

II. Boucles

1 Boucle *pour*

Définition 1. Une boucle *pour* est une structure de contrôle permettant de répéter une ou plusieurs instructions un nombre de fois déterminé, elle dispose d'un compteur d'itérations.

Exemple 1. Calcul de 2^n .

```

Entrée: variable entière  $n$ 
Sortie: variable entière  $p$  dont la valeur est égale à  $2^n$ 
Début
|  $p \leftarrow 1$ 
| Pour  $k$  allant de 1 à  $n$  faire
| |  $p \leftarrow 2p$ 
| FinPour
Fin
    
```

	valeur de p
<i>Début</i>	1
$k = 1$	2^1
$k = 2$	2^2
\vdots	
$k = n$	2^n
<i>Fin</i>	2^n

Exercice 1. Écrire un algorithme permettant d'afficher les n premiers entiers impairs et décrire son fonctionnement au moyen d'un tableau indiquant les affichages ainsi que l'évolution des valeurs des variables au fil des itérations.

Le compteur d'itérations peut être utilisé dans l'instruction itérée.

Exemple 2. Calcul de $1 + 2 + \dots + n$.

```

Entrée: variable entière  $n$ 
Sortie: variable entière  $s$  dont la valeur est égale à  $1 + 2 + \dots + n$ 
Début
|  $s \leftarrow 0$ 
| Pour  $k$  allant de 1 à  $n$  faire
| |  $s \leftarrow s + k$ 
| FinPour
Fin
    
```

	valeur de s
<i>Début</i>	0
$k = 1$	1
$k = 2$	$1 + 2$
\vdots	
$k = n$	$1 + 2 + \dots + n$
<i>Fin</i>	$1 + 2 + \dots + n$

Exercice 2. Écrire un algorithme permettant de calculer $n! = 1 \times 2 \times 3 \times \dots \times n$ pour $n \in \mathbb{N}$ et décrire son fonctionnement au moyen d'un tableau indiquant l'évolution des valeurs des variables au fil des itérations.

2 Boucle *tant que*

Définition 2. Une boucle *tant que* est une structure de contrôle permettant de répéter une ou plusieurs instructions tant qu'une condition donnée est réalisée.

Exemple 3. Calcul de la plus petite puissance de deux supérieure ou égale à un entier n .

Entrée: variable entière n

Sortie: variable entière p dont la valeur est égale à la plus petite puissance de deux supérieure ou égale à n

Début

| $p \leftarrow 1$

| **TantQue** $p < n$ faire

| | $p \leftarrow 2p$

| **FinTantQue**

Fin

Exercice 3. Qu'effectuent les algorithmes suivants ?

Début

| $k \leftarrow 0$

| **TantQue** $k > 0$ faire

| | **Afficher** k

| | $k \leftarrow k + 1$

| **FinTantQue**

Fin

Début

| $k \leftarrow 1$

| **TantQue** $k > 0$ faire

| | **Afficher** k

| | $k \leftarrow k + 1$

| **FinTantQue**

Fin

Exercice 4. Écrire un algorithme permettant d'afficher les multiples de 7 inférieurs ou égaux à un entier n donné.

Exercice 5. Écrire un algorithme permettant d'afficher les carrés inférieurs ou égaux à un entier n donné.

On utilise en général une boucle *tant que* lorsque le nombre d'itérations à effectuer n'est pas connu, une boucle *tant que* peut cependant remplacer une boucle *pour* en introduisant un *compteur*.

Exemple 4. Calcul de 2^n .

Entrée: variable entière n

Sortie: variable entière p dont la valeur est égale à 2^n

Début

| $p \leftarrow 1$

| $k \leftarrow 1$

| **TantQue** $k \leq n$ faire

| | $p \leftarrow 2p$

| | $k \leftarrow k + 1$

| **FinTantQue**

Fin

Exercice 6. Écrire un algorithme permettant de calculer $n!$ pour $n \in \mathbb{N}$ en utilisant une boucle *tant que*.

Exercices supplémentaires

Exercice 7. *Écrire un algorithme permettant d'afficher les entiers de 1 à n dans l'ordre décroissant, décrire son fonctionnement au moyen d'un tableau indiquant les affichages ainsi que l'évolution des valeurs des variables au fil des itérations.*

Exercice 8. *Écrire un algorithme permettant de calculer la somme des n premiers entiers impairs, décrire son fonctionnement au moyen d'un tableau indiquant l'évolution des valeurs des variables au fil des itérations.*

Exercice 9. *Écrire un algorithme permettant de calculer le terme d'indice n de la suite du traicteur paresseux¹ définie par $\begin{cases} u_0 = 1 \\ u_{n+1} = u_n + n + 1, n \in \mathbb{N}^* \end{cases}$, décrire son fonctionnement au moyen d'un tableau indiquant l'évolution des valeurs des variables au fil des itérations.*

Exercice 10. *Écrire un algorithme permettant de calculer le n -ième terme de la suite de Fibonacci définie par $\begin{cases} u_1 = 1 \\ u_2 = 1 \\ u_{n+2} = u_{n+1} + u_n, n \in \mathbb{N}^* \end{cases}$, décrire son fonctionnement au moyen d'un tableau indiquant l'évolution des valeurs des variables au fil des itérations.*

Exercice 11. *Écrire un algorithme permettant de calculer le nombre d'années nécessaire pour doubler un capital placé à intérêts composés avec un taux annuel de $t\%$.*

Exercice 12. *Écrire un algorithme permettant de déterminer combien de fois un entier n non nul donné est divisible par 2.*

Exercice 13. *Écrire un algorithme permettant de calculer le plus grand diviseur commun de deux entiers m et n au moyen de soustractions successives.*

