Ecriture décimale d'un nombre rationnel

Exercice

En partant de l'écriture irréductible d'un nombre rationnel donnée par deux nombres a et b (le numérateur et le dénominateur de la fraction), écrire une fonction qui donne la nature du nombre $\frac{a}{b}$ (entier, décimal non entier ou rationnel non décimal) et la suite des chiffres qui se répète dans l'écriture décimale d'un rationnel non décimal. Si on entre 1 et 7, la fonction doit renvoyer 142857. Si le nombre est décimal, la fonction donne son écriture décimale.

Il faut effectuer une division euclidienne de a par b jusqu'à reconnaître le reste dans la liste des restes déjà obtenus. Les restes sont mis dans une liste tant qu'ils n'y sont pas déjà. Pour détecter la séquence périodique, on doit juste déterminer le rang du reste qui a permis de détecter la répétition.

```
def decimal(a,b):
  quotient=str(a//b)  # la partie entiere du quotient
  reste=a%b
  texte=0
  if reste!=0:
    suiteRestes=[reste]
    quotient+=","
    while texte==0:
      quotient+=str(reste*10//b)
      reste=(reste*10)%b
      suiteRestes.append(reste)
      if reste==0:
        texte=1
        break
      for i in range(len(suiteRestes)-1):
        if reste==suiteRestes[i]:
          long=len(suiteRestes)-i-1
          sequence=str(quotient[-long:])
          quotient+=sequence+"..."
          print("Sequence periodique de longueur "+str(long)+" : "+sequence)
          texte=2
  return texte,quotient

num,denom=1,7
t,q=decimal(num,denom)
nature=["entier","decimal non entier","rationnel non decimal"]
```

```
27 print(num,"/",denom,"est un nombre "+nature[t],":",q)
```

La fonction `decimal` réalise cela et renvoie un quotient décimal contenant deux répétitions de la séquence périodique lorsque le rationnel n'est pas décimal. L'affichage de la séquence et de sa longueur est juste donné en bonus, pour éviter d'avoir à la retrouver ou à la compter. L'instruction `quotient[-|long|:]` permet de retrouver la séquence en découpant dans la chaîne de caractères quotient une tranche de longueur `long`, en commençant par la fin. Cette instruction utilise le fait que Python considère les chaînes de caractères comme des listes. Par exemple, si `t='bonjour'` alors `t[0]` vaut `'b'` et `t[-2:]` vaut `'ur'`.

Sur la calculatrice, une amélioration notable de l'affichage serait obtenue en coupant les grands nombres toutes les 25 ou 30 décimales (à ajuster) et en concevant des lignes plus courtes pour le texte accompagnateur. Pour la découpe en tranches d'un texte, la fonctionnalité `L[d:f]` de Python sera utilement mise à profit. Ici, pour le quotient `q`, on affichera `q[:25]` sur la ligne 1, puis `q[26,50]` sur la ligne 2, jusqu'à `q[x:]` où le rang `x` est à déterminer.

```
deg                PYTHON
>>> from decimal import *
Sequence periodique de longueu
1 / 7 est un nombre rationnel
>>> decimal(1,7)
Sequence periodique de longueu
(2, '0,142857142857...')
>>> decimal(8,16)
(1, '0,5')
>>> decimal(8,4)
(0, '2')
>>> |
```