# Sommation formelle

On se donne une somme $S_n = \Sigma_{k=n_0}^{n} a_k$, existe-t-il une expression "fermée" de $S_n$, en fonction de n ?

## Exemples

- $\Sigma_{k=0}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$

- $\Sigma_{k=1}^{n} \frac{1}{k(k+1)} = \frac{n}{n+1}$

- $\Sigma_{k=1}^{n} \frac{n^3-2n^2-1}{n^4+n^2+1}(n-1)! = \frac{n!}{n^2+n+1}$

- $\Sigma_{k=1}^{n} \frac{1}{n!} = ?$

- $\Sigma_{k=1}^{n} \frac{2n+3}{n(n+1)} = ?$

## Formes fermées

On s'interesse aux sommes $S_n = \Sigma_{k=n_0}^{n} a_k$ telles que : $\frac{S_n}{S_{n-1}}$ soit une fraction rationnelle en n.

**Proposition 1.** *$\frac{S_n}{S_{n-1}}$ est une fraction rationelle en n ssi il existe deux fractions rationelles $R(X)$ et $\alpha(X)$ telles que :*

- $\forall n \in \mathbb{N},\ \frac{a_n}{a_{n-1}} = R(n)$

- $\forall n \in \mathbb{N},\ S_n = \alpha(n) a_n$

Si c'est le cas on dit que $S_n$ admet une forme fermée.

**Algorithme de Gosper** : Etant donné la fraction rationelle R(X) (determinée à l'aide de l'expression de $a_n$), l'algorithme doit determiner si la fraction rationelle $\alpha(X)$ existe et dans ce cas doit la calculer.

## Principe de l'algorithme

**Proposition 2.** *Il existe 3 polynômes p(X), q(X), et r(X) tels que :*

- $R(X) = \frac{p(X)q(X)}{p(X-1)r(X)}$
- $\forall k \in \mathbb{N},\ q(X) \wedge r(X+k) = 1$

**Preuve algorithmique** :
On définit séquentiellement des polynomes $p_j(X)$, $q_j(X)$ et $r_j(X)$ tels que : $R(X) = \frac{p_j(X)q_j(X)}{p_j(X-1)r_j(X)}$ :

- On définit : $p_0(X) = 1$ et $R(X) = \frac{q_0(X)}{r_0(X)}$ avec $q_0(X) \wedge r_0(X) = 1$
- Si $\forall k \in \mathbb{N}$, $q_j(X) \wedge r_j(X+k) = 1$ on s'arrête
- Si $\exists k \in \mathbb{N}$, $q_j(X) \wedge r_j(X+k) \neq 1$ :
  On pose $g(X)$ le pgcd de $q_j(X)$ et de $r_j(X+k)$ et :
  $q_{j+1}(X) = \frac{q_j(X)}{g(X)}$, $r_{j+1}(X) = \frac{r_j(X)}{g(X-k)}$ et
  $p_{j+1}(X) = p_j(X)g(X)g(X-1)...g(X-k+1)$
  On a $R(X) = \frac{p_{j+1}(X)q_{j+1}(X)}{p_{j+1}(X-1)r_{j+1}(X)}$ et $\deg q_{j+1}(X) < \deg q_j(X)$

**Problème** :
Comment tester algorithmiquement si : $\forall k \in \mathbb{N}$, $q(X) \wedge r(X+k) = 1$ ?
On introduit le **résultant** par rapport à la variable $X$ des polynômes $q(X)$ et $r(X+Y)$ :
$G(Y) = Res(q(X), r(X+Y))$
On teste ensuite si ce polynôme G admet des racines entières.

**Proposition 3.** *Si la fraction rationelle $\alpha(X)$ existe, alors elle est de la forme*

$$\alpha(X) = \frac{q(X+1)}{p(X)}f(X)$$

*avec f(X) polynôme vérifiant l'équation fonctionelle :*

$$p(X) = q(X+1)f(X) - f(X-1)r(X)$$

**Proposition 4.** *Majoration du degré de $f(X)$ :*
*On pose $s_+(X) = q(X+1) + r(X)$, $s_-(X) = q(X+1) - r(X)$*

- *Si $deg\ s_-(X) \neq deg\ s_+(X) - 1$ alors*

$$deg\ f(X) = deg\ p(X) - max(deg\ s_-(X),\ deg\ s_+(X) - 1)$$

- *Si $l = deg\ s_-(X) = deg\ s_+(X) - 1$*
*On écrit : $s_-(X) = u_l X^l + \ldots$ et $s_+(X) = v_{l+1} X^{l+1} + \ldots$*
*On pose $n_0 = -2\frac{u_l}{u_{l+1}}$*
*Alors $deg\ f \leq \begin{cases} deg\ p - l & si\ n_0 \notin \mathbb{N} \\ max(deg\ p - l, n_0) & sinon. \end{cases}$*

## Exemple : $\Sigma_{k=0}^{n} k^2$

- $a_n = n^2$, $\frac{a_n}{a_{n-1}} = \frac{n^2}{n^2-2n+1}$ donc $R(X) = \frac{X^2}{X^2-2X+1}$

- On écrit $R(X) = \frac{p(X)q(X)}{p(X-1)r(X)}$ : on trouve $p(X) = X^2$, $q(X) = r(X) = 1$.