1. On peut obtenir au maximum u_n parts dans un gâteau au moyen de n coupes rectilignes d'un bord à l'autre.

Réponses

1) **Entrée:** variable entière n
Sortie: affichage des n premiers entiers impairs
Début
 $i \leftarrow 1$
Pour k allant de 1 à n faire
 Afficher i
 $i \leftarrow i + 2$
FinPour
Fin

	affichages	valeur de i
Début		1
$k = 1$	1	3
$k = 2$	3	5
\vdots		
$k = n$		
Fin		

2) **Entrée:** variable entière n
Sortie: variable entière p dont la valeur est égale à $n!$
Début
 $p \leftarrow 1$
Pour k allant de 1 à n faire
 $p \leftarrow p \times k$
FinPour
Fin

	valeur de p
Début	1
$k = 1$	1×1
$k = 2$	$1 \times 1 \times 2$
\vdots	
$k = n$	$1 \times 1 \times 2 \times \dots \times n$
Fin	$n!$

3) Le premier algorithme ne fait rien car la condition de la boucle *tant que* n'est jamais réalisée, le second affiche les entiers strictement positifs (boucle infinie).

4) **Entrée:** variable entière n
Sortie: affichage des multiples de 7 inférieurs ou égaux à n
Début
 $m \leftarrow 0$
TantQue $m \leq n$ faire
 Afficher m
 $m \leftarrow m + 7$
FinTantQue
Fin

5) **Entrée:** variable entière n
Sortie: affichage des carrés inférieurs ou égaux à n
Début
 $k \leftarrow 0$
TantQue $k^2 \leq n$ faire
 Afficher k^2
 $k \leftarrow k + 1$
FinTantQue
Fin

6) **Entrée:** variable entière n
Sortie: variable entière p dont la valeur est égale à $n!$
Début
 $p \leftarrow 1$
 $k \leftarrow 1$
TantQue $k \leq n$ faire
 $p \leftarrow p \times k$
 $k \leftarrow k + 1$
FinTantQue
Fin

7) **Entrée:** variable entière n
Sortie: affichage des entiers de 1 à n dans l'ordre décroissant

```

Début
|  $i \leftarrow n$ 
| Pour  $k$  allant de 1 à  $n$  faire
| | Afficher  $i$ 
| |  $i \leftarrow i - 1$ 
| FinPour
Fin
    
```

	affichages	valeur de i
Début		n
$k = 1$	n	$n - 1$
$k = 2$	$n - 1$	$n - 2$
\vdots		
$k = n$		
Fin		

8) **Entrée:** variable entière n
Sortie: variable entière s dont la valeur est égale à la somme des n premiers entiers impairs

```

Début
|  $s \leftarrow 0$ 
|  $i \leftarrow 1$ 
| Pour  $k$  allant de 1 à  $n$  faire
| |  $s \leftarrow s + i$ 
| |  $i \leftarrow i + 2$ 
| FinPour
Fin
    
```

	valeur de s	valeur de i
Début	0	1
$k = 1$	1	3
$k = 2$	1 + 3	5
\vdots		
$k = n$		
Fin		

9) **Entrée:** variable entière n
Sortie: calcul du terme d'indice n de la suite du traiteur paresseux

```

Début
|  $a \leftarrow 1$ 
| Pour  $k$  allant de 1 à  $n$  faire
| |  $a \leftarrow a + k$ 
| FinPour
| Afficher  $a$ 
Fin
    
```

	valeur de a
Début	1
$k = 1$	1 + 1 = 2
$k = 2$	2 + 2 = 4
$k = 3$	4 + 3 = 7
$k = 4$	7 + 4 = 11
\vdots	
$k = n$	
Fin	

10) **Entrée:** variable entière n
Sortie: calcul du n -ième terme de la suite de Fibonacci

```

Début
|  $a \leftarrow 1$ 
|  $b \leftarrow 1$ 
| Pour  $k$  allant de 1 à  $n - 1$  faire
| |  $t \leftarrow a$ 
| |  $a \leftarrow b$ 
| |  $b \leftarrow t + b$ 
| FinPour
| Afficher  $a$ 
Fin
    
```

	valeur de t	valeur de a	valeur de b
Début		1	1
$k = 1$	1	1	2
$k = 2$	1	2	3
$k = 3$	2	3	5
$k = 4$	3	5	8
\vdots			
$k = n$			
Fin			

11)

Entrée: variable réelle t
Sortie: variable entière n dont la valeur est égale au nombre d'années nécessaire pour doubler un capital placé à intérêts composés avec un taux annuel de $t\%$.

Début
 $c \leftarrow 1$
 $n \leftarrow 0$
TantQue $c < 2$ **faire**
 $c \leftarrow c + c \times \frac{t}{100}$
 $n \leftarrow n + 1$
FinTantQue
Afficher n
Fin

12)

Entrée: variable entière n non nulle
Sortie: variable entière i dont la valeur est égale au nombre de fois que l'on peut diviser n par 2

Début
 $k \leftarrow n$
 $i \leftarrow 0$
TantQue k *pair* **faire**
 $k \leftarrow \frac{k}{2}$
 $i \leftarrow i + 1$
FinTantQue
Afficher i
Fin

13)

Entrée: variables entières m et n
Sortie: calcul du plus grand diviseur commun de m et n

Début
 $a \leftarrow m$
 $b \leftarrow n$
TantQue $a \neq b$ **faire**
 Si $a < b$ **alors**
 $b \leftarrow b - a$
 sinon
 $a \leftarrow a - b$
 FinSi
FinTantQue
Afficher a
Fin