} = 0{,}046\ \text{V}$.

24

On applique la loi d'Ohm : $U = R \times I$ soit $I = \dfrac{U}{R}$.

A.N. : $I = \dfrac{3{,}3\ \text{V}}{3{,}3 \times 10^{3}\ \Omega} = 1{,}0 \times 10^{-3}\ \text{A} = 1{,}0\ \text{mA}$.

25

On applique la loi d'Ohm : $U = R \times I$ soit $R = \dfrac{U}{I}$.

A.N. : $R = \dfrac{10\ \text{V}}{0{,}050\ \text{A}} = \dfrac{10\ \text{V}}{5{,}0 \times 10^{-2}\ \text{A}} = 2{,}0 \times 10^{2}\ \Omega$.

26 Utiliser la loi d'Ohm

On applique la loi d'Ohm : $U_{AB} = R \times I$, soit dans le cas de la figure : $U_{BA} = -R \times I$.

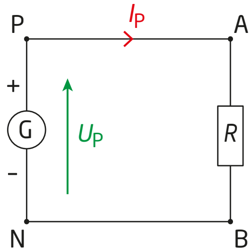
A.N. : $U_{BA} = -R \times I = -330\ \Omega \times 1{,}0 \times 10^{-3}\ \text{A} = -3{,}3 \times 10^{-1}\ \text{V} = -0{,}33\ \text{V}$.

27 Déterminer un point de fonctionnement

Corrigé dans le manuel.

28 Utiliser un générateur de courant

Le schéma du circuit est le suivant :



L'intensité commune du courant traversant le générateur et la résistance R vaut : $I_p = 1{,}0 \times 10^{-3}$ A.

On applique la loi d'Ohm : $U_{AB} = R \times I_p = U_p$.

A.N. : $U_p = R \times I = 1000\ \Omega \times 1{,}0 \times 10^{-3}\ \text{A} = 1{,}0\ \text{V}$.

Les coordonnées du point de fonctionnement sont donc : $I_p = 1{,}0 \times 10^{-3}$ A et $U_p = 1{,}0$ V.

29 Identifier une situation de proportionnalité

Corrigé dans le manuel.

30 Limiter les risques électriques

S'AUTOÉVALUER

Corrigé dans le manuel.

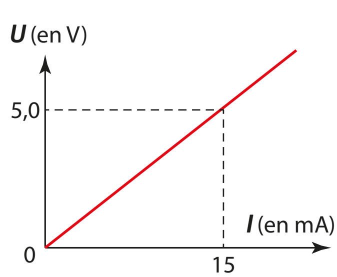
31 Représenter la caractéristique d'une résistance

La représentation graphique de la caractéristique d'une résistance suit la loi d'Ohm et il s'agit donc d'une droite linéaire (droite qui passe par l'origine du repère). Il faut au minimum deux points pour représenter une droite. Le premier point est l'origine O du repère (O, I, U) de coordonnées (0 A, 0 V).

Le deuxième point est donné par la loi d'Ohm et il correspond au point le plus éloigné du point O. Ses coordonnées sont :

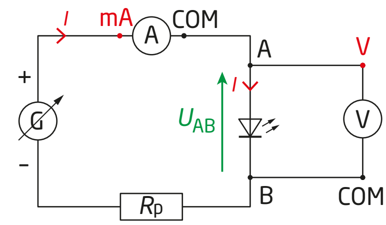
$U = 5{,}0$ V et $I = \dfrac{U}{R} = \dfrac{5{,}0\ \text{V}}{330\ \Omega} = 0{,}015\ \text{A} = 15\ \text{mA}$.

On en déduit la représentation graphique de la caractéristique de la résistance ci-dessous.

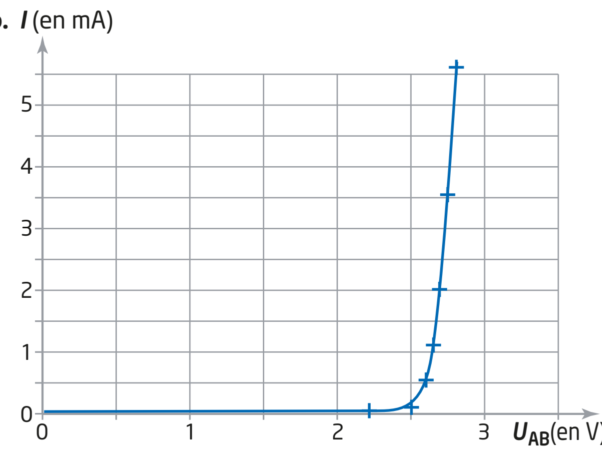


32 Étudier la caractéristique d'une DEL

a.



b.



c. La loi des mailles appliquée à l'unique maille du circuit donne : $U_g - U_{AB} - U_{Rp} = 0$.

La loi d'Ohm aux bornes de la résistance R_p s'écrit : $U_{Rp} = R_p \times I_{max}$.

On a donc : $U_g - U_{AB} - R_p \times I_{max} = 0$ soit $I_{max} = \dfrac{(U_g - U_{AB})}{R_p}$

La valeur de l'intensité du courant sera maximale lorsque la tension aux bornes du générateur sera également maximale.

A.N. : $I_{max} = \dfrac{(12 - 2{,}8)\ \text{V}}{470\ \Omega} = 9{,}1 \times 10^{-3}\ \text{A} = 9{,}1\ \text{mA} < 20\ \text{mA}$, la résistance de protection limite l'intensité du courant à une valeur de l'ordre de 9 mA lorsque la tension aux bornes du générateur atteint 12 V.

*Le choix de la valeur 2,8 V pour la tension U_{AB} se justifie par le fait que, d'après le graphe obtenu à la question **b.**, la valeur de la tension aux bornes de la DEL ne peut pas être inférieure à 2,8 V pour un courant supérieur à 5,63 mA.*

33 Modéliser une caractéristique

DIFFÉRENCIATION

Corrigé dans le manuel et en Vidéo *sur* **sirius.nathan.fr.**

➜ **Fichiers numériques disponibles sur sirius.nathan.fr.**

*Cet exercice est proposé avec **trois niveaux de difficulté différents** :*

*– un niveau **INITIATION** à destination des élèves n'ayant jamais utilisé le langage de programmation Python ou ayant de grandes difficultés avec celui-ci. Dans cette première version, il n'est pas demandé de créer un programme : l'élève doit uniquement lire les résultats expérimentaux indiqués dans l'énoncé de l'exercice puis les recopier dans les lignes 11 et 12 du code source Python avant d'exécuter le programme ;*

*– un niveau **INTERMÉDIAIRE** à destination des élèves ayant déjà utilisé le langage de programmation Python mais ayant encore des difficultés pour l'appliquer. Dans cette version, l'élève ne doit pas créer un programme mais il doit être capable de compléter*

plusieurs lignes d'un code déjà fourni selon les consignes inscrites en commentaires ;

*– un niveau **CONFIRMÉ** à destination des élèves maîtrisant le langage de programmation Python. Dans cette version, l'élève doit compléter plus d'une quinzaine de lignes d'un code déjà fourni selon les consignes inscrites en commentaires.*

*Quel que soit le niveau choisi, les codes sources proposés sont construits de manière à ce que l'élève puisse travailler en autonomie en s'appuyant sur le **Point numérique 1** du manuel.*

*L'objectif des exercices utilisant un langage de programmation est de **travailler les capacités exigibles de physique-chimie** en effectuant un changement de registre. Le but principal de ces exercices n'est pas l'apprentissage du langage de programmation Python.*

Pour cela, il est important d'insister lors de la présentation de cet exercice sur les nombreuses fonctions d'une résistance dans un circuit électrique et sur l'importance de l'exploitation de sa caractéristique, pour déterminer par exemple la valeur de sa résistance.

Il peut être intéressant de rappeler qu'une résistance peut servir de résistance de protection (voir l'exercice 32 p. 322 du manuel) ou de capteur d'intensité de courant électrique (voir la synthèse page 317 ainsi que les activités 7-8 p. 314-315, les exercices 35-36 p. 322, 50-51 page 326, 52 p. 327 et l'ÉCE p. 329 du manuel).

Remarques :

• Le langage de programmation Python a été choisi car il est préconisé dans les programmes de physique-chimie et enseigné en classe de 2de en mathématiques et en sciences numériques et technologie. Le niveau de maîtrise de ce langage de programmation est toutefois différent pour chaque élève et dépend aussi de la période de l'année à laquelle cette activité est réalisée, c'est pourquoi trois niveaux de difficulté différents sont proposés pour cet exercice.

