

III. Fonctions

1 Notion de fonction

Définition 1. On appelle fonction un algorithme qui renvoie une valeur. Elle peut posséder des paramètres dont les valeurs sont utilisées par l'algorithme.

Remarque 1. On appelle aussi parfois fonction (ou procédure) un algorithme qui ne renvoie aucune valeur.

Exemple 1. Fonction produit.

```

Fonction: produit( $m, n$ )
Action: Calcul du produit des entiers  $m$  et  $n$ 
Début
| Renvoyer  $m \times n$ 
Fin
    
```

Exercice 1. Écrire une fonction qui calcule la moyenne de deux nombres x et y .

Définition 2. On appelle variables locales des variables internes à une fonction.

Exemple 2. Fonction factorielle.

```

Fonction: factorielle( $n$ )
Action: Calcul de la factorielle  $f$  d'un entier  $n$ 
Début
|  $f \leftarrow 1$ 
| Pour  $k$  allant de 1 à  $n$  faire
| |  $f \leftarrow kf$ 
| FinPour
| Renvoyer  $f$ 
Fin
    
```

La variable f est utilisée uniquement lors de l'appel de la fonction factorielle et n'existe pas en dehors de celle-ci

Exercice 2. Écrire une fonction qui calcule le produit des entiers de m à n .

Définition 3. On appelle variables globales des variables externes aux différentes fonctions utilisées dans le programme.

Exemple 3. Fonction stockage.

```

Fonction: stockage( $x$ )
Action: stockage de la valeur de  $x$  dans la variable globale  $m$ 
Début
| Variable globale  $m$ 
|  $m \leftarrow x$ 
Fin
    
```

Lors de l'appel de la fonction stockage, la variable globale m prend pour valeur la valeur de x .

Remarque 2. On évite généralement de donner le même nom à une variable locale et une variable globale.

Exercice 3. Écrire une fonction qui échange les valeurs de deux variables globales x et y .

2 Algorithmique modulaire

Définition 4. *L'algorithmique modulaire consiste à décomposer un algorithme en plusieurs fonctions simples et réutilisables.*

Exemple 4. *Calcul de $\mathcal{Re}[(1 - 2i)^{17}]$.*

Fonction: $\text{prod}((a, b), (c, d))$

Action: Calcul du produit de deux nombres complexes $a+ib$ et $c+id$

Début

| Renvoyer $(ac - bd, ad + bc)$

Fin

Fonction: $\text{re}((a, b))$

Action: Calcul de la partie réelle d'un nombre complexe $a+ib$

Début

| Renvoyer a

Fin

$p \leftarrow (1, 0)$

Pour k allant de 1 à 17 faire

| $p \leftarrow \text{prod}(p, (1, -2))$

FinPour

Afficher $\text{re}(p)$

Exercice 4. *Créer une bibliothèque de fonctions permettant le calcul de la partie réelle et de la partie imaginaire d'un nombre complexe ainsi que de la somme, de la différence, du produit et du quotient de deux nombres complexes.*

Exercice 5. *Créer une bibliothèque de fonctions de géométrie plane permettant le calcul du produit scalaire de deux vecteurs, le calcul de la norme d'un vecteur ainsi que la caractérisation du type d'un triangle.*

Exercices supplémentaires

Exercice 6. *Écrire un algorithme permettant de calculer le terme d'indice n de la suite de Héron de paramètre a définie par*
$$\begin{cases} u_0 = 1 \\ u_{n+1} = \frac{1}{2} \left(u_n + \frac{a}{u_n} \right), \quad n \in \mathbb{N}^* \end{cases}$$

Exercice 7. *Écrire une fonction permettant de déterminer si un entier $n \geq 2$ est un nombre premier.*

Exercice 8. *Écrire une fonction permettant de déterminer le nombre de diviseurs d'un entier n non nul.*

Exercice 9. *On appelle nombre parfait¹ un entier égal à la moitié de la somme de ses diviseurs. Écrire une fonction permettant d'afficher les n premiers nombres parfaits.*

Exercice 10. *Écrire une fonction qui affiche la valeur d'une variable lors de ses 10 premières utilisations, puis affiche "vous avez atteint le nombre maximum d'utilisations" lors des utilisations suivantes.*

Exercice 11. *Créer une bibliothèque de fonctions permettant le calcul sur les fractions.*

