

```

from doctest import testmod

class Noeud:
    def __init__(self, g, v, d):
        self.gauche = g
        self.valeur = v
        self.droit = d

    def __str__(self):
        return str(self.valeur)

    def est_une_feuille(self):
        '''Renvoie True si et seulement si le noeud est une feuille'''
        return self.gauche is None and self.droit is None

def expression_infixe(e):
    """
    Exemples et tests :
    >>> e = Noeud(Noeud(Noeud(None, 3, None), '*', Noeud(Noeud(None, 8, None), '+',
    Noeud(None, 7, None))), '-', Noeud(Noeud(None, 2, None), '+', Noeud(None, 1, None)))
    >>> expression_infixe(e)
    '((3*(8+7))-(2+1))'
    """
    # initialisation de la chaîne à renvoyer
    s = ""
    if e.gauche is not None:
        # appel récursif pour afficher d'abord le sous-arbre gauche
        s = '(' + s + expression_infixe(e.gauche)

    # ajout de la valeur du noeud
    # en utilisant la méthode dédiée
    s = s + str(e.valeur)

    if e.droit is not None:
        # appel récursif pour afficher ensuite le sous-arbre droit
        # s'il est non vide
        s = s + expression_infixe(e.droit) + ')'

    # ATTENTION SUJET FAUX : IL FAUT COMMENTER LES DEUX DERNIÈRES
    # LIGNES ET AJOUTER UN RENVOIE FINAL
    # if ... :
    #     return s

    return s

testmod()

```