TD répétitif

Toutes les fonctions sont à écrire en Python

1 Itérations

Exercice 1.1 (Zorglub)

Que calcule la fonction suivante lorsqu'elle est appelée avec un entier n strictement positif ?

```
fonction zorglub(entier n) : entier
    variables
        entier    i, j, k
debut
    j ← 1
    k ← 0
    i ← 1
    tant que i <= n faire
        j ← i*j
        k ← j+k
        i ← i+1
    fin tant que
    retourne k
fin
```

Traduire cette fonction en Python.

Exercice 1.2 (Multiplication)

1. Écrire une fonction qui calcule $x \times y$, avec $(x, y) \in \mathbb{N}^2$ en n'utilisant que les opérateurs + et -.
2. Écrire une fonction qui calcule $x \times y$, avec cette fois-ci $(x, y) \in \mathbb{Z}^2$.

Exercice 1.3 (Puissance)

Écrire une fonction calculant le résultat de x^n, avec $x \in \mathbb{R}$ et $n \in \mathbb{N}$.

Exercice 1.4 (Fibonacci)

Écrire une fonction itérative calculant le $n^{ème}$ terme de la suite de Fibonacci.

$$fibo(0) = fibo(1) = 1$$
$$fibo(n) = fibo(n-1) + fibo(n-2)$$

Exercice 1.5 (Suite)

Soit une suite numérique u dont le $\mathrm{n}^{ème}$ terme peut être calculé par l'appel à la fonction `u(n)`.

1. Écrire une fonction qui retourne la somme S_n des n premiers termes de la suite u (de u_1 à u_n).
2. Sans utiliser la fonction précédente, écrire une fonction qui retourne

$$\sum_{i=1}^{n} S_i$$

   où S_i est la somme des i premiers termes de la suite u.

3. Si ce n'est pas déjà le cas, écrire de nouveau la fonction précédente avec une seule boucle !

2 Répétitions

Exercice 2.1 (Euclide)

Écrire une fonction calculant le pgcd de 2 entiers a et b strictement positifs en utilisant la méthode d'Euclide dont le principe est rappelé ci-dessous.

Algorithme d'Euclide :

Si a et b sont deux entiers avec $a \geq b$, si r est le reste de la division euclidienne de a par b : $a = bq + r$ avec $r < b$, alors le pgcd de a et b vaut le pgcd de b et r.
Si a est divisible par b, alors le pgcd de a et b est b.

Exercice 2.2 (Miroir)

Écrire une fonction qui donne le "miroir" d'un entier passé en paramètre s'il est positif.

Exemple : $1278 \rightarrow 8721$.

Remarque : si l'entier donné est 1250, le résultat sera 521.

Exercice 2.3 (Quotient)

Soient deux entiers naturels non nuls a et b. Écrire une fonction calculant le quotient entier de a sur b en n'utilisant que des opérateurs additifs (les seules opérateurs autorisés sont + et -).

Exercice 2.4 (Factorielle calculable ?)

Écrire une fonction qui, à partir d'une valeur entière donnée *limite*, calcule le plus grand nombre entier pair n tel que $n! < limite$. Si la valeur ne peut pas être calculée ($limite \leq 0$), la fonction retournera 0.

Par exemple : si $limite = 150$, $\quad 5! < 150 < 6!$
donc le plus grand entier pair dont la factorielle ne dépasse pas 150 est 4.

Exercice 2.5 (Où on cherche la puissance)

Écrire une fonction qui à partir de deux entiers naturels non nuls a et b, détermine si a est une puissance de b, c'est-à-dire $\exists p \in \mathbb{N} \;/\; a = b^p$. Une erreur devra être déclenchée en cas d'entrées non valides.

Exemples :

- si $a = 16$ et $b = 2$, alors la fonction retourne $vrai$.
- si $a = 15$ et $b = 5$, alors la fonction retourne $faux$.
- si $a = 81$ et $b = 3$, alors la fonction retourne $vrai$.

Exercice 2.6 (Nombre premier)

Écrire une fonction qui détermine si un entier strictement supérieur à 1 est premier ou non.

Exercice 2.7 (Bonus : Multiplication égyptienne)

Écrire une fonction qui calcule $x \times y$ en n'utilisant que des additions, des multiplications par 2 et des divisions par 2.

Indices :

— $10 \times 13 = 2 \times (5 \times 13)$ car 10 est pair.
— $11 \times 13 = 2 \times (5 \times 13) + 13$ car 11 est impair.