1. Par exemple, 6 est parfait car il est divisible par 1, 2, 3 et 6 et $\frac{1+2+3+6}{2} = 6$

Réponses

- 1) **Fonction:** moyenne(x, y)
Action: Calcul de la moyenne des nombres x et y
Début
| Renvoyer $\frac{x+y}{2}$
Fin
- 2) **Fonction:** factoriellepartielle(m, n)
Action: Calcul du produit f des entiers de m à n
Début
| $f \leftarrow 1$
| **Pour k allant de m à n faire**
| | $f \leftarrow kf$
| **FinPour**
| Renvoyer f
Fin
- 3) **Fonction:** echange()
Action: échange des valeurs des variables globales x et y
Début
| **Variables globales x, y**
| $t \leftarrow x$
| $x \leftarrow y$
| $y \leftarrow t$
Fin
- 4) **Fonction:** re((a, b))
Action: Calcul de la partie réelle d'un nombre complexe $a + ib$
Début
| Renvoyer a
Fin
Fonction: im((a, b))
Action: Calcul de la partie imaginaire d'un nombre complexe $a + ib$
Début
| Renvoyer b
Fin
Fonction: somme($(a, b), (c, d)$)
Action: Calcul de la somme de deux nombres complexes $a + ib$ et $c + id$
Début
| Renvoyer $(a + c, b + d)$
Fin
Fonction: diff($(a, b), (c, d)$)
Action: Calcul de la différence de deux nombres complexes $a + ib$ et $c + id$
Début
| Renvoyer $somme((a, b), (-c, -d))$
Fin
Fonction: prod($(a, b), (c, d)$)
Action: Calcul du produit de deux nombres complexes $a + ib$ et $c + id$
Début
| Renvoyer $(ac - bd, ad + bc)$
Fin
Fonction: quo($(a, b), (c, d)$)
Action: Calcul du quotient de deux nombres complexes $a + ib$ et $c + id$
Début
| Renvoyer $prod(prod((a, b), (c, -d)), (\frac{1}{c^2+d^2}, 0))$
Fin

5)

```

Fonction: ps((x1, y1), (x2, y2))
Action: Calcul du produit scalaire des vecteurs  $\begin{pmatrix} x_1 \\ y_1 \end{pmatrix}$  et  $\begin{pmatrix} x_2 \\ y_2 \end{pmatrix}$ 
Début
    | Renvoyer  $x_1 \times x_2 + y_1 \times y_2$ 
Fin
Fonction: norme((x, y))
Action: Calcul de la norme du vecteur  $\begin{pmatrix} x \\ y \end{pmatrix}$ 
Début
    | Renvoyer  $\sqrt{\text{ps}((x, y), (x, y))}$ 
Fin
Fonction: longueurs((xA, yA), (xB, yB), (xC, yC))
Action: Calcul des trois longueurs associées aux points (xA, yA), (xB, yB) et (xC, yC)
Début
    | Renvoyer norme(xB - xA, yB - yA), norme(xC - xB, yC - yB), norme(xA - xC, yA - yC)
Fin
Fonction: plat(l1, l2, l3)
Action: Teste si les longueurs l1, l2 et l3 correspondent à des points alignés
Début
    | Si  $2 \times \max(l_1, l_2, l_3) = l_1 + l_2 + l_3$  alors
        | | Renvoyer Vrai
    | sinon
        | | Renvoyer Faux
    | FinSi
Fin
Fonction: equilateral(l1, l2, l3)
Action: Teste si les longueurs l1, l2 et l3 correspondent à un triangle équilatéral
Début
    | Si  $l_1 = l_2$  et  $l_2 = l_3$  alors
        | | Renvoyer Vrai
    | sinon
        | | Renvoyer Faux
    | FinSi
Fin
Fonction: isocèle(l1, l2, l3)
Action: Teste si les longueurs l1, l2 et l3 correspondent à un triangle isocèle
Début
    | Si  $l_1 = l_2$  ou  $l_2 = l_3$  ou  $l_3 = l_1$  alors
        | | Renvoyer Vrai
    | sinon
        | | Renvoyer Faux
    | FinSi
Fin
Fonction: rectangle(l1, l2, l3)
Action: Teste si les longueurs l1, l2 et l3 correspondent à un triangle rectangle
Début
    | Si  $2 \times \max(l_1, l_2, l_3)^2 = l_1^2 + l_2^2 + l_3^2$  alors
        | | Renvoyer Vrai
    | sinon
        | | Renvoyer Faux
    | FinSi
Fin
Fonction: triangle((xA, yA), (xB, yB), (xC, yC))
Action: Détermine la nature du triangle associé aux points (xA, yA), (xB, yB) et (xC, yC)
Début
    |  $l_1, l_2, l_3 = \text{longueurs}((x_A, y_A), (x_B, y_B), (x_C, y_C))$ 
    | Si plat(l1, l2, l3) alors
        | | Afficher "les points sont alignés"
    | sinon
        | | Si equilateral(l1, l2, l3) alors
            | | | Afficher "les points forment un triangle équilatéral"
        | | sinon
            | | | Si isocèle(l1, l2, l3) alors
                | | | | Si rectangle(l1, l2, l3) alors
                    | | | | | Afficher "les points forment un triangle isocèle rectangle"
                | | | | sinon
                    | | | | | Afficher "les points forment un triangle isocèle"
                | | | | FinSi
            | | | sinon
                | | | | Si rectangle(l1, l2, l3) alors
                    | | | | | Afficher "les points forment un triangle rectangle"
                | | | | sinon
                    | | | | | Afficher "les points forment un triangle quelconque"
                | | | | FinSi
            | | | FinSi
        | | FinSi
    | FinSi
Fin

