

EXERCICE 43

Analyser une situation et écrire un programme



On empile des sphères, formant ainsi une pyramide de base carrée.

1. Déterminer le nombre de sphères nécessaires pour constituer une pyramide de deux « niveaux ».

Il faut cinq sphères : quatre au 1^{er} niveau et une au 2^e niveau.

2. Déterminer le nombre de sphères nécessaires pour constituer une pyramide de cinq « niveaux ».

Il faut 55 sphères : une au 5^e niveau, 4 au 4^e, 9 au 3^e, 16 au 2^e et 25 au 1^{er}.

3. Sachant qu'il faut 140 sphères pour construire une pyramide de sept « niveaux », déterminer le nombre de sphères nécessaires pour construire une pyramide de huit « niveaux ».

$140 + 8^2 = 204$, donc il faut 204 sphères.

4. En utilisant la structure ci-dessous, compléter le programme de la fonction `nb_sphères` d'argument un entier naturel non nul N et qui retourne le nombre B de sphères nécessaires pour constituer une pyramide comportant N « niveaux ».

```
def nb_sphères (...N...):  
    B:=0.....  
    for i in range (1,N+1):  
        B=B+i*2.....  
    return(B).....
```

EXERCICE 42

Comprendre et programmer un algorithme

Soit n un entier naturel non nul. On pose $S = 1 + 2 + \dots + n$.

1. Calculer les sommes $1 + 2$; $1 + 2 + 3$ et $1 + 2 + 3 + 4$, puis calculer S lorsque $n = 5$.

$1 + 2 = 3$; $1 + 2 + 3 = 6$; $1 + 2 + 3 + 4 = 10$; $1 + 2 + 3 + 4 + 5 = 15$.

2. On considère l'algorithme ci-dessous.

```
S ← 0  
Pour i variant de 1 à n  
    S ← S + i  
Fin Pour
```

Exprimer en fonction de n , la valeur que contient la variable S en fin d'algorithme.

La valeur que contient la variable S est celle de la somme $1 + 2 + 3 + 4 + \dots + n$.

3. Programmer une fonction d'argument n qui retourne la valeur de la variable S de l'algorithme précédent. En déduire la somme des 1 000 premiers entiers naturels non nuls.

```
def somme(n):  
    S:=0.....  
    for i in range(1,n+1):  
        S=S+i.....  
    return(S).....
```

La somme des 1 000 premiers entiers non nuls est :

$1 + 2 + 3 + \dots + 1000 = 500\,500$.