

BACCALAUREAT

SESSION 2022

Épreuve de l'enseignement de spécialité

NUMERIQUE et SCIENCES INFORMATIQUES

Partie pratique

Classe Terminale de la voie générale

Sujet n°18

DUREE DE L'ÉPREUVE : 1 heure

**Le sujet comporte 3 pages numérotées de 1 / 3 à 3 / 3
Dès que le sujet vous est remis, assurez-vous qu'il est complet.**

Le candidat doit traiter les 2 exercices.

EXERCICE 1 (4 points)

On a relevé les valeurs moyennes annuelles des températures à Paris pour la période allant de 2013 à 2019. Les résultats ont été récupérés sous la forme de deux listes : l'une pour les températures, l'autre pour les années :

```
t_moy = [14.9, 13.3, 13.1, 12.5, 13.0, 13.6, 13.7]
annees = [2013, 2014, 2015, 2016, 2017, 2018, 2019]
```

Écrire la fonction `mini` qui prend en paramètres le tableau `releve` des relevés et le tableau `date` des dates et qui renvoie la plus petite valeur relevée au cours de la période et l'année correspondante.

Exemple :

```
>>> mini(t_moy, annees)
12.5, 2016
```

EXERCICE 2 (4 points)

Un mot palindrome peut se lire de la même façon de gauche à droite ou de droite à gauche : *bob*, *radar*, et *non* sont des mots palindromes.

De même certains nombres sont eux aussi des palindromes : 33, 121, 345543.

L'objectif de cet exercice est d'obtenir un programme Python permettant de tester si un nombre est un nombre palindrome.

Pour remplir cette tâche, on vous demande de compléter le code des trois fonctions ci-dessous sachant que la fonction `est_nbre_palindrome` s'appuiera sur la fonction `est_palindrome` qui elle-même s'appuiera sur la fonction `inverse_chaine`.

La fonction `inverse_chaine` inverse l'ordre des caractères d'une chaîne de caractères `chaine` et renvoie la chaîne inversée.

La fonction `est_palindrome` teste si une chaîne de caractères `chaine` est un palindrome. Elle renvoie `True` si c'est le cas et `False` sinon. Cette fonction s'appuie sur la fonction précédente.

La fonction `est_nbre_palindrome` teste si un nombre `nbre` est un palindrome. Elle renvoie `True` si c'est le cas et `False` sinon. Cette fonction s'appuie sur la fonction précédente.

Compléter le code des trois fonctions ci-dessous.

```
def inverse_chaine(chaine):  
    result = ...  
    for caractere in chaine:  
        result = ...  
    return result  
  
def est_palindrome(chaine):  
    inverse = inverse_chaine(chaine)  
    return ...  
  
def est_nbre_palindrome(nbre):  
    chaine = ...  
    return est_palindrome(chaine)
```

Exemples :

```
>>> inverse_chaine('bac')  
'cab'  
>>> est_palindrome('NSI')  
False  
>>> est_palindrome('ISN-NSI')  
True  
>>> est_nbre_palindrome(214312)  
False  
>>> est_nbre_palindrome(213312)  
True
```

```

"""
Author: Pascal Padilla
Source: correction de l'exercice 1 du sujet 18 des épreuves pratiques NSI 2022
"""

from doctest import testmod

def mini(releve, date):
    """
    Fonction qui renvoie la plus petite valeur relevée dans le
    tableau `releve` et l'année correspondante issue du tableau `date`.

    Args:
        releve (List[float]): tableau de températures (flottants)
        date (List[int]): tableau de nombres entiers (les années)

    Returns:
        tuple: couple (valeur minimale, année associée)

    Tests:
    >>> t_moy = [14.9, 13.3, 13.1, 12.5, 13.0, 13.6, 13.7]
    >>> annees = [2013, 2014, 2015, 2016, 2017, 2018, 2019]
    >>> mini(t_moy, annees)
    (12.5, 2016)
    """
    # BOUCLE
    # invariant:
    #     ¿ `i_min`: est l'indice de la valeur minimale de la
    #     portion de tableau releve[0 .. i-1]
    # initialisation:
    #     ¿ `i_min`: est fixé à 0 (première case du tableau)
    #     ¿ `i`: indice courant de la boucle fixé à 1
    i_min = 0

    # condition d'arrêt: tout le tableau parcouru
    #     ¿ i == len(releve)
    nb_relevés = len(releve)

    for i in range(1, nb_relevés):
        # mise à jour de l'invariant si une valeur plus
        # petite est trouvée
        if releve[i] < releve[i_min]:
            i_min = i

    # POST boucle
    # i_min est l'indice de la valeur minimale
    releve_min = releve[i_min]
    date_min = date[i_min]
    return (releve_min, date_min)

# tests avec une assertion
t_moy = [14.9, 13.3, 13.1, 12.5, 13.0, 13.6, 13.7]
annees = [2013, 2014, 2015, 2016, 2017, 2018, 2019]
assert mini(t_moy, annees) == (12.5, 2016)

# tests avec des affichages
print(mini(t_moy, annees))

# tests avec doctest
testmod()

```

```

"""
Author: Pascal Padilla
Source: correction de l'exercice 2 du sujet 18 des épreuves pratiques NSI 2022
"""

def inverse_chaine(chaine):
    # initialisation avec une chaîne vide
    result = ''

    # boucle qui parcourt les symboles de la chaîne de
    # caractère un par un
    # `caractere` vaut le symbole courant
    for caractere in chaine:

        # concaténation du caractère courant au **début** de
        # la chaîne résultat qui sera renvoyée
        result = caractere + result

    return result

def est_palindrome(chaine):
    # inverse contient la chaîne inversée
    inverse = inverse_chaine(chaine)

    # tests booléen qui vaut:
    # Vrai ; la chaîne et son inverse sont égales
    #
    # la fonction renvoie le résultat du test booléen
    return chaine == inverse

def est_nbre_palindrome(nbre):
    # transforme le nbre en chaîne de caractère
    chaine = str(nbre)
    return est_palindrome(chaine)

# tests avec des assertions
assert inverse_chaine('bac') == 'cab'
assert est_palindrome('NSI') == False
assert est_palindrome('ISN-NSI') == True
assert est_nbre_palindrome(214312) == False
assert est_nbre_palindrome(213312) == True

# tests avec des affichages
print(inverse_chaine('bac'))
print(est_palindrome('NSI'))
print(est_palindrome('ISN-NSI'))
print(est_nbre_palindrome(214312))
print(est_nbre_palindrome(213312))

```