**FICHE de PRÉSENTATION d’activités**

|  |  |  |
| --- | --- | --- |
| ***Niveau*** | ***seconde*** | |
| ***Séquence*** | ***Signaux et capteurs*** | |
| ***Titre de l’activité*** | ***Caractéristique d’un conducteur ohmique*** | |
| ***Type d'activité*** | ***Expérimentale et programmation en ½ groupe*** | |
| ***Références au programme*** | Notions et contenus  *Caractéristique tension-courant d’un dipôle.*  *Résistance et systèmes à comportement de type ohmique. Loi d’Ohm.* | Capacités exigibles  *Mesurer une tension et une intensité.*  *Représenter et exploiter la caractéristique d’un dipôle.*  ***Capacités numériques :*** *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.* |
| ***Compétences mobilisées*** | ❑ S’approprier ❑ Analyser/raisonner  ❑ Réaliser ❑ Valider | |
| ***Mise en œuvre*** | Pré-requis:  Quelques notions de python (variable, exécution d’un programme sous Edupython) | |
| Durée : 1h30 | |
| Contraintes matérielles : salle informatique avec EduPython | |
| *Liens photos* | *aucun* | |
| *Auteur* | **olivier.chaumette@ac-lyon.fr** | |
| **pour le GRD groupe lycée de l’académie de LYON** | |

**Fiche élève : activité**

Le but de cette activité est de **déterminer la résistance d’un conducteur ohmique inconnu**. Pour cela, il faudra représenter et modéliser la caractéristique du conducteur ohmique inconnu à l’aide d’un programme Python.

**1. Documents: *(compétence travaillée : S’APPROPRIER)***

La **caractéristique** d’un dipôle est la **courbe** représentant les variations de la tension **U** appliquée à ses bornes **en fonction** de l’intensité **I** du courant électrique qui le traverse. *Pour faire simple*, il s’agit de la courbe représentant **les variations de U en fonction de I**.

 

La mesure de la **tension U** aux bornes d’un dipôle se fait à l’aide d’un **voltmètre** branché en **dérivation**.

La mesure de l’**intensité I** du courant traversant un dipôle se fait à l’aide d’un **ampèremètre** branché en **série**.



***Matériel à disposition :***

- un générateur de tension électrique variable.

- un conducteur ohmique de résistance inconnue

- deux multimètres (pouvant jouer le rôle de voltmètre ou ampèremètre)

- des fils de connexion

**2. Protocole permettant de tracer la caractéristique :**

***2.1. Proposition du protocole : (compétence travaillée : ANALYSER)***

En vous appuyant sur les documents du paragraphe 1, **proposer un montage et un protocole simple** permettant d’obtenir, pour une dizaine de valeurs de tension du générateur bien choisies, les valeurs de l’intensité du courant circulant dans le conducteur ohmique.

**Appeler le professeur pour qu’il valide votre protocole**

***2.2. Mise en œuvre du protocole : (compétence travaillée : REALISER)***

Réaliser le montage et mettre en œuvre le protocole. Les résultats des mesures (au moins 10) devront être réunis dans un tableau.

**Appeler le professeur pour qu’il valide vos mesures**

**3. Traitement numérique à l’aide d’un programme en langage Python :**

***3.1. Ouverture et appropriation du programme Python : (compétence travaillée : S’APPROPRIER)***

**1.** Lancer l’éditeur Python (Edupython.exe) et ouvrir le programme **« Caracteristique\_R\_modelisation.py ».**

**2.** Lancer le programme pour vérifier que tout fonctionne correctement. Fermer la fenêtre qui s’est ouverte.

***3.2. Affichage des points associés à la caractéristique : (compétence travaillée : REALISER)***

*Pour l’instant, le programme n’affiche rien. Vous devrez le modifier.*

*Les* ***différentes parties*** *du programme sont repérées par des commentaires précédés du symbole dièse : #*

**1. TRAVAIL 1 :** **Création de listes à partir des valeurs mesurées expérimentalement.** Suivre les consignes écrites en rose dans le programme au niveau de « TRAVAIL 1 ».

