

Autour de la somme [rc02] - Exercice

Karine Zampieri, Stéphane Rivière

Unisciel  algoprogram  Version 21 mai 2018

Table des matières

1	Autour de la somme / pgsomme	2
1.1	Somme récursive	2
1.2	Somme récursive des pairs	3

C - Autour de la somme (TP)



Mots-Clés Récursivité des actions ■

Requis Algorithmes paramétrés ■

Difficulté ●○○ (20 min) ■



Objectif

Cet exercice calcule des sommes récursivement.

1 Autour de la somme / pgsomme

1.1 Somme récursive



Écrivez une fonction récursive `somme(n)` qui calcule et renvoie la somme des entiers compris entre 1 et `n` (entier).



Explicitez le calcul de `somme(3)`.



Écrivez une fonction récursive `sommetr(n)` qui « trace » le calcul de la somme des entiers compris entre 1 et `n` (entier). Voici le résultat d'exécution pour `\lstinline=3@`.

```
--> sommetr(3)
--> sommetr(2)
--> sommetr(1)
--> sommetr(0)
<-- sommetr(0)=0
<-- sommetr(1)=1
<-- sommetr(2)=3
<-- sommetr(3)=6
==> sommetr(3) vaut 6
```

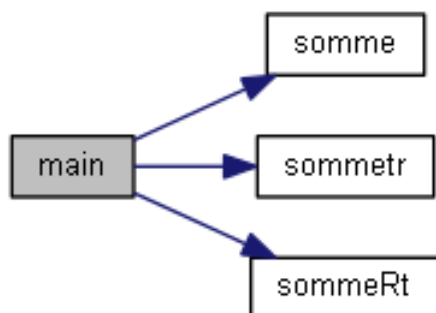
Les appels à la fonction `somme` sont repérés par le signe `-->` et les valeurs retournées sont repérées par le signe `<--`.



Écrivez une fonction récursive terminale `sommeRt(n,a)` qui calcule et renvoie la somme des entiers compris entre 1 et `n` (entier), l'entier `a` étant le résultat.



Écrivez une procédure `test1` qui saisit un entier puis appelle et affiche le résultat des diverses fonctions.



Testez. Exemple d'exécution :

```
Somme de 1 à ? 3
==> somme(3) vaut 6
--> sommetr(3)
```

```
--> sommetr(2)
--> sommetr(1)
--> sommetr(0)
<-- sommetr(0)=0
<-- sommetr(1)=1
<-- sommetr(2)=3
<-- sommetr(3)=6
==> sommetr(3) vaut 6
==> sommeRt(3) vaut 6
```

1.2 Somme récursive des pairs



Écrivez une fonction `sommePairs(x,n)` qui réalise la somme des `n` nombres naturels pairs consécutifs, à partir d'une valeur paire donnée `x` (la valeur de départ sera comprise dans la somme).



Écrivez une procédure `test2` qui saisit deux entiers puis appelle et affiche le résultat de la fonction.



Testez.