

Informatique – TP14

Vésale Nicolas - Henrik Thys

Exercice 1: Correction d'une erreur trouvée lors d'un test.

On propose la fonction suivante, permettant de résoudre le problème du rendu de monnaie en utilisant une méthode gloutonne. Cette fonction prend en paramètre la somme à rendre (en euro) et renvoie une liste de pièces qu'il est possible de rendre.

```
def rendu(s):
    index = 0
    tab_piece = [2, 1, 0.5, 0.2, 0.1, 0.05, 0.02, 0.01]
    somme_rendue = 0
    pieces_rendues = []
    while somme_rendue != s :
        somme_a_rendre = s - somme_rendue
        if somme_a_rendre < tab_piece[index]:
            index = index + 1
        else :
            somme_rendue = somme_rendue + tab_piece[index]
            pieces_rendues.append(tab_piece[index])
    return pieces_rendues
```

Le test `rendu(0.3)` rend une erreur. Expliquez pourquoi. Proposez une modification pour résoudre ce problème.

Exercice 2: Un jeu de test comme aide à la programmation.

Dans cette exercice, nous allons créer une fonction `racinemax(a,b,c)` qui prend pour paramètres trois `int` et qui rend la plus grande racine réelle, si elle existe, du polynôme:

$$P(x) = ax^2 + bx + c$$

et une erreur sinon.

1. Proposer un jeu de tests pour la fonction `racinemax`. Celui-ci devra comprendre au moins 6 éléments.
2. Écrire la fonction `racinemax`. Évidemment, elle devra passer avec succès les tests de la question 1.

Exercice 3: Amélioration d'une fonction.

Dans le sujet CCPINP MP math 1, il était demandé de donner une fonction `flottantVersBase3(n,x)` qui prend pour paramètres un flottant `x` de $[0, 1[$ et un entier `n` et qui rend une liste composée des `n` premiers chiffres après la virgule de l'écriture en base 3 de `x`. Le code suivant a été proposé par un étudiant:

```
def flottantVersBase3(n,x):
    p,L=1, []
    for i in range(1,n+1):
        u=p*x #???
        t=int(3*u)-3*int(u)
        L.append(t)
        p=3*p
    return L
```

1. Calculer la valeur de `u` lors de la `i`-ième itération. Ajouter cette valeur en commentaire à la place de `???`.
2. Que donne la fonction si on appelle `flottantVersBase3(5,1)`? Modifier la fonction en spécifiant les types de `n` et `x` et utilisant des instructions `assert` pour interrompre le programme lorsque les pré-conditions sur `n` et `x` de l'énoncé ne sont pas remplies.
3. Les chiffres de l'écriture en base 3 étant toujours des entiers de $\{0, 1, 2\}$, la liste renvoyée doit être composée de tels entiers. Exécutez: `flottantVersBase3(50,1/3)`. Que constatez-vous? Comment expliquer ce comportement? Utilisez une instruction `assert` pour stopper le programme dès que le résultat ne vérifie plus cette condition.