*• Les fichiers numériques **INITIATION**, **INTERMÉDIAIRE** ou **CONFIRMÉ** proposés peuvent être utilisés quel que soit l'environnement de travail utilisé au lycée (tous les fichiers sont proposés avec deux extensions différentes : .py pour les environnements Spyder, Edupython, Pyzo, Pycharm, IDLE, etc. et .ipynb pour les calepins du logiciel Jupyter dans les interfaces JupyterLab, Colaboratory, Azure Notebook).*

• Si aucun environnement de travail n'est encore installé au lycée, nous conseillons d'installer la suite Anaconda : anaconda.com/distribution/ *(version 3.7 de Python à la date de la rédaction de ce guide pédagogique), et d'utiliser dans la suite Anaconda l'environnement Spyder, tel que cela est préconisé par l'inspection générale de sciences physiques.*

Code source proposé pour le niveau INITIATION :

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  # ====================================================================
4  # 2C16 Exercice 33 page 322                                 INITIATION
5  # Programme permettant de représenter et de modéliser
6  # la caractéristique U=f(I) d'une résistance
7  # ====================================================================
8  import numpy as np
9  from matplotlib import pyplot as plt
10
11 U = [.....................................] # Valeurs de U (en V)
12 I = [.....................................] # Valeurs de I (en mA)
13
14 # Figure représentant la caractéristique U=f(I)
15
16 plt.figure('Etude d\'une résistance')# Initialisation de la figure
17 plt.title('Caractéristique U=f(I)')  # Titre du graphe
18 plt.xlabel('I (en mA)')              # Légende axe I
19 plt.ylabel('U (en V)')               # Légende axe U
20 plt.axis([min(I),max(I),min(U),max(U)])# Minimum et maximum des axes
21 plt.plot(I,U,'r+',ms=10,label='U=f(I) exp')# Trace le nuage de points
22                             # '+' rouge taille 10, nommé U=f(I) exp
23
24 # Modélisation de la caractéristique par une fonction linéaire
25
26 # Modélisation du nuage de points par une droite d'équation U = a*I+b
27 # Calcule les valeurs de a et b grâce à la fonction np.polyfit(I,U,1)
28 # et les range dans cet ordre dans un tableau nommé Modèle
29 Modele = np.polyfit(I,U,1)
30 # Affecte les valeurs rangées dans Modele aux variables a et b
31 a,b = Modele[0],Modele[1]
32
33 # D'après une étude statistique préalable des incertitudes de mesure,
34 # l'approximation linéaire U_mod=a*I de la modélisation est possible
35 # si la valeur absolue de b est inférieure à 0.04 V:
36 if abs(b)<0.04:
37     U_mod = [a*i for i in I]    # Calcule les ordonnées de U_mod =a*I
38     plt.plot(I,U_mod,'b-',label='U=f(I) modélisé') # Trace U_mod=f(I)
39                           # points bleus reliés, nommé U=f(I) modélisé
40     # Affiche le texte:
41     # 'Modélisation de la caractéristique de la résistance'
42     print('Modélisation de la caractéristique de la résistance')
43     # Affiche l'équation U=f(I) de la droite modélisant la
44     # caractéristique en arrondissant la valeur de a à 4 décimales
45     # et en précisant les unités de U et de I
46     print('U(en V) =',round(a,4),'x I(en mA)')
47 else: # sinon
48     # Affiche le texte : Le dipole ne semble pas être une résistance.
49     print('Le dipole ne semble pas être une résistance...')
50
51 plt.grid()                # Affiche une grille
52 plt.legend()              # Affiche la légende
53 plt.show()                # Affiche la figure
```

Aides :

Dans le langage de programmation Python :

– un nombre décimal s'écrit avec un point (et pas avec une virgule) ;

– dans une liste, deux nombres différents sont séparés par une virgule ;

– les deux listes U et I des lignes 11 et 12 doivent contenir le même nombre d'éléments ;

– les lignes commençant par # sont des commentaires permettant de donner des consignes ou de mieux comprendre le programme ; ce ne sont pas des lignes de code.

Commentaires concernant les codes proposés :

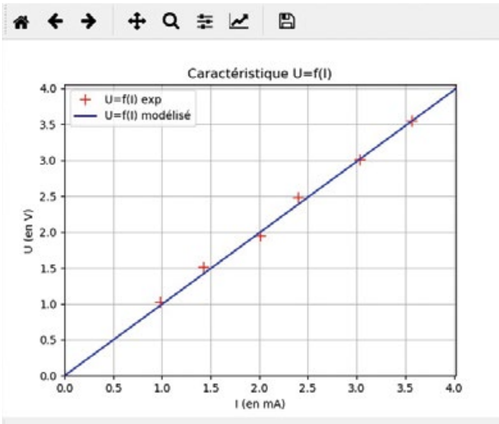
*Les codes sources proposés pour les trois niveaux de difficulté différents permettent de modéliser les nuages de points expérimentaux par une droite d'équation $U = a \times I + b$ puis d'en faire l'approximation linéaire $U = a \times I$ lorsque la valeur absolue de b est inférieure à 0,04 V. Cette valeur a été obtenue à partir de l'étude statistique d'une série de 30 mesures d'intensité et de tension aux bornes d'une même résistance ($R = 1$ kΩ). Pour plus d'informations sur les calculs d'incertitudes, voir la **Fiche méthode 1** du manuel.*

Correction de l'exercice pour le niveau INITIATION :

a.

```
11 U = [0.00,1.03,1.51,1.95,2.48,3.01,3.55,4.05] # Valeurs de U (en V)
12 I = [0.00,0.98,1.43,2.01,2.40,3.03,3.57,4.02] # Valeurs de I (en mA)
```

Exécution du programme :




```
Modélisation de la caractéristique de la résistance
U(en V) = 0.9936 x I(en mA)
```

Dans l'environnement de travail Spyder, l'exécution du programme peut être réalisée en appuyant sur la touche F5.

b. L'expression du modèle donnée par l'exécution du code Python donne : U (en V) $= 0{,}9936 \times I$ (en mA).

D'après la loi d'Ohm, la tension U aux bornes d'une résistance est proportionnelle à l'intensité I qui la traverse soit $U = R \times I$.

Ainsi, d'après ce qui précède et la loi d'Ohm, la valeur de la résistance R est la valeur du coefficient directeur de la droite $U = f(I)$ modélisée, soit $R = 0{,}9936\ \mathrm{V \cdot mA^{-1}} = \dfrac{0{,}9936\ \mathrm{V}}{10^{-3}\ \mathrm{A}} = 993{,}6\ \Omega$.

Les valeurs expérimentales de l'énoncé étant données avec trois chiffres significatifs, on écrit :

$R = 994\ \Omega$.

Il peut être nécessaire de rappeler aux élèves que le coefficient directeur d'une droite a une unité : celle des valeurs en ordonnées (ici V) divisée par celle des valeurs en abscisse (ici mA). Afin d'obtenir des ohms, il faut que ces unités soient converties en celles du Système international.

Code source proposé pour le niveau INTERMÉDIAIRE :

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
# ==========================================================================
# 2C16 Exercice 33 page 322                                    INTERMEDIAIRE
# Programme permettant de représenter et de modéliser
# la caractéristique U=f(I) d'une résistance
# ==========================================================================
import numpy as np
from matplotlib import pyplot as plt

U = [....................................] # Valeurs de U (en V)
I = [....................................] # Valeurs de I (en mA)

# Figure représentant la caractéristique U=f(I)

plt.figure('Etude d\'une résistance')# Initialisation de la figure
plt.title('Caractéristique U=f(I)')  # Titre du graphe
plt.xlabel('.........')              # Légende axe I
plt.ylabel('.........')              # Légende axe U
plt.axis([min(I),max(I),min(U),max(U)])# Minimum et maximum des axes
plt.plot(..., ... , ... ,ms= ... ,label='..........')# Trace le nuage
                    # de points '+' rouge taille 10, nommé U=f(I) exp

# Modélisation de la caractéristique par une fonction linéaire

# Modélisation du nuage de points par une droite d'équation U = a*I+b
# Calcule les valeurs de a et b grâce à la fonction np.polyfit(I,U,1)
# et les range dans cet ordre dans un tableau nommé Modèle
....... = np.polyfit(I,U,1)
# Affecte les valeurs rangées dans Modele aux variables a et b
...,... = Modele[...],Modele[...]

# D'après une étude statistique préalable des incertitudes de mesure,
# l'approximation linéaire U_mod=a*I de la modélisation est possible
# si la valeur absolue de b est inférieure à 0.04 V:
if abs(b)< ... :
    U_mod = [a*i for i in ...]  # Calcule les ordonnées de U_mod =a*I
    plt.plot(...,...,...,label='.........')        # Trace U_mod=f(I)
                         # points bleus reliés, nommé U=f(I) modélisé
    # Affiche le texte:
    # 'Modélisation de la caractéristique de la résistance'
    print('..................................')
    # Affiche l'équation U=f(I) de la droite modélisant la
    # caractéristique en arrondissant la valeur de a à 4 décimales
    # et en précisant les unités de U et de I
    print('......... =',round(a,...),'x .........')
else: # sinon
    # Affiche le texte : Le dipole ne semble pas être une résistance.
    .....('..................................')

..................                 # Affiche une grille
..................                 # Affiche la légende
..................                 # Affiche la figure
```

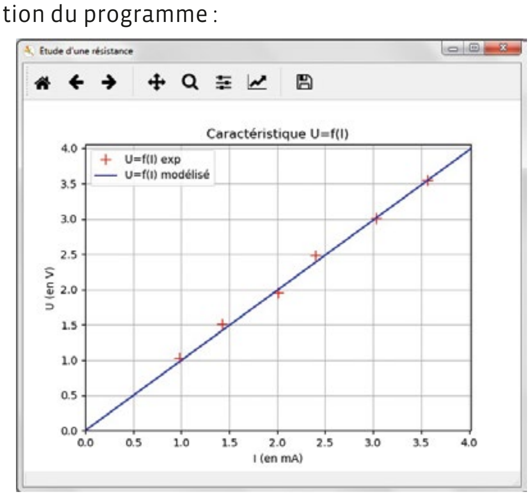
*Le **Point numérique 1** du manuel donne toutes les aides nécessaires pour résoudre cet exercice.*

Correction de l'exercice pour le niveau INTERMÉDIAIRE :

a.