**2. TRAVAIL 2 :** **Préparation du graphique.** Suivre les consignes en rose dans le programme au niveau de « TRAVAIL 2» :

- Donner un nom pertinent au graphique

- Ecrire les noms des grandeurs représentées sur les axes **et leur unité**.

- Donner un titre au graphique.

- Assigner à la variable Imax la valeur de la plus grande intensité mesurée. Assigner à la variable Umax la valeur de la plus grande tension mesurée.

**3. TRAVAIL 3 :** **Tracé de la courbe expérimentale**:

Pour tracer la courbe représentant une grandeur Y en fonction d’une grandeur X, il faut taper en Python :

**plt.plot(X,Y,“kx”)** *“kx” signifie que les points affichés seront noirs (“k”) et représentés par des croix (“x”)*

Dans le programme, en dessous des consignes correspondantes à TRAVAIL 3, taper le code permettant de :

- Tracer l’évolution de la tension U en fonction de I, avec des points rouges représentés par une croix (voir annexe)

- Modifier les coordonnées du texte « nuage de points expérimentaux » de manière à ce qu’il soit situé proche des points affichés et de la même couleur que les points. L’instruction permettant cela est :

**plt.text (x,y,"texte",color='k'**) où x et y sont les coordonnées où sera affiché « texte » en noir (k)

***3.3. Analyse de la caractéristique : (compétence travaillée : ANALISER)***

**1.** Que pensez-vous du nuage de points associé à la caractéristique (quel type de courbe forme-t-il) ?

**2.** Quelle fonction mathématique permettrait de modéliser la courbe formée par le nuage de points : Linaire (droite passant par O) ? Affine (droite ne passant pas par O) ? Parabolique ? Autre ?

***3.4. Modélisation de la caractéristique : (compétence travaillée : REALISER)***

**Si la courbe ressemble à:**

|  |  |  |
| --- | --- | --- |
| **Une parabole** | **Une droite passant par l’origine** | **Une droite ne passant pas par l’origine** |
| **Il faut taper :**  **Umod = a\*I\*I** | **Il faut taper :**  **Umod = a\*I** | **Il faut taper :**  **Umod = a\*I + b** |

**1. TRAVAIL 4 :** **Saisie du code et tracé du modèle.** Suivre les consignes en rose au niveau de « TRAVAIL 4» :

- Taper le code correspondant au modèle que vous avez choisi

- Taper le code pour tracer Umod en fonction de I, en bleu avec des points non affichés mais reliés.

- Modifier les coordonnées du texte « Modèle de la caractéristique » (proche de la courbe et même couleur).

**2. Recherche du meilleur modèle :** Modifier,dans le programme, la valeur de la variable “**a”** afin que la courbe modélisée se rapproche le plus possible de la courbe expérimentale.

**4. Exploitation du modèle de la caractéristique :**

***4.1. Réponse à la problématique : (compétence travaillée : S’APPROPRIER)***

**1.** Quelle est l’équation du modèle de la caractéristique (c'est-à-dire la relation numérique en U et I) ?

**2.** En vous appuyant sur les documents du début, répondre à la problématique : **quelle est la valeur de la résistance du conducteur ohmique inconnu ?**

***4.2. Validation de notre résultat : (compétence travaillée : VALIDER)***

**1.** A l’aide d’un multimètre branché en ohm-mètre, mesurer la valeur de la résistance du conducteur ohmique.

**2.** Comparer à celle obtenue suite à votre modélisation.

**ANNEXE -** *Quelques options de mise en forme du tracé avec le module MATPLOTLIB utilisé par Python*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tracé | | Type de points tracés | | | | | Couleurs | | | | |
| - | -- | o | . | x | + | v | r | b | g | k | m |
| Points reliés | Points reliés en pointillé | Gros « ronds » | Petit point | Croix | Croix + | Triangle | Rouge | Bleu | vert | noir | magenta |

