MATHÉMATIQUES POUR L'INFORMATIQUE 4
TD1

Exercices à résoudre sur papier

1. On considère la suite (u_n) définie par la récurrence

$$u_n = u_{n-1} + 2n - 1, \quad u_0 = 0.$$

a) Calculer u_n pour $n \leq 6$.
b) Deviner une formule pour u_n.
c) Prouver la formule par récurrence.
d) Exprimer u_n comme une somme (sans récurrence) et imaginer un dessin rendant la solution évidente.

2. Mêmes questions pour les sommes alternées

$$S_n = \sum_{k=1}^{n} (-1)^{n-k} k^2.$$

3. Soit (v_n) la suite définie par la récurrence

$$v_n = 3v_{n-1} - 1, \quad v_0 = 1.$$

a) Calculer les 6 premiers termes.
b) Trouver une expression de la série génératrice

$$V(x) = \sum_{n \geq 0} v_n x^n.$$

c) En déduire la valeur de v_n.

Exercices sur machine

Afficher la page `http://igm.univ-mlv.fr/~jyt/L3_MPI4/td1L3.html`

Le module Python `sympy` fait du calcul symbolique. On peut lui déclarer des variables au sens mathématique (plus précisément, des indéterminées), et les manipuler formellement :

```
>>> from sympy import *
>>> var('x y')
(x, y)
>>> expand((1+x)**3) # développement d'un produit
x**3 + 3*x**2 + 3*x + 1
>>> factor(1-4*x+3*x**2) # factorisation d'un polynôme
(x - 1)*(3*x - 1)
>>> R = 1/_; R # R est une fraction rationnelle
1/((x - 1)*(3*x - 1))
>>> apart(_) # décomposition en éléments simples
-3/(2*(3*x - 1)) + 1/(2*(x - 1))
>>> series(R,x,0,8).removeO() # il sait développer en série (removeO() enlève le O(x**8))
3280*x**7 + 1093*x**6 + 364*x**5 + 121*x**4 + 40*x**3 + 13*x**2 + 4*x + 1
>>> Rational(4,12) # il connaît les fractions
1/3
```

4. Soit (u_n) la suite définie par

$$u_n = 2u_{n-1} + u_{n-2} - 2u_{n-3}, \quad u_0 = 1,\ u_1 = 0, u_2 = 2.$$

a) Trouver une expression de la série génératrice $U(x)$ de (u_n).
b) Calculer son développement en série à l'ordre 12 avec `sympy`.
c) Programmer la fonction `u(n)` à partir de la récurrence et afficher les 12 premiers termes.
d) Recommencer jusqu'à ce que les questions b) et c) donnent les mêmes résultats.
e) Trouver une formule explicite pour u_n.
f) Programmer cette suite sous forme d'une fonction `u(n)`.
g) Étrangement, il se trouve que cette suite est connue. Interrogez l'encyclopédie en ligne des suites d'entiers (`http://oeis.org`) pour voir ce qu'on en sait.

5. On considère la suite (q_n) définie par

$$q_n = \frac{1 + q_{n-1}}{q_{n-2}}, \quad q_0 = a,\ q_1 = b.$$

Il n'existe pas de méthode générale pour résoudre ce genre de récurrence.
a) Programmer la suite avec `sympy`.
b) En étudiant les premiers termes, deviner la réponse.
c) Reste-t-il quelque chose à prouver ?

6. Soit A_n la somme alternéee

$$A_n = \sum_{k=0}^{n} (-1)^k \frac{(2k+1)^3}{(2k+1)^4 + 4}.$$

a) Programmer la fonction `A(n)` avec `sympy` (utiliser `Rational`).
b) Afficher les 12 premiers termes.
c) Deviner une formule pour le signe, le numérateur, et le dénominateur de A_n.
d) Prouver la formule par récurrence (`sympy` peut faire les calculs algébriques requis pour la vérification).