Généralités sur les suites numériques

> Connaître la valeur d'un terme d'une suite

Exemple 1 : suite définie par une formule explicite

Soit (u_n) la suite définie sur $\mathbb{N}$ par $u_n = 3n^2 - 6n + 2$.
Voici un programme Python qui permet de connaître le terme u_n de cette suite.

```
1 def suite(n):
2     u=3*pow(n,2)-6*n+2
3     return(u)
```

Ligne 1 : on donne le nom du programme. Ici, on l'appelle « suite ».
Ligne 2 : u est une variable réelle. Elle prend comme valeur l'expression de u_n donnée dans l'exemple.
Ligne 3 : le programme retourne le résultat.

Si on veut connaître u_{10}, on doit donc taper « suite(10) ». Et on trouve $u_{10} = 242$.

Exemple 2 : suite définie par une relation de récurrence

Soit (u_n) la suite définie sur $\mathbb{N}$ par $u_0 = -4$ et pour tout entier naturel n par $u_{n+1} = -0,25u_n + 8$.
Voici un programme Python qui permet de connaître le terme u_n de cette suite.

```
1 def reccurence(n):
2     u=-4
3     for i in range (1,n+1):
4         u=-0.25*u+8
5     return(u)
```

Ligne 1 : On donne le nom de programme. Ici, on l'appelle « Reccurence ».
Ligne 2 : u est une variable. On fait ici une initialisation avec $u_0 = -4$.
Ligne 3 : C'est le début d'une boucle qui s'arrête jusqu'à $n + 1$. Autrement dit, si on souhaite connaître le terme u_7 on doit saisir $n = 6$.
Ligne 4 : C'est ici qu'on calcule le terme suivant jusqu'à l'indice souhaité.
Ligne 5 : Le programme retourne le terme u_{n+1}.

Si on veut connaître u_{10}, on doit donc taper « reccurence(9) ». Et on trouve $u_{10} = 6,399990081787109$.

Exemple 3 : la suite de Fibonacci

La suite de Fibonacci (mathématicien italien né à Pise vers 1170) est la suite (u_n) définie sur $\mathbb{N}$ par $u_0 = 1$, $u_1 = 1$ et pour tout entier naturel $n \geqslant 2$ par $u_n = u_{n-1} + u_{n-2}$. Pour connaître un terme, il faut donc faire la somme des deux précédents.
Voici un programme qui permet de calculer le terme de rang $n \geqslant 2$ de cette suite.

```
1 def Fibonacci(n):
2     u0=1
3     u1=1
4     for i in range (1,n+1):
5         u=u0+u1
6         u0=u1
7         u1=u
8     return(u)
```

Ligne 2 et 3 : Au lieu de préciser le premier terme de cette suite définie par récurrence, on précise les deux premiers : u_0 et u_1.
Ligne 5 : On calcule le terme suivant de la suite.
Ligne 6 : Le terme u_{n-2} prend la valeur du terme u_{n-1}.
Ligne 7 : Le terme u_{n-1} prend la valeur du terme u_n. On peut ainsi recommencer l'étape de la ligne 5 avec les nouvelles valeurs de la suite.

Si on souhaite connaître le terme u_7 de la suite de Fibonacci, on doit saisir « Fibonacci(6) » et on trouve $u_7 = 21$.

Cette suite possède une particularité. Plus n devient grand, plus $\dfrac{u_{n+1}}{u_n}$ se rapproche du nombre d'or $\dfrac{1+\sqrt{5}}{2}$.
On peut demander au programme de nous donner des approximations de cette valeur en ajoutant la ligne 6 ci-dessous et en modifiant la valeur à retourner.

```
1 def Fibonacci(n):
2     u0=1
3     u1=1
4     for i in range (1,n+1):
5         u=u0+u1
6         f=u/u1
7         u0=u1
8         u1=u
9     return(f)
```

> Rechercher un seuil

Exemple 1 : Placer de l'argent sur un compte épargne

Jean-Kevin possède un livret A qui lui rapporte 2% par an. Il place 1 000€ sur ce compte. Il veut savoir au bout de combien d'année il aura 3 000€ sur ce livret.

On considère la suite (u_n) définie par u_0 =1 000 et pour tout entier naturel $n \geqslant 1$ par $u_n = 1,02u_n$ où u_n représente la somme d'argent sur le compte de jean-Kevin au bout de n année.

On va demander à Python de calculer les termes de la suite et dès que $u_n \geqslant 3000$, le programme nous retourne la valeur de n correspondante.

```
1 def argent(S):
2     u=1000
3     n=0
4     while u<3000:
5         u=1.02*u
6         n=n+1
7     return(n)
```

Ligne 2 : On donne la valeur de u_0, l'argent déposée.
Ligne 3 : On est à $n = 0$ année.
Ligne 4 : Tant que u_n est inférieure à la somme S d'argent voulue, on continue de calculer les u_n.
Ligne 5 : On calcule le terme suivant grâce à la formule de réccurence.
Ligne 6 : On ajoute une année.
Ligne 7 : Quand la condition est respectée, le programme retourne la valeur de n.

Pour savoir au bout de combien d'année Jean-Kevin aura dépassé les 3 000€ sur son compte épargne, on saisi « argent(3000) ». D'après le programme, ce sera au bout de 56 ans.

> Calcul de sommes de termes d'une suite

Exemple 1 : somme des entiers naturels

On souhaite ici calculer la somme $1 + 2 + 3 + 4 + ... + n$ où n est un entier donné par l'utilisateur.
On considère donc la suite (u_n) définie par $u_0 = 1$ et pour tout entier naturel $n \geqslant 1$ par $u_{n+1} = u_n + 1$.
On s'aide donc du programme de la première partie de ce document (exemple 2) et on ajoute la ligne 5 ainsi qu'une nouvelle variable S qui ajoute au fur et à mesure les termes de cette suite.

```
1 def somme(n):
2     u=1
3     S=0
4     for i in range (1,n):
5         S=S+u
6         u=u+1
7     return(S)
```

Si on veut connaître le résultat de $1 + 2 + ... + 10$ on doit donc saisir « somme(10) ». On trouve alors 55.