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
# ==========================================================================
# 2C16 Exercice 33 page 322                                       CORRECTION
# Programme permettant de représenter et de modéliser
# la caractéristique U=f(I) d'une résistance
# ==========================================================================
import numpy as np
from matplotlib import pyplot as plt

U = [0.00,1.03,1.51,1.95,2.48,3.01,3.55,4.05] # Valeurs de U (en V)
I = [0.00,0.98,1.43,2.01,2.40,3.03,3.57,4.02] # Valeurs de I (en mA)

# Figure représentant la caractéristique U=f(I)

plt.figure('Etude d\'une résistance')# Initialisation de la figure
plt.title('Caractéristique U=f(I)')  # Titre du graphe
plt.xlabel('I (en mA)')              # Légende axe I
plt.ylabel('U (en V)')               # Légende axe U
plt.axis([min(I),max(I),min(U),max(U)])# Minimum et maximum des axes
plt.plot(I,U,'r+',ms=10,label='U=f(I) exp')# Trace le nuage de points
                       # '+' rouge taille 10, nommé U=f(I) exp

# Modélisation de la caractéristique par une fonction linéaire

# Modélisation du nuage de points par une droite d'équation U = a*I+b
# Calcule les valeurs de a et b grâce à la fonction np.polyfit(I,U,1)
# et les range dans cet ordre dans un tableau nommé Modele
Modele = np.polyfit(I,U,1)
# Affecte les valeurs rangées dans Modele aux variables a et b
a,b = Modele[0],Modele[1]

# D'après une étude statistique préalable des incertitudes de mesure,
# l'approximation linéaire U_mod=a*I de la modélisation est possible
# si la valeur absolue de b est inférieure à 0.04 V:
if abs(b)<0.04:
    U_mod = [a*i for i in I]    # Calcule les ordonnées de U_mod =a*I
    plt.plot(I,U_mod,'b-',label='U=f(I) modélisé') # Trace U_mod=f(I)
                         # points bleus reliés, nommé U=f(I) modélisé
    # Affiche le texte:
    # 'Modélisation de la caractéristique de la résistance'
    print('Modélisation de la caractéristique de la résistance')
    # Affiche l'équation U=f(I) de la droite modélisant la
    # caractéristique en arrondissant la valeur de a à 4 décimales
    # et en précisant les unités de U et de I
    print('U(en V) =',round(a,4),'x I(en mA)')
else: # sinon
    # Affiche le texte : Le dipole ne semble pas être une résistance.
    print('Le dipole ne semble pas être une résistance...')

plt.grid()                      # Affiche une grille
plt.legend()                    # Affiche la légende
plt.show()                      # Affiche la figure
```

Exécution du programme :




```
Modélisation de la caractéristique de la résistance
U(en V) = 0.9936 x I(en mA)
```

b. Même correction que pour le niveau INITIATION.

Code source proposé pour le niveau CONFIRMÉ :

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 # ==========================================================================
4 # 2C16 Exercice 33 page 322                                        CONFIRME
5 # Programme permettant de représenter et de modéliser
6 # la caractéristique U=f(I) d'une résistance
7 # ==========================================================================
8 import numpy as np
9 from matplotlib import pyplot as plt
10
11 U = [......................................] # Valeurs de U (en V)
12 I = [......................................] # Valeurs de I (en mA)
13
14 # Figure représentant la caractéristique U=f(I)
15
16 plt.figure('Etude d\'une résistance')# Initialisation de la figure
17 ...................                   # Titre du graphe
18 ...................                   # Légende axe I
19 ...................                   # Légende axe U
20 ...................                   # Minimum et maximum des axes
21 ...................                   # Trace le nuage de points
22                                 # '+' rouge taille 10, nommé U=f(I) exp
23
24 # Modélisation de la caractéristique par une fonction linéaire
25
26 # Modélisation du nuage de points par une droite d'équation U = a*I+b
27 # Calcule les valeurs de a et b grâce à la fonction np.polyfit(I,U,1)
28 # et les range dans cet ordre dans un tableau nommé Modèle
29 ....... = ............(...,...,...)
30 # Affecte les valeurs rangées dans Modele aux variables a et b
31 ...,... = ...................
32
33 # D'après une étude statistique préalable des incertitudes de mesure,
34 # l'approximation linéaire U_mod=a*I de la modélisation est possible
35 # si la valeur absolue de b est inférieure à 0.04 V:
36 ... abs(b) < ... :
37     ..... = [a*i for i in ...]# Calcule les ordonnées de U_mod =a*I
38     ...................................          # Trace U_mod=f(I)
39                             # points bleus reliés, nommé U=f(I) modélisé
40     # Affiche le texte :
41     # 'Modélisation de la caractéristique de la résistance'
42     print('...................................')
43     # Affiche l'équation U=f(I) de la droite modélisant la
44     # caractéristique en arrondissant la valeur de a à 4 décimales
45     # et en précisant les unités de U et de I
46     ......('......',round(...,...),'......')
47 ...... : # sinon
48     # Affiche le texte : Le dipole ne semble pas être une résistance
49     ......('....................................')
50
51 ...................                 # Affiche une grille
52 ...................                 # Affiche la légende
53 ...................                 # Affiche la figure
```

*• Le **Point numérique 1** du manuel donne les aides nécessaires pour résoudre cet exercice.*

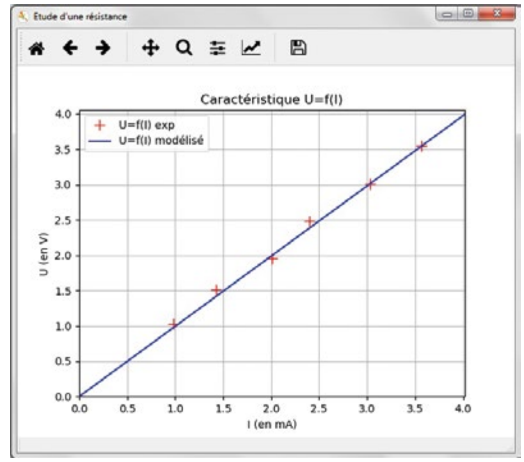
• La ligne 20 permet de définir les limites des axes et d'imposer les échelles des axes à la fenêtre graphique qui est par défaut calibrée automatiquement. Deux syntaxes sont envisageables :
– soit à l'aide de l'instruction plt.axis() qui permet de définir simultanément les limites des deux axes :
20 plt.axis([min(I),max(I),min(U),max(U)]) ;
– soit à l'aide des instructions plt.xlim() et plt.ylim() qui permettent de les définir séparément et dans ce cas, la virgule entre les deux instructions est indispensable :
20 plt.xlim(min(I),max(I)), plt.ylim(min(U),max(U)).

Correction de l'exercice pour le niveau CONFIRMÉ :

a.

