```python
# ds2.py

def inverser(mot):
    reponse = ''
    for lettre in mot:
        reponse = lettre+reponse
    return reponse

def est_palindrome(mot):
    return mot==inverser(mot)

def compter(lettre, mot):
    nb = 0
    for c in mot:
        if c==lettre:
            nb+=1
    return nb

def est_anagramme(mot1, mot2):
    for c1 in mot1:
        if compter(c1,mot2)!=compter(c1,mot1):
            return False
    return len(mot1)==len(mot2)

def melange(liste1, liste2):
    assert len(liste1)==len(liste2)
    l = []
    for k in range(len(liste1)):
        l.append(liste1[k])
        l.append(liste2[k])
    return l

def est_liste_entier(liste_nb):
    nb = 0
    for x in liste_nb:
        if type(x)==int:
            nb +=1
    return nb==len(liste_nb)

def moyenne(liste_nb):
    assert est_liste_entier(liste_nb)
    S = 0
    for x in liste_nb:
        S += x
    return S//len(liste_nb)

def decoupe(liste_nb):
    moy = moyenne(liste_nb)
    liste_petit, liste_grand = [], []
    for x in liste_nb:
        if x<=moy:
            liste_petit.append(x)
        else:
            liste_grand.append(x)
    return liste_petit, liste_grand

# La fonction exo9 tri une liste de nombres par dichotomie.
# On decoupe la liste entre les petits et les grands.
# Puis on place les petits devant et les grand triés récurssivement.
# La fonction s'applique si on il n'y a pas d'élements en double.
# On peut ajouter la condition
# for x liste_nb:
#    assert compter(x,liste_nb)==1
```