

Imperative is not Iterative

1 Alternatives et fonctions

Exercice 1.1 (Maximum)

L'algorithme suivant lit trois valeurs entières, puis affiche le maximum des trois.

```
fonction max3(entier x, y, z) : entier
  variables
    entier    max
  debut
    si (x > y) et (x > z) alors
      max ← x
    sinon
      si (y > x) et (y > z) alors
        max ← y
      sinon
        si (z > x) et (z > y) alors
          max ← z
        fin si
      fin si
    fin si
  retourne max
fin

variables
  entier    a, b, c

debut      /* algorithme principal */
  lire (a)
  lire (b)
  lire (c)
  ecrire (max3 (a, b, c))
fin
```

Traduire cet algorithme en Python après l'avoir corrigé (éventuellement) et simplifié : il doit y avoir le moins de tests possibles.

Exercice 1.2 (Date du lendemain)

Une date est définie par trois entiers représentant l'année, le mois et le jour.

Écrire un script Python permettant d'obtenir, à partir d'une date donnée, la date du lendemain. Le script doit tester si la date donnée par l'utilisateur est correcte (ex : il n'existe pas de 30 février).

2 Fonctions récursives

Exercice 2.1 (Liste vers 9)

On donne un nombre entier positif AB à deux chiffres A et B tels que A différent de B .

Soit par exemple $AB = 19$.

— En inversant les chiffres on obtient 91.

— On effectue la différence entre 91 et 19 pour obtenir le nombre suivant : $91-19=72$.

On répète ce procédé avec 72 (on obtient $45 = 72-27$). Une dernière répétition de cette méthode donne $54-45=9$. La liste de nombres ainsi générée est appelée "liste vers 9" car elle s'arrête avec le nombre 9. Si les deux chiffres A et B sont égaux, la liste s'arrêtera avec 0.

Écrire un script Python qui affiche la liste vers 9 d'un entier donné.

Exercice 2.2 (Nombres Parfaits)

Un nombre est dit parfait s'il est plus grand que 1 et égal à la somme de tous ses diviseurs qui lui sont strictement inférieurs.

Exemple : $28 = 1 + 2 + 4 + 7 + 14$.

Écrire un script qui lit un entier, puis, s'il est valide, affiche tous ses diviseurs et détermine si celui-ci est parfait.

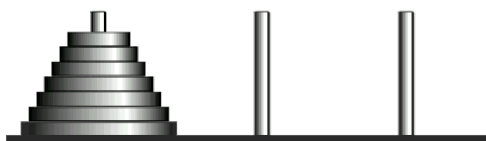
Exemple d'affichage :

```
1 give n
2 28
3 divisors are: 1, 2, 4, 7, 14,
4 28 is perfect.
```



Bonus : écrire une version optimisée qui affiche les diviseurs dans l'ordre.

Exercice 2.3 (Bonus : Tours de hanoi)



Le but du jeu est de bouger tous les disques du piquet "source" vers un nouveau piquet (la "destination") en se servant du piquet "intermédiaire" et en respectant les règles suivantes :

- On ne peut déplacer qu'un disque à la fois.
- On ne peut déplacer un disque qui se trouve sous un autre.
- Un disque ne peut être déposé sur un disque plus petit.

Écrire un script qui résout le problème des tours de Hanoi en affichant les déplacements effectués. Le script affichera de plus le nombre de déplacements.

Quel est le nombre de déplacements pour n disques ?