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 # ==========================================================================
4 # 2C16 Exercice 33 page 322                                      CORRECTION
5 # Programme permettant de représenter et de modéliser
6 # la caractéristique U=f(I) d'une résistance
7 # ==========================================================================
8 import numpy as np
9 from matplotlib import pyplot as plt
10
11 U = [0.00,1.03,1.51,1.95,2.48,3.01,3.55,4.05] # Valeurs de U (en V)
12 I = [0.00,0.98,1.43,2.01,2.40,3.03,3.57,4.02] # Valeurs de I (en mA)
13
14 # Figure représentant la caractéristique U=f(I)
15
16 plt.figure('Etude d\'une résistance')# Initialisation de la figure
17 plt.title('Caractéristique U=f(I)')  # Titre du graphe
18 plt.xlabel('I (en mA)')              # Légende axe I
19 plt.ylabel('U (en V)')               # Légende axe U
20 plt.axis([min(I),max(I),min(U),max(U)])# Minimum et maximum des axes
21 plt.plot(I,U,'r+',ms=10,label='U=f(I) exp')# Trace le nuage de points
22                                 # '+' rouge taille 10, nommé U=f(I) exp
23
24 # Modélisation de la caractéristique par une fonction linéaire
25
26 # Modélisation du nuage de points par une droite d'équation U = a*I+b
27 # Calcule les valeurs de a et b grâce à la fonction np.polyfit(I,U,1)
28 # et les range dans cet ordre dans un tableau nommé Modèle
29 Modele = np.polyfit(I,U,1)
30 # Affecte les valeurs rangées dans Modele aux variables a et b
31 a,b = Modele[0],Modele[1]
32
33 # D'après une étude statistique préalable des incertitudes de mesure,
34 # l'approximation linéaire U_mod=a*I de la modélisation est possible
35 # si la valeur absolue de b est inférieure à 0.04 V:
36 if abs(b)<0.04:
37     U_mod = [a*i for i in I]    # Calcule les ordonnées de U_mod =a*I
38     plt.plot(I,U_mod,'b-',label='U=f(I) modélisé') # Trace U_mod=f(I)
39                             # points bleus reliés, nommé U=f(I) modélisé
40     # Affiche le texte:
41     # 'Modélisation de la caractéristique de la résistance'
42     print('Modélisation de la caractéristique de la résistance')
43     # Affiche l'équation U=f(I) de la droite modélisant la
44     # caractéristique en arrondissant la valeur de a à 4 décimales
45     # et en précisant les unités de U et de I
46     print('U(en V) =',round(a,4),'x I(en mA)')
47 else: # sinon
48     # Affiche le texte : Le dipole ne semble pas être une résistance.
49     print('Le dipole ne semble pas être une résistance...')
50
51 plt.grid()                        # Affiche une grille
52 plt.legend()                      # Affiche la légende
53 plt.show()                        # Affiche la figure
```

Exécution du programme :