```

6) **Fonction:** heron(a, n)
Action: Calcul du terme d'indice n de la suite de Héron de paramètre a
Début
 | $t \leftarrow 1$
 | **Pour** k allant de 1 à n faire
 | | $t \leftarrow \frac{1}{2} \left(t + \frac{a}{t} \right)$
 | **FinPour**
 | Renvoyer t
Fin

7) **Fonction:** testdeprimalite(n)
Action: test de la primalité de l'entier $n \geq 2$
Début
 | $k \leftarrow 2$
 | **TantQue** k ne divise pas n faire
 | | $k \leftarrow k + 1$
 | **FinTantQue**
 | **Si** $k = n$ alors
 | | Renvoyer *Vrai*
 | **sinon**
 | | Renvoyer *Faux*
 | **FinSi**
Fin

8) **Fonction:** nombredediviseurs(n)
Action: Calcul du nombre d de diviseurs d'un entier n non nul
Début
 | $d \leftarrow 0$
 | **Pour** k allant de 1 à n faire
 | | **Si** k divise n alors
 | | | $d \leftarrow d + 1$
 | | **FinSi**
 | **FinPour**
 | Renvoyer d
Fin

9) **Fonction:** sommedesdiviseurs(n)
Action: Calcul de la somme s des diviseurs d'un entier n non nul
Début
 | $s \leftarrow 0$
 | **Pour** k allant de 1 à n faire
 | | **Si** k divise n alors
 | | | $s \leftarrow s + k$
 | | **FinSi**
 | **FinPour**
 | Renvoyer s
Fin
Fonction: nombresparfaits(n)
Action: Affichage des n premiers nombres parfaits
Début
 | $s \leftarrow 0$
 | $k \leftarrow 0$
 | **TantQue** $s < n$ faire
 | | $k \leftarrow k + 1$
 | | **Si** $\text{sommedesdiviseurs}(k) = 2k$ alors
 | | | Afficher k
 | | | $s \leftarrow s + 1$
 | | **FinSi**
 | **FinTantQue**
Fin

10)

```

u ← 0
Fonction: affichage(x)
Action: Affichage de la valeur de la variable x avec 10 utilisations maximum
Début
| Variable globale u
| Si u < 10 alors
| | Afficher x
| | u ← u + 1
| sinon
| | Afficher "vous avez atteint le nombre maximum d'utilisations"
| FinSi
Fin
    
```

11)

```

Fonction: pgcd((a, b))
Action: Calcul du plus grand diviseur commun des entiers a et b
Début
| m ← a
| n ← b
| TantQue m ≠ n faire
| | Si m < n alors
| | | n ← n - m
| | sinon
| | | m ← m - n
| | FinSi
| FinTantQue
| Renvoyer m
Fin
Fonction: simpl((a, b))
Action: Simplification de la fraction a/b
Début
| Renvoyer ( $\frac{a}{pgcd(|a|,|b|)}$ ,  $\frac{b}{pgcd(|a|,|b|)}$ )
Fin
Fonction: somme((a, b), (c, d))
Action: Calcul de la somme de deux fractions a/b et c/d
Début
| Renvoyer simplification((ad + bc, bd))
Fin
Fonction: diff((a, b), (c, d))
Action: Calcul de la différence de deux fractions a/b et c/d
Début
| Renvoyer somme((a, b), (-c, d))
Fin
Fonction: prod((a, b), (c, d))
Action: Calcul du produit de deux fractions a/b et c/d
Début
| Renvoyer simplification((ac, bd))
Fin
Fonction: quo((a, b), (c, d))
Action: Calcul du quotient de deux fractions a/b et c/d
Début
| Renvoyer prod((a, b), (d, c))
Fin
    
```