**Fiche professeur**

# ##########################################################################

# Programme Python permettant de tracer la caractéristique d'un conducteur ohmique

# et de programmer la modélisation

# à destination des élèves qui peuvent modifier le code sous les parties roses (ou triples guillemets)

# ##########################################################################

# O. CHAUMETTE - Lycée JP SARTRE - 69500 BRON - olivier.chaumette@ac-lyon.fr

# ##########################################################################

# ##########################

# IMPORTATION DES MODULES #

# #########################

# importation de NUMPY (pour gestion tableaux et calculs) sous l'alias "np"

**import** numpy **as** np

# importation de PYPLOT (du module MATPLOTLIB, pour le tracé de courbes) sous l'alias "plt"

**import** matplotlib**.**pyplot **as** plt

# ##############################

# Début du Programme principal #

# #############################

# ###################################################################

# Création de listes à partir des données en abscisse et ordonnée:

""" TRAVAIL 1:

----------

Entrer , ci-dessous, entre les crochets et séparées par des virgules, les différentes valeurs de l'intensité et de la tension """

I **=** np**.**array**([**0**,**0.001**,** 0.002**,** 0.0029**,** 0.004**])**

U **=** np**.**array**([**0**,**2**,** 4.1**,** 5.9**,** 8**])**

""" TRAVAIL 2:

----------

Suivre les consignes écrites en rose ci-dessous: """

# Titre Du graphique (OPTIONNEL, peut-être supprimé)

""" Entrer ci-dessous, entre les guillemets, le titre du graphique SANS ACCENTS !!!! """

plt**.**title**(**"Tension aux bornes du conducteur ohmique en fonction de I"**)**

# Légende des axes (OPTIONNEL, peut-être supprimé)

""" Entrer ci-dessous les noms des grandeurs pour chaque axe

(remplacer "abscisse" et "ordonnee") SANS ACCENTS !!!! """

plt**.**xlabel**(**"Intensité I en A"**)**

plt**.**ylabel**(**"Tension U en V"**)**

# Modification des maxis/minis des axes (OPTIONNEL, peut-être supprimé)

""" Vraiment OPTIONNEL !! MATPLOTLIB peut gérer tout seul l'échelle.

Sinon, modifier les valeurs de Imin,Imax,Umin et Umax ci-dessous """

Imin**=**0

Imax**=**0.005

Umin **=**0

Umax **=** 10

plt**.**axis**([**Imin**,** Imax**,** Umin**,** Umax**])**

# #######################################

# Affichage de la grille (OPTIONNEL, peut-être supprimé)

plt**.**grid**()**

""" TRAVAIL 3:

----------

Suivre les consignes écrites en rose ci-dessous: """

# #######################################

# Tracé de la courbe expérimentale

""" Taper ci-dessous le code permettant de représenter U en fonction de I

sous forme de croix rouges"""

plt**.**plot**(**I**,** U**,**"rx"**)**

""" Modifier ci-dessous pour placer le texte aux coordonnées voulues et à la couleur voulue """

plt**.**text **(**0.0005**,**6**,**"Nuage des points experimentaux"**,**color**=**'r'**)**

""" TRAVAIL 4:

----------

Suivre les consignes écrites en rose ci-dessous: """

# #######################################

# Tracé de la courbe modélisée

# Variables permettant d'ajuster le modèle

a **=** 2000 # bon modèle: a = 2000

b **=** 2

# Modèle choisi:

""" Taper ci-dessous le code du modèle qui correspond le mieux à votre courbe """

Umod **=** a**\***I

""" TRAVAIL 5:

----------

Suivre les consignes écrites en rose ci-dessous: """

# #######################################

# Tracé de la courbe expérimentale

""" Taper ci-dessous le code permettant de représenter la courbe de modélisation Umod en fonction de I

en bleu, points non affichés mais reliés par un trait"""

plt**.**plot**(**I**,** Umod**,**"b-"**)**

""" Modifier ci-dessous pour placer le texte aux coordonnées voulues et à la couleur voulue """

plt**.**text **(**0.001**,**0.5**,**"Courbe modélisée U="**+**str**(**a**)+**".I"**,**color**=**'b'**)**

# ##################################

# Affichage de la fenêtre MATPLOTLIB

plt**.**show**()**