```
Modélisation de la caractéristique de la résistance
U(en V) = 0.9936 x I(en mA)
```

c. Même correction que pour le niveau INITIATION.

d.

Pour aller plus loin :

• La représentation graphique dans le langage de programmation Python se fait grâce au module pyplot de la bibliothèque spécifique matplotlib. Pour tracer une courbe, les coordonnées des points sont rangées dans des objets dont la nature diffère suivant la syntaxe choisie pour écrire le programme.
Dans une première syntaxe, utilisée dans cette activité, les coordonnées des points sont rangées dans des objets de type « liste », les listes étant des objets évoqués en mathématiques dès la classe de 2de lors de l'étude des statistiques, puis explicitement étudiés comme nouvelle notion en classe de 1re. L'avantage de cette syntaxe est donc d'être en accord avec les notions rencontrées en mathématiques sans en introduire de nouvelles. De plus, la manipulation de listes permet de renforcer les connaissances de collège sur l'utilisation des boucles itératives et la construction des algorithmes.
*Une deuxième syntaxe, utilisée dans l'**Activité 4** du **Chapitre 9**, fait appel à une deuxième bibliothèque, la bibliothèque Numpy, habituellement employée pour la représentation graphique et le traitement de données scientifiques. Les coordonnées des points sont rangées dans des objets de type « tableau à une dimension » dont le comportement fait appel à des règles spécifiques (pour plus d'informations, voir le **Point numérique 1** du manuel. L'avantage de cette bibliothèque est sa simplicité d'utilisation et la facilité de lecture du code pour un programme simple de tracé de courbes.*

• Pour assurer une lecture du code source Python sous Linux ou sous Mac quel que soit l'environnement de travail choisi, il est nécessaire de rajouter en début de programme les 2 lignes de commande :

#!/usr/bin/env python3
 # -- coding: utf-8 -*-*

La première ligne indique le chemin d'accès à l'environnement Python, la deuxième ligne permet la lecture des caractères accentués.

• Les codes sources présentés dans les pages précédentes sont les fichiers .py disponibles à l'adresse sirius.nathan.fr *et dans le manuel numérique. Les codes sources des fichiers .ipynb bientôt disponibles à cette même adresse et dans le manuel numérique seront différents afin de tenir compte des spécificités de Jupyter.*

34 ORAL *Vérifier que la présentation utilise la notion de courbe d'étalonnage et de capteur résistif et propose un montage utilisant au minimum un générateur idéal de tension, une thermistance et une résistance utilisée comme capteur de courant associé à un voltmètre ou un microcontrôleur pour pouvoir mesurer la valeur de la tension aux bornes de la résistance. La présentation doit également utiliser la loi des mailles et la loi d'Ohm.*

35 La température peut se mesurer à l'aide d'une thermistance, l'intensité d'un courant électrique se mesure à l'aide d'une résistance et l'éclairement peut se mesurer à l'aide d'une photodiode.

36 Citer des exemples de capteurs

Corrigé dans le manuel.

37 Apprendre à rédiger

Lorsque la température vaut 50°C, la valeur de la résistance vaut :

$R_{50°C} = \dfrac{98{,}8}{\sqrt{\theta}} - 10{,}0$ avec R en kΩ et $\theta = 50$°C.

La thermistance est un capteur résistif et obéit donc à la loi d'Ohm : $U = R_\theta \times I$ avec U la tension aux bornes de la résistance, I l'intensité du courant électrique qui traverse la résistance et R_θ la résistance de la thermistance à la température θ.

Si la thermistance portée à une température de 50°C est parcourue par un courant d'intensité $I = 0{,}50$ mA, alors la tension aux bornes de la thermistance vaut : $U = R_{50°C} \times I = \left(\dfrac{98{,}8}{\sqrt{\theta}} - 10{,}0\right) \times I$.

A.N. : $U = \left(\dfrac{98{,}8}{\sqrt{50}} - 10{,}0\right) \text{k}\Omega \times 0{,}50 \text{ mA} = 3{,}97 \text{ k}\Omega \times 0{,}50 \text{ mA}$

$= 3{,}97 \times 10^3 \ \Omega \times 0{,}50 \times 10^{-3} \text{ A} = 1{,}985 \text{ V}.$

En ne gardant que deux chiffres significatifs comme la donnée la moins précise (0,50 mA), on obtient $U = 2{,}0$ V.

38 Exploiter une série de mesures

DIFFÉRENCIATION

Corrigé dans le manuel.

Exercices 39 et 40

Corrigés dans le manuel.

41 Témoin lumineux de surchauffe moteur

Exercice résolu, corrigé dans le manuel.

42 Témoin lumineux de risque de verglas

APPLICATION

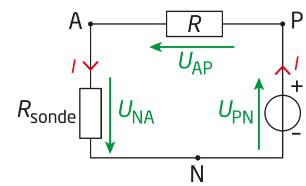
En reprenant la méthode de l'exercice 41 précédent, et à partir du schéma du circuit électrique associé :

$U_R = 1{,}36$ V lorsque le témoin d'alerte de risque de verglas s'allume.

Avec la loi d'Ohm $U_R = R \times I$, on obtient $I = \dfrac{U_R}{R}$.

A.N. : $I = \dfrac{U_R}{R} = \dfrac{1{,}36 \text{ V}}{10{,}0 \times 10^3 \ \Omega} = 1{,}36 \times 10^{-4}$ A.

On applique ensuite la loi des mailles après avoir représenté sur le schéma les tensions aux bornes des dipôles du circuit de détection :



$U_{PN} + U_{NA} + U_{AP} = 0$ soit, avec $U_{AP} = -U_R$ et $U_{NA} = -U_{sonde}$, $U_g - U_R - U_{sonde} = 0$.

A.N. : $U_{sonde} = U_g - U_R = 5{,}00 \text{ V} - 1{,}36 \text{ V} = 3{,}64$ V.

Les deux résultats précédents permettent de calculer la valeur de la résistance de la thermistance :

$U_{sonde} = R_{sonde} \times I$ soit $R_{sonde} = \dfrac{U_{sonde}}{I}$, ce qui permet ensuite d'obtenir la température θ_d :

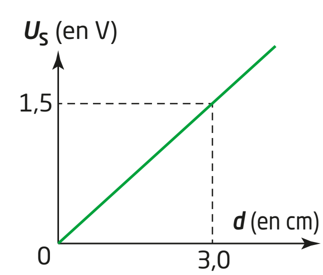
$$\theta_d = \sqrt{\frac{4{,}1 \times 10^4}{R_{sonde} + 1{,}65}} - 35 = \sqrt{\frac{4{,}1 \times 10^4}{\frac{U_{sonde}}{I} + 1{,}65}} - 35.$$

$$\text{A.N. : } \theta_d = \sqrt{\frac{4{,}1 \times 10^4}{\frac{U_{sonde}}{I} + 1{,}65}} - 35 = \sqrt{\frac{4{,}1 \times 10^4}{\frac{3{,}64}{1{,}36 \times 10^{-4}} + 1{,}65}} - 35 = 2{,}98 \text{ °C}$$

que l'on arrondit à 3,0 °C en ne gardant que deux chiffres significatifs.

43 Capteur de déplacement

a. La tension de sortie U_S du capteur est proportionnelle au déplacement d du fader donc la courbe d'étalonnage est une fonction linéaire (droite qui passe par l'origine du repère utilisé pour la représentation graphique), caractérisée par une tension de sortie de 1,5 V pour un déplacement de 3,0 cm :



b. U_S est proportionnelle au déplacement donc : $U_S = k \times d$ d'où $k = \dfrac{U_S}{d}$.

A.N. : $k = \dfrac{1{,}50 \text{ V}}{3{,}0 \text{ cm}} = 0{,}50 \text{ V} \cdot \text{cm}^{-1}$.

On a donc la relation $U_S = 0{,}50 \times d$ soit $d = \dfrac{U_S}{0{,}50}$.

A.N. : $U_S = 2{,}60$ V donne un déplacement $d = \dfrac{2{,}60 \text{ V}}{0{,}50 \text{ V} \cdot \text{cm}^{-1}} = 5{,}2$ cm.

44 In english please

a. La résistance du capteur décroît lorsque la norme de la force appliquée augmente.

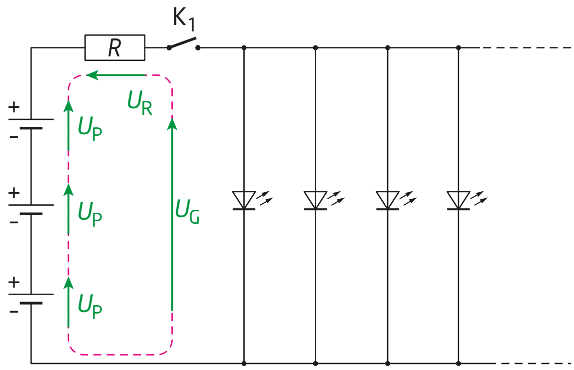
b. Ce capteur est optimisé pour les systèmes électriques directement contrôlés par le toucher humain.

c. La dernière ligne du texte indique que le capteur de force, de type Force Sensing Resistors, n'est pas utilisable pour les mesures de précisions.

45 Guirlande de Noël DIFFÉRENCIATION

Aides en fin de manuel.

a.



b. La loi des mailles dans la maille 1 s'écrit : $U_P + U_P + U_P - U_R - U_G = 0$.
L'interrupteur étant ouvert, l'intensité du courant est nulle dans la résistance R et la tension U_R est nulle également. On a donc :
$3 \times U_P - U_G = 0$ $U_G = 3 \times U_P$.
A.N. : $U_G = 3 \times 1{,}5\ \text{V} = 4{,}5\ \text{V}$.

c. $U_G = 3 \times U_P$ donc $U_P = \dfrac{U_G}{3}$.

A.N. : $U_P = \dfrac{3{,}9\ \text{V}}{3} = 1{,}3\ \text{V}$.

d. Soit I_{DEL} l'intensité qui circule dans une DEL. L'intensité I du courant qui circule dans le générateur est égale à la somme des intensités des courants qui circulent dans les DEL :
$I = I_{DEL} \times$ nombre de DEL.
A.N. : $I = 1{,}1\ \text{mA} \times 30 = 33\ \text{mA}$.

e. Le raisonnement est identique à celui de la question :
$I = I_{DEL} \times$ nombre de DEL.
A.N. : $I = 1{,}1\ \text{mA} \times (30 - 7) = 1{,}1\ \text{mA} \times 23 = 25{,}3\ \text{mA}$.

46 Tension aux bornes d'un fil de connexion

a. Il suffit d'appliquer la formule donnée.

A.N. : $R = \rho \times \dfrac{L}{S} = 1{,}7 \times 10^{-8}\ \Omega \cdot \text{m}^{-1} \times \dfrac{0{,}50\ \text{m}}{1{,}0 \times 10^{-6}\ \text{m}^2} = 8{,}5 \times 10^{-3}\ \Omega$.

b. On applique la loi d'Ohm : $U = R \times I$.

A.N. : $U = 8{,}5 \times 10^{-3}\ \Omega \times 300\ \text{mA} = 8{,}5 \times 10^{-3}\ \Omega \times 3{,}00 \times 10^{-3}\ \text{A}$
$= 25{,}5 \times 10^{-6}\ \text{V} = 2{,}6 \times 10^{-4}\ \text{V} = 0{,}26\ \text{mV}$.

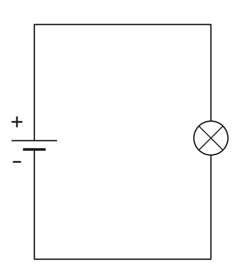
On ne garde que deux chiffres significatifs au résultat final comme la donnée la moins précise qui a servi à faire le calcul.

c. Dans un circuit électrique, les tensions rencontrées sont de l'ordre du volt. On néglige généralement les tensions aux bornes des fils de connexion car celles-ci sont de l'ordre du dixième de mV, soit dix mille fois plus faibles que les autres tensions.

47 La fée électricité

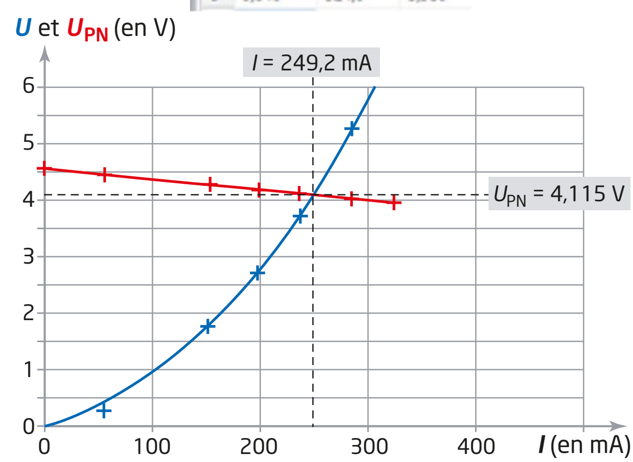
a. Les deux dipôles utilisés dans cette expérience sont une pile et une lampe.

b.



c. et **d.**

i	U	I	Upn
	V	mA	
0	0,000	0,000	4,560
1	0,2500	54,00	4,460
2	1,720	153,0	4,280
3	2,720	198,0	4,200
4	3,720	237,0	4,130
5	5,280	285,0	4,060
6	6,640	324,0	3,980



Les coordonnées du point de fonctionnement sont $I_P = 249$ mA et $U_P = 4{,}12$ V.

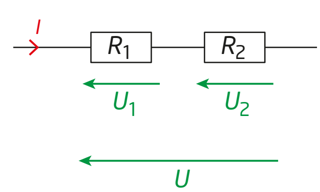
e. On applique la loi d'Ohm : $U = R \times I$, soit $I = \dfrac{U}{R}$, en prenant la valeur minimale de 10 kW de la résistance R et $U = 4{,}5$ V la tension aux bornes de la pile : $I = \dfrac{U}{R}$.

A.N. : $I = \dfrac{4{,}5\ \text{V}}{10\ \text{k}\Omega} = \dfrac{4{,}5\ \text{V}}{10 \times 10^3\ \Omega} = 4{,}5 \times 10^{-4}\ \text{A} = 0{,}45\ \text{mA}$.

On constate que l'intensité de ce courant électrique se situe dans la zone jaune du diagramme et ne présente donc aucun danger d'électrocution quel que soit le temps d'exposition.

48 Résistance équivalente à deux résistances en série

a. b. et **c.**



d. La loi d'additivité des tensions associée à la loi d'Ohm donne :
$U = U_1 + U_2 = R_1 \times I + R_2 \times I = (R_1 + R_2) \times I$.

e. $R_{eq} = \dfrac{U}{I} = \dfrac{(R_1 + R_2) \times I}{I} = R_1 + R_2$.

La résistance équivalente à deux résistances montées en série est égale à la somme des deux résistances.

49 ✱ Modélisation d'une caractéristique

DIFFÉRENCIATION

➜ **Fichiers numériques disponibles sur sirius.nathan.fr.**

Cet exercice est proposé à deux niveaux de difficulté : INTERMÉDIAIRE et CONFIRMÉ.

*Le niveau **CONFIRMÉ** correspond à l'extrait de code donné dans l'énoncé. Quel que soit le niveau choisi, il s'agit de compléter les lignes de code selon les consignes inscrites en commentaires. L'ensemble est construit de manière à ce que l'élève puisse travailler en autonomie en s'appuyant sur le **Point numérique 1** du manuel.*

*Niveau **INTERMÉDIAIRE** : pour les élèves maîtrisant peu les éléments du langage de programmation Python relatifs à la représentation graphique et à la modélisation.*

La fonction np.polyfit(X,Y,1) permettant de modéliser le nuage de points par la fonction d'équation $y = a \times x + b$ est donnée.

