

1. Écrire les algorithmes permettant de calculer :
(a) une multiplication par additions successives.

```
Multiplier (a, b) {
  produit <- 0;
  Pour i <- 1 A a Faire
    produit <- produit + b;
  retourner produit;
}

Multiplier (a, b) {
  produit <- 0;
  Repeter
    produit <- produit + b;
    a <- a - 1;
  Jusqu'à a = 0
  retourner produit;
}
```

Problème posé par la version utilisant la boucle *Repeter* : cas $a = 0$.

- (b) une division par soustractions successives.

```
Diviser (a, b) {
  quotient <- 0;
  Tant que a >= b Faire {
    a <- a - b;
    quotient <- quotient + 1;
  }
  retourner quotient;
}
```

- (c) une élévation à la puissance par multiplications successives.

```
Puissance (a, b) {
  puiss <- 1;
  Pour i <- 1 A b Faire
    puiss <- puiss * a;
  retourner puiss;
}
```

- (d) le pgcd de deux nombres par soustractions successives.

$$\begin{aligned} \text{pgcd}(a, b) &= \text{pgcd}(a - b, b) && \text{si } a > b \\ \text{pgcd}(a, b) &= \text{pgcd}(a, b - a) && \text{si } b > a \\ \text{pgcd}(a, b) &= a && \text{si } a = b \end{aligned}$$

On suppose que les opérandes sont des entiers positifs.

```
PGCD (a, b) {
  Tant que a <> b Faire
    Si a > b Alors
      a <- a - b;
    Sinon
      b <- b - a;
  retourner a;
}
```

2. Calcul de $\sum_{i=1}^{i=n} i$ et de $\sum_{i=1}^{i=n} x^i$.

```
Somme_1_n (n) {  
  somme <- 0;  
  Pour i <- 1 A n Faire  
    somme <- somme + i;  
  retourner somme;  
}
```

```
Somme_puiss (x, n) {  
  somme <- 0;  
  puiss_x <- 1;  
  Pour i <- 1 A n Faire {  
    puiss_x <- puiss_x * x;  
    somme <- somme + puiss_x;  
  }  
  retourner somme;  
}
```