

Elements d'Algorithmique

CMTD4 : Fonctions Récursives

Fonctions Récursives

Répétition d'actions

Si on veut exécuter un certain de nombre de fois la même séquence d'actions :

- La boucle For
- La boucle Tant Que
- Les fonctions récursives

Factorielle

$$\text{factorielle}(n) = n! = n \times (n - 1) \times \dots \times 2 \times 1$$

Entrée : entier n

```
1: fonction FACTORIELLE( $n$ )  
2:    $f = 1$   
3:   pour  $i \leftarrow 2$  à  $n$  faire  
4:      $f = f \times i$   
5:   retourne  $f$ 
```

Factorielle

$$\text{factorielle}(n) = n! = n \times (n - 1) \times \dots \times 2 \times 1$$

Entrée : entier n

```
1: fonction FACTORIELLE( $n$ )  
2:    $f = 1$   
3:   pour  $i \leftarrow 2$  à  $n$  faire  
4:      $f = f \times i$   
5:   retourne  $f$ 
```

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

$\log_2(n)$ = Plus grand entier k tel que $2^k \leq n$

Entrée : entier n

1: **fonction** LOGARITHME(n)

2: $l = 0$

3: **tant que** $n \geq 1$ **faire**

4: $n = n/2$

5: $l = l + 1$

6: **retourne** l

$\log_2(n)$ = Plus grand entier k tel que $2^k \leq n$

Entrée : entier n

```
1: fonction LOGARITHME( $n$ )  
2:    $l = 0$   
3:   tant que  $n \geq 1$  faire  
4:      $n = n/2$   
5:      $l = l + 1$   
6:   retourne  $l$ 
```

Entrée : entier n

```
1: fonction LOGREC( $n$ )  
2:   si  $n \leq 1$  alors  
3:     retourne 0  
4:   sinon  
5:     retourne  $1 + \text{LOGREC}(n/2)$ 
```

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(4) $f = 4 \times ?$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(3)	
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(2)	
factRec(3)	$f = 3 \times \searrow$
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(1)	
factRec(2)	$f = 2 \times \searrow$
factRec(3)	$f = 3 \times \searrow$
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(0)
factRec(1) $f = 1 \times \searrow$
factRec(2) $f = 2 \times \searrow$
factRec(3) $f = 3 \times \searrow$
factRec(4) $f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(0)	$f = 1$
factRec(1)	$f = 1 \times \searrow$
factRec(2)	$f = 2 \times \searrow$
factRec(3)	$f = 3 \times \searrow$
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(1)	$f = 1 \times 1$
factRec(2)	$f = 2 \times \searrow$
factRec(3)	$f = 3 \times \searrow$
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(2)	$f = 2 \times 1$
factRec(3)	$f = 3 \times \searrow$
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(3)	$f = 3 \times 2$
factRec(4)	$f = 4 \times \searrow$

Pile d'exécution

Entrée : entier n

```
1: fonction FACTREC( $n$ )  
2:   si  $n = 0$  alors  
3:     retourne 1  
4:   sinon  
5:     retourne  $n \times \text{FACTREC}(n - 1)$ 
```

Pile d'exécution :

factRec(4) $f = 4 \times 6$

Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

1: **fonction** FACTREC(n)

2: **si** $n = 0$ **alors**

3: **retourne** 0

4: **sinon**

5: **retourne** $n \times \text{FACTREC}(n - 1)$

Entrée : entier n

1: **fonction** FACTINTER(n, f)

2: **si** $n = 0$ **alors**

3: **retourne** f

4: **sinon**

5: **retourne** FACTINTER($n - 1, n \times f$)

6: **fonction** FACTRECTER(n)

7: **retourne** FACTINTER($n, 1$)

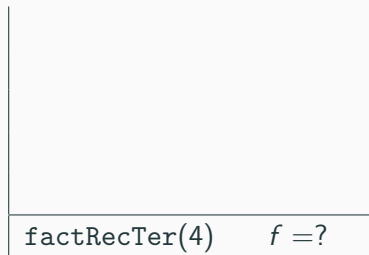
Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

```
1: fonction FACTINTER( $n, f$ )  
2:   si  $n = 0$  alors  
3:     retourne  $f$   
4:   sinon  
5:     retourne FACTINTER( $n - 1, n \times f$ )  
6: fonction FACTRECTER( $n$ )  
7:   retourne FACTINTER( $n, 1$ )
```

Pile d'exécution :



Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

```
1: fonction FACTINTER( $n, f$ )
```

2: **si** $n = 0$ **alors**

```
3:     retourne f
```

4: sinon

```
5:   retourne FACTINTER( $n - 1, n \times f$ )
```

6: **fonction** FACTRECTER(n)

```

7:   retourne FACTINTER( $n, 1$ )

```

Pile d'exécution :

factInter(3, 4)	
factRecTer(4)	$f = \searrow$

 $f = \searrow$

Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

```
1: fonction FACTINTER( $n, f$ )  
2:   si  $n = 0$  alors  
3:     retourne  $f$   
4:   sinon  
5:     retourne FACTINTER( $n - 1, n \times f$ )  
6: fonction FACTRECTER( $n$ )  
7:   retourne FACTINTER( $n, 1$ )
```

Pile d'exécution :

factInter(2, 12)	
factRecTer(4)	$f = \searrow$

Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

```
1: fonction FACTINTER( $n, f$ )  
2:   si  $n = 0$  alors  
3:     retourne  $f$   
4:   sinon  
5:     retourne FACTINTER( $n - 1, n \times f$ )  
6: fonction FACTRECTER( $n$ )  
7:   retourne FACTINTER( $n, 1$ )
```

Pile d'exécution :

factInter(1, 24)	
factRecTer(4)	$f = \searrow$

Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

```
1: fonction FACTINTER( $n, f$ )
```

2: **si** $n = 0$ **alors**

```
3:     retourne f
```

4: sinon

```
5:   retourne FACTINTER( $n - 1, n \times f$ )
```

6: **fonction** FACTRECTER(n)

```

7:   retourne FACTINTER( $n, 1$ )

```

Pile d'exécution :

factInter(0, 24)	
factRecTer(4)	$f = \searrow$

Fonction Récursive Terminale

Fonction récursive **terminale** : La dernière instruction est un appel récursif.

Entrée : entier n

```
1: fonction FACTINTER( $n, f$ )  
2:   si  $n = 0$  alors  
3:     retourne  $f$   
4:   sinon  
5:     retourne FACTINTER( $n - 1, n \times f$ )  
6: fonction FACTRECTER( $n$ )  
7:   retourne FACTINTER( $n, 1$ )
```

Pile d'exécution :

factRecTer(4)	$f = 24$