```
1 # ==========================================================================
2 # 2C16 Exercice 49 page 325                                    INTERMEDIAIRE
3 # a. Compléter les lignes 9 à 19 et 38 à 40
4 # b. Oter les guillemets lignes 20 et 37. Compléter les lignes 21 à 36.
5 # ==========================================================================
6 import numpy as np
7 from matplotlib import pyplot as plt
8
9 # Définition de 2 listes pour les 2 variables I et U
10 I=[.......................................] # I en mA
11 U=[.......................................] # U en V
12
13 #-----Affichage des points de coordonnées (I,U): U=f(I)
14 plt.figure('Etude d\'un dipole') # Initialisation de la figure
15 plt.title('......')              # Titre du graphe
16 plt.xlabel('.....')              # Légende axe I
17 plt.ylabel('.....')              # Légende axe U
18 plt.axis([...,...,...,...])      # Minimum et maximum des axes
19 plt.plot(...,...,...,ms=...,label='.....') # Trace le nuage de points
20 """
21 # Modélisation du nuage de points par la fonction np.polyfit(X,Y,1)
22 # Calcule les coefficients a et b de la fonction d'équation y = a*x+b
23 # modélisant le nuage de pts et les range dans le tableau nommé Modele
24 ...... = np.polyfit(... , ... , ...)
25
26 # Affecte les coefficients du modèle aux variables a et b
27 a,b = [coef for coef in ......]
28
29 # Liste des ordonnées de la modélisation
30 U_mod = [a*...  for i in I]
31
32 # Trace les points de coordonnées I et U_mod en bleu et reliés
33 plt.plot(... , ... , ... ,label='U=f(I) modélisé')
34
35 print('Expression du modèle')
36 print('U(...) =',round(...,4),'......')# Affiche l'équation du modèle
37 """
38 ............                   # Affiche une grille
39 ............                   # Affiche la légende
40 ............                   # Affiche la figure
```

*Niveau **CONFIRMÉ** : pour les élèves ayant compris les éléments du langage de programmation Python relatifs à la représentation graphique et à la modélisation étudiés dans l'**Activité 6** et l'exercice 33 de ce chapitre.*

```
1 # ==========================================================================
2 # 2C16 Exercice 49 page 325                                        CONFIRME
3 # a. Compléter les lignes 9 à 19 et 38 à 40
4 # b. Oter les guillemets lignes 20 et 37. Compléter les lignes 21 à 36.
5 # ==========================================================================
6 import numpy as np
7 from matplotlib import pyplot as plt
8
9 # Définition de 2 listes pour les 2 variables I et U
10 I=[.......................................] # I en mA
11 U=[.......................................] # U en V
12
13 #-----Affichage des points de coordonnées (I,U): U=f(I)
14 plt.figure('Etude d\'un dipole') # Initialisation de la figure
15 ...................              # Titre du graphe
16 ...................              # Légende axe I
17 ...................              # Légende axe U
18 ...................              # Minimum et maximum des axes
19 ...................              # Trace le nuage de points
20 """
21 #-----Modélisation du nuage de points par la fonction np.polyfit()
22 # Calcule les coefficients de la fonction modélisant le nuage
23 # de points et les range dans un tableau nommé Modele
24 ...... = np.polyfit(... , ... , ...)
25
26 # Affecte les coefficients du modèle aux variables a,b,c...(à choisir)
27 ......... = [coef for coef in ......]
28
29 # Liste des ordonnées de la modélisation
30 U_mod = [..........  for i in I]
31
32 # Trace les points de coordonnées I et U_mod en bleu et reliés
33 plt.plot(... , ... , ... ,label='U=f(I) modélisé')
34
35 print('Expression du modèle')
36 print('U(...) = ...........)  # Affiche l'équation du modèle
37 """
38 ............                   # Affiche une grille
39 ............                   # Affiche la légende
40 ............                   # Affiche la figure
```

a. Comme la consigne l'indique en début de programme quel que soit le niveau de différenciation, il s'agit de compléter les lignes 9 à 19 et 38 à 40 du code source fourni.

```
9 # Définition de 2 listes pour les 2 variables I et U
10 I=[0.,0.99,2.00,2.98,3.95,4.96,5.92] # I en mA
11 U=[0.,1.02,2.03,3.02,4.00,5.03,6.01] # U en V
12
13 #-----Affichage des points de coordonnées (I,U): U=f(I)
14 plt.figure('Etude d\'un dipole')   # Initialisation de la figure
15 plt.title('Caractéristique U=f(I)')# Titre du graphe
16 plt.xlabel('I (en mA)')            # Légende axe I
17 plt.ylabel('U (en V)')             # Légende axe U
18 plt.axis([min(I),max(I),min(U),max(U)])# Minimum et maximum des axes
19 plt.plot(I,U,'r+',ms=10,label='U=f(I) exp')# Trace le nuage de points
20
21 #-----Modélisation du nuage de points par la fonction np.polyfit()
22 # Calcule les coefficients de la fonction modélisant le nuage
23 # de points et les range dans un tableau nommé Modele
24 Modele = np.polyfit(I,U,1)
25
26 # Affecte les coefficients du modèle aux variables a,b,c...(à choisir)
27 a,b = [coef for coef in Modele]
28
29 # Liste des ordonnées de la modélisation
30 U_mod = [a*i for i in I]
31
32 # Trace les points de coordonnées I et U_mod en bleu et reliés
33 plt.plot(I,U_mod,'b-',label='U=f(I) modélisé')
34
35 print('Expression du modèle')
36 print('U(en V)=',round(a,4),'x I(en mA)')#Affiche l'équation du modèle
37
38 plt.grid()                         # Affiche une grille
39 plt.legend()                       # Affiche la légende
40 plt.show()                         # Affiche la figure
```

Pour compléter la ligne 18 du code, deux syntaxes sont envisageables :

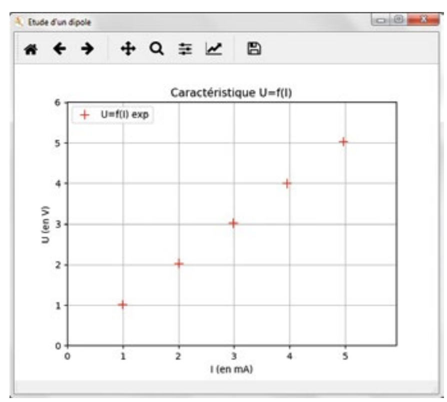
– soit à l'aide de l'instruction plt.axis() qui permet de définir simultanément les limites des deux axes :

18 plt.axis([min(I),max(I),min(U),max(U)])

– soit à l'aide des instructions plt.xlim() et plt.ylim() qui permettent de les définir séparément et dans ce cas, la virgule entre les deux commandes est indispensable :

18 plt.xlim(min(I),max(I)), plt.ylim(min(U),max(U))

L'exécution du programme donne la fenêtre de visualisation suivante :



b. Les points représentant la tension U en fonction de l'intensité I aux bornes du dipôle AB sont au voisinage d'une droite passant par l'origine ; il y a donc proportionnalité entre ces deux grandeurs. Les lignes 21 à 37 sont donc complétées pour proposer une modélisation linéaire de la caractéristique :

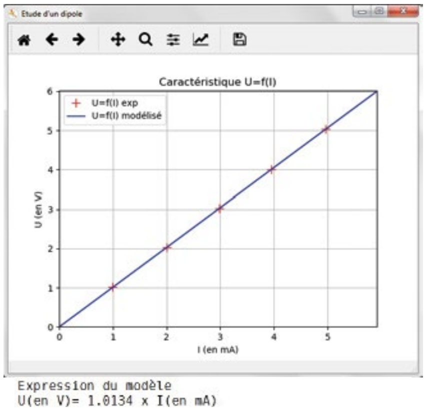
– la ligne 24 calcule les valeurs de a et b de la fonction d'équation $U = a \times I + b$ modélisant le nuage de points grâce à l'instruction np.pypolyfit(I,U,1) ;

– la ligne 27 affecte ces valeurs aux variables a et b ;

– la ligne 30 permet de calculer les ordonnées des points de la droite passant par l'origine d'équation $U\_mod = a \times I$ modélisant la caractéristique.

```
21 #-----Modélisation du nuage de points par la fonction np.polyfit()
22 # Calcule les coefficients de la fonction modélisant le nuage
23 # de points et les range dans un tableau nommé Modele
24 Modele = np.polyfit(I,U,1)
25
26 # Affecte les coefficients du modèle aux variables a,b,c...(à choisir)
27 a,b = [coef for coef in Modele]
28
29 # Liste des ordonnées de la modélisation
30 U_mod = [a*i for i in I]
31
32 # Trace les points de coordonnées I et U_mod en bleu et reliés
33 plt.plot(I,U_mod,'b-',label='U=f(I) modélisé')
34
35 print('Expression du modèle')
36 print('U(en V)=',round(a,4),'x I(en mA)')#Affiche l'équation du modèle
37
```

L'exécution du programme donne la fenêtre de visualisation suivante :



L'expression de modèle donnée par l'exécution du code Python donne : U (en V) = 1,0134 × I (en mA).
Les valeurs expérimentales de l'énoncé sont presque toutes données avec trois chiffres significatifs donc on écrit U (en V) = 1,01 × I (en mA).

c. D'après ce qui précède, comme la tension U aux bornes du dipôle AB est proportionnelle à l'intensité I du courant qui le traverse, le dipôle AB est une résistance.
Selon la loi d'Ohm, la valeur de la résistance R est la valeur du coefficient directeur de la droite modélisée, soit :

$$R = \frac{1{,}01\ \text{V}}{\text{mA}} = \frac{1{,}01\ \text{V}}{10^{-3}\ \text{A}} = 1{,}01 \times 10^{3}\ \Omega.$$

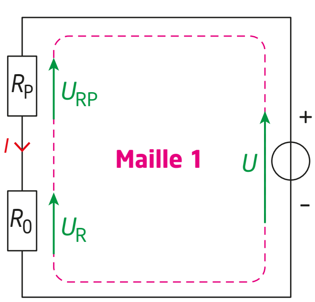
Soit R = 1,01 kΩ.

Il peut être ici nécessaire de rappeler aux élèves que le coefficient directeur d'une droite a une unité : celle des valeurs en ordonnées (ici V) divisée par celle des valeurs en abscisse (ici mA). Afin d'obtenir des ohms, il faut que ces unités soient converties en celles du Système international.
Pour assurer une lecture du code source Python sous Linux ou sous Mac quel que soit l'environnement de travail choisi, il peut être nécessaire de rajouter en début de programme les 2 lignes de commande :

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
```

50 ✱✱ Retour sur l'ouverture du chapitre

a.



Dans la maille 1 comprenant le générateur idéal de tension U, la photorésistance R_p et la résistance R :

$$U - U_{Rp} - U_R = 0.$$

L'application de la loi d'Ohm aux bornes de la résistance puis de la photorésistance donne :
$U_R = R \times I$ et $U_{Rp} = R_p \times I$, soit avec la loi des mailles précédente :
$U = U_R + U_{Rp} = R \times I + R_p \times I = (R + R_p) \times I$ soit $I = \dfrac{U}{(R + R_p)}$.

On a donc $U_R = R \times I = \dfrac{U \times R}{(R + R_p)}$.
Avec la tension délivrée par le microcontrôleur U = 5 V, on obtient la relation : $U_R = \dfrac{5 \times R}{(R + R_p)}$.

b. Pour R = 2,0 kΩ : E = 10 lux donne $U_R = \dfrac{5{,}00\ \text{V} \times 2{,}0\ \text{k}\Omega}{(2{,}0\ \text{k}\Omega + 9{,}43\ \text{k}\Omega)} = 0{,}87$ V

E = 350 lux donne $U_R = \dfrac{5{,}00\ \text{V} \times 2{,}00\ \text{k}\Omega}{(2{,}0\ \text{k}\Omega + 0{,}58\ \text{k}\Omega)} = 3{,}9$ V

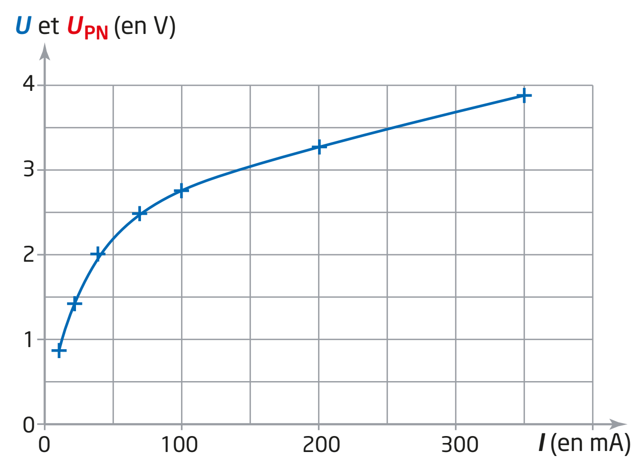
Pour R = 10,0 kΩ : E = 10 lux donne $U_R = \dfrac{5{,}00\ \text{V} \times 10{,}0\ \text{k}\Omega}{(10{,}0\ \text{k}\Omega + 9{,}43\ \text{k}\Omega)} = 2{,}57$ V

E = 350 lux donne $U_R = \dfrac{5{,}00\ \text{V} \times 10{,}0\ \text{k}\Omega}{(10{,}0\ \text{k}\Omega + 0{,}58\ \text{k}\Omega)} = 4{,}73$ V.

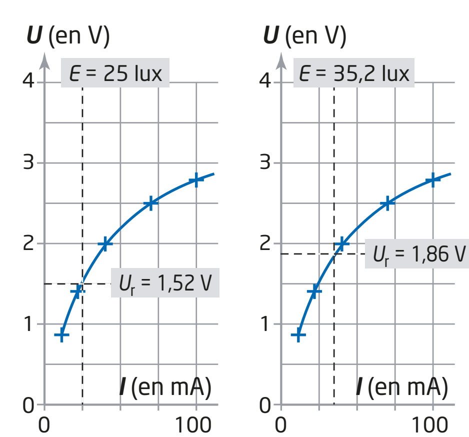
On constate que la plage de tension pour la résistance de 2,0 kΩ s'étale de 0,87 à 3,9 V soit une variation de 3,9 – 0,87 = 3,03 V pour des valeurs d'éclairement allant de 10 à 350 lux alors que la plage de tension couverte pour la résistance de 10,0 kΩ s'étale de 2,57 V à 4,73 V ce qui donne une variation de 4,73 – 2,57 = 2,16 V pour la même plage de variation des valeurs de l'éclairement : la sensibilité du capteur de lumière sera donc meilleure avec une résistance de 2,0 kΩ.

c.

i	Rp	E	Ur
	kOhm	lux	V
0	0,580	350	3,88
1	1,02	201	3,31
2	1,60	100	2,78
3	2,00	70,0	2,50
4	2,98	39,0	2,01
5	5,02	21,0	1,42
6	9,43	10,0	0,875



d.



Pour 25 lux, on obtient $U_R = 1{,}52$ V.
Pour 35 lux, on obtient $U_R = 1{,}86$ V.

e. L'analyse du programme montre que la valeur de la tension U_R, comprise entre 0 et 5,00 V, est recalculée à partir d'une variable *x*, comprise entre 0 et 1023, qui récupère la valeur numérisée de U_R sur l'entrée A0. D'après le programme utilisé, le déclenchement de la production d'un son se fait par l'instruction conditionnelle **if ((Ur > 0,5) && (Ur<1,9))**, ce qui signifie que si la tension U_R est comprise entre 0,5 V et 1,9 V, alors un son de fréquence 440 Hz sera produit sur la sortie numérique 2 pendant une durée de 100 ms.
Le seuil inférieur pour la tension U_R pour le déclenchement du son est donc beaucoup trop bas, ce qui fait que le son est produit pour un éclairement inférieur à 10 lux. Cette condition impose donc une ambiance lumineuse proche de l'obscurité pour éviter que le son soit produit en permanence. Pour éviter ce problème, le programme doit être corrigé en augmentant le seuil inférieur de déclenchement et en le fixant à la valeur de 1,5 V correspondant à la valeur de 25 lux déterminée à la question précédente.

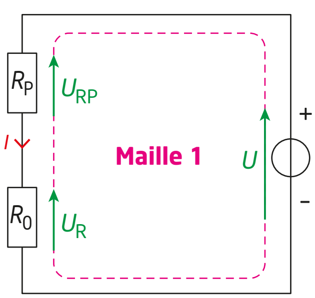
Ce programme est disponible en ligne sur le site ***sirius.nathan.fr*** *et peut être testé avec le montage présenté dans cette exercice. Pour cela, il suffit juste d'ajuster les seuils de déclenchement en fonction de la photorésistance et de la résistance utilisée.*

51 Trigger de batterie expérimental

a. $U_{Rc} = R_c \times I$

A.N. : $U_{Rc} = 300\ \Omega \times 1{,}3\ \text{mA} = 300\ \Omega \times 1{,}3 \times 10^{-3}\ \text{A} = 0{,}39$ V.

b. Le schéma équivalent est :



Dans la maille comprenant le générateur idéal de tension *U*, le capteur de force de résistance R_c et la résistance R_0, on a :
$U - U_{Rc} - U_{R0} = 0$ soit $U_{R0} = U - U_{Rc}$

A.N. : avec $U_{Rc} = 0{,}39$ V et $U = 5{,}00$ V, on obtient :
$U_{R0} = 5{,}00 - 0{,}39 = 4{,}61$ V

L'application de la loi d'Ohm aux bornes de la résistance donne :
$U_{R0} = R_0 \times I$ soit $R_0 = \dfrac{U_{R0}}{I}$.

c. En utilisant la loi des mailles $U - U_{Rc} - U_{R0} = 0$ et la loi d'Ohm aux bornes des deux dipôles résistifs du circuit $U_{max} = R_0 \times I_{max}$ et $U_{Rc} = R_C \times I_{max}$, on obtient :

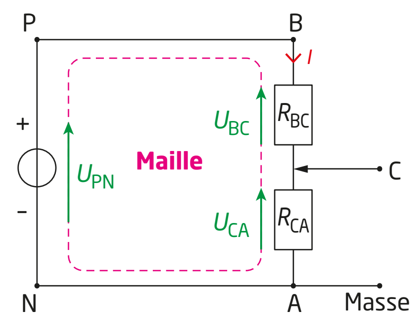
$U - R_C \times I_{max} - R_0 \times I_{max} = 0$ soit $I_{max} = \dfrac{U}{(R_C + R_0)}$ et $U_{max} = R_0 \times I_{max} = \dfrac{R_0 \times U}{(R_C + R_0)}$

A.N. : $U_{max} = \dfrac{10 \times 10 \times 10^3\ \Omega \times 5{,}00\ \text{V}}{(300 + 10 \times 10^3)\ \Omega} = 4{,}85$ V.

d. L'analyse du programme montre que la valeur de la tension U_{R0}, comprise entre 0 et 5,00 V, est numérisée par l'intermédiaire de la variable **capteur** qui donne une valeur comprise entre 0 et 1023.
D'après le programme utilisé, le déclenchement de la production d'un son se fait par l'instruction conditionnelle **if (capteur > 30)**, ce qui signifie que si la variable capteur dépasse la valeur 30 alors un son sera produit.
Cette valeur 30 pour la variable capteur correspond à une valeur de la tension $U_R = \dfrac{10 \times 5{,}00}{1023} = 0{,}14$ V ce qui est une valeur trop faible, rendant le montage trop sensible par l'intermédiaire du programme utilisé. En augmentant cette valeur, on diminue le seuil de déclenchement du son et on diminue donc la sensibilité du montage : en effet, plus la tension U_R est élevée et plus la tension aux bornes du capteur de force sera faible, ce qui indique sa résistance sera faible et donc que la force exercée sur ce capteur sera élevée.

52 Contrôle du son sur une guitare électrique

1.



2. On applique la loi des mailles et la loi d'Ohm :
$U_{PN} - U_{BC} - U_{CA} = 0$, $U_{BC} = R_{BC} \times I$ et $U_{CA} = R_{CA} \times I$.
$U_{PN} - R_{BC} \times I - R_{CA} \times I = 0$.
$(R_{BC} + R_{CA}) \times I = U_{PN}$.
$I = \dfrac{U_{PN}}{(R_{BC} + R_{CA})} = \dfrac{U_{PN}}{R_{BA}}$.

3. $U_{CA} = R_{CA} \times I = \dfrac{R_{CA} \times U_{PN}}{(R_{BC} + R_{CA})} = \dfrac{R_{CA} \times U_{PN}}{R_{BA}}$.

La tension U_{CA} est proportionnelle à la tension U_{PN} et toujours plus faible : le montage est donc appelé diviseur de tension car il divise la tension d'entrée U_{PN} par le rapport $\dfrac{R_{BA}}{R_{CA}}$ qui varie entre un et plus l'infini, ce qui permet d'obtenir une tension U_{CA} variant entre une valeur maximale égale à U_{PN} et une valeur nulle : c'est ainsi que se fait le contrôle du niveau du signal provenant du micro de la guitare électrique.

4. Pour 0° : $R_{CA} = 0$.
Pour 330° : $R_{CA} = R_{BA}$.

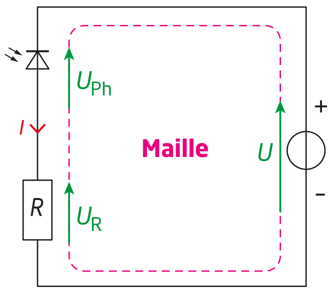
Donc pour 180° : $R_{CA} = \dfrac{R_{BA} \times 180°}{330°} = 0{,}545 \times R_{BA}$.

On en déduit donc $U_{CA} = \dfrac{R_{CA} \times U_{PN}}{R_{BA}} = \dfrac{R_{BA} \times \left(\dfrac{180°}{330°}\right) \times U_{PN}}{R_{BA}} = \dfrac{U_{PN} \times 180°}{330°}$.

A.N. : $U_{CA} = \dfrac{100\ \text{mV} \times 180°}{330°} = 54{,}5$ mV.

53 Détecteur de fumée

1.



La loi des mailles donne : $U - U_R - U_{Ph} = 0$ soit $U_R = U - U_{Ph}$.
D'après la caractéristique de la photodiode pour un éclairement E_1 sans fumée, l'intensité du courant qui traverse la photodiode est quasiment nulle quelle que soit la tension, donc la tension U_1 aux bornes de la résistance R est quasiment nulle car $U_1 = R \times I_1 = R \times 0 = 0$ V.

2. En présence de fumée, l'éclairement E_2 sur la photodiode fait que celle-ci est parcourue par un courant dont l'intensité vaut 8,0 µA. La tension U_2 aux bornes de la résistance R vaut donc $U_2 = R \times I_2$.

A.N. : $U_2 = R \times I_2 = 200\ \text{k}\Omega \times 8\ \mu\text{A} = 200 \times 10^3\ \Omega \times 8{,}0 \times 10^{-6}\ \text{A} = 1{,}6$ V.

3. Pour $U_R = 0$ V : $N = 0$.
Pour $U_R = 5{,}00$ V : $N = 1023$.

Donc pour N = 150 : $U_R = \dfrac{5{,}00 \times 150}{1023} = 0{,}73$ V.

L'alarme se déclenche car la tension aux bornes de U_R en présence de fumées est égale à 1,6 V, ce qui est largement supérieur au seuil de déclenchement de 0,73 V.

54 Principe d'un joystick DIFFÉRENCIATION

➛ **Fichiers d'évaluation par compétences et d'aide à la notation disponibles sur sirius.nathan.fr.**

1. Ajouter une ligne au tableau donné ou effectuer les calculs permettant de vérifier la relation $R_{BA} = R_{BC} + R_{CA}$ pour chaque couple de valeurs R_{CA} et R_{BC}.

Saisir les données des positions angulaires α et de R_{CA} dans un tableur-grapheur puis afficher la représentation graphique $R_{CA} = f(\alpha)$.

2. La représentation graphique de $R_{CA} = f(\alpha)$ est une droite passant par l'origine (fonction linéaire).

3. La tension générée par le microcontrôleur appliquée aux bornes du potentiomètre est de 5,00 V. Le relevé de mesures dans le moniteur série de l'I.D.E. de l'Arduino donne :

Position angulaire › α (en °)	0	60	120	180	240	300
U_{CA} (en V)	0,00	1,00	2,00	3,00	4,00	5,00

On a bien $U_{CA} = k \times \alpha$ avec $k = \dfrac{1\ \text{V}}{60°} = 0{,}0167$ V/° ou plus simplement 60° par volt : le potentiomètre est bien un capteur de position angulaire.
La position du joystick $posJoy_1 = -30°$ doit correspondre à la position $\alpha_1 = 120°$ du potentiomètre et la position du joystick $posJoy_2 = 30°$ doit correspondre à la position $\alpha_2 = 180°$ du potentiomètre. On en déduit que $posJoy = \alpha - 150$.
À la ligne 16 du code, il faut donc dé-commenter (enlever //) et saisir l'expression : *posJoy = alpha* -150.
À la ligne 24, il faut saisir l'intervalle de valeurs limites des positions angulaires du joystick *posJoy*, soit –30° et 30°. Il faut dé-commenter (enlever //) et écrire : if (posJoy >= –30 && posJoy <= 30) {.
De la ligne 27 à la ligne 31, il faut dé-commenter (enlever //) et écrire les conditions pour que l'affichage de la position du joystick dans le moniteur série soit :
– « supérieure à 30° » si posJoy > 30° ;
– « inférieure à 30° » si posJoy < 30°.

4. La variation de tension entre les bornes C et A est proportionnelle à la variation de position angulaire du joystick.
Si le joystick se déplace entre ses deux positions extrêmes, de –30° = 120° sur le potentiomètre à 30° = 180° sur le potentiomètre, alors la tension U_{CA} passe de 2,00 V à 3,00 V, ce qui permet le contrôle d'un système en temps réel par l'action directe de la main sur le joystick.