- On cherche f vérifiant (*) : $p(X) = q(X+1)f(X) - f(X-1)r(X)$.

  La majoration donne f de degré au plus **3**.

- On cherche donc f sous la forme $f(X) = w_0 + w_1 X + w_2 X^2 + w_3 X^3$
  (*) s'écrit $X^2 = (w_1 - w_2 + w_3) + (2w_2 - 3w_3)X + 3w_3 X^2$ donc matriciellement :

$$\begin{pmatrix} 0 & 1 & -1 & 1 \\ 0 & 0 & 2 & -3 \\ 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

- On trouve alors $f(X) = 1/3X^3 + 1/2X^2 + 1/6X$.
  Donc $\alpha(X) = \frac{q(X+1)}{p(X)} f(X) = f(X)/X^2$
  Puis $S_n = n^2\alpha(n) = 1/3n^3 + 1/2n^2 + 1/6 = \frac{n(n+1)(2n+1)}{6}$

# Programmation de l'algorithme en Ocaml

Programmation difficile car il faut en particulier :

- Gérer les polynômes, les fractions rationelles sur $\mathbb{Q}$
- Gérer les matrices
- Calculer le résultant de deux polynômes de $(\mathbb{Q}[\mathbb{X}])[\mathbb{Y}]$
- Résoudre un système linéaire
- Trouver les racines entières d'un pôlynome

**Solution :** développer une librairie d'algèbre en Ocaml travaillant sur un corps quelconque.
On utilise les fonctionnalités de **Modules** d'Ocaml : module, module de type et foncteurs.

## Définition d'un corps en Ocaml

```
module type DRing = sig
  type elem
  val zero: elem
  val one : elem
  val add : elem -> elem -> elem
  val minus : elem -> elem
  val mult : elem -> elem -> elem
  val inv : elem -> elem
  val print : elem -> string
end
```

### Utilisation des foncteurs

On prend un module en paramètre pour créer un nouveau module :

```
  module Matrix = functor (C : DRing) ->
    struct
    type matrix = C.elem array array
      (* ... *)
    end
```

Librairie complète : 450 lignes de code.

## Implémentation de l'algorithme à l'aide de cette librairie

175 lignes de code

```
open Algebra
module C = DRing_Rat(* Les rationnels *)
module Q = Frac(DRing_Rat)(*Les fractions rat.*)
module Py = Poly(Q.DRing_F) (* Les fractions
  rationnelles a 2 variables*)

(* 1ière étape : on ecrit R sous la forme : *)
(* R(X) = p(X)q(X) / p(X-1)r(X) avec pour tout k € N
    : q(X) ^ r(X+k) = 1 *)
(* Il faut donc calculer les polynomes p, q, r *)

(* 2ième étape : Majoration du degré de f *)

(* 3ième étape calcul explicite de f(X) *)

let solve R = (* *)
let solve_q a_n k0 = (* *)
```

# Exemples

- $\Sigma_{k=0}^{n} k^2$

```
# solve_q ([((1, 1), 2)], [((1,1), 0)]) 0;;
p = 1x^2
q = 1
r = 1
majoration du degre : 3
f = 1/6 x + 1/2 x^2 + 1/3 x^3
Resultat : 1/6 x + 1/2 x^2 + 1/3 x^3
```

  On a donc $\Sigma_{k=0}^{n} k^2 = 1/6n + 1/2n^2 + 1/3n^3 = \frac{n(n+1)(2n+1)}{6}$

- $\Sigma_{k=0}^{n} \frac{1}{(k+2)(k+5)}$
  $(k+2)(k+5) = k^2 + 7k + 10$

```
# solve_q ([((1, 1), 0)], [((10,1),0); ((7,1),
  1); ((1,1), 2)]) 0;;
p = 432 + 252x + 36x^2
q = -1/6 + -1/6 x
r = -5/6 + -1/6 x
majoration du degre : 3
f = 3384 + 1728x + 216x^2
Resultat : ( 6 + 323/36 x + 10/3 x^2 + 13/36
  x^3)/( 60 + 47x + 12x^2 + 1x^3)
```

  On a donc $\Sigma_{k=0}^{n} \frac{1}{(k+2)(k+5)} = \frac{216+323n+120n^2+13n^3}{36\left(60+47n+12n^2+n^3\right)}$

- $\Sigma_{k=1}^{n}\frac{n^3-2n^2-1}{n^4+n^2+1}(n-1)!$
  $\frac{a_n}{a_{n-1}}=\frac{3-6n+10n^2-16n^3+14n^4-6n^5+n^6}{-4+3n-2n^2+3n^3-4n^4+n^5}$

```
#let gR = [((3, 1), 0); ((-6, 1), 1); ((10, 1),
   2); ((-16,1), 3); ((14,1), 4); ((-6,1),5);
   ((1,1), 6)], [((-4, 1), 0); ((3, 1), 1);
   ((-2,1),2); ((3,1), 3); ((-4,1), 4); ((1,1),
   5)];;
print $ solve gR;;

p =  -21/16  +  -21/8 x^2 +  21/16 x^3
q =  -16/7  +  32/7 x +  -64/21 x^2 +  16/21 x^3
r =  16/21  +  16/21 x +  16/21 x^2
majoration du degre : d = 0
f =  441/256
Resultat : ( 1x +  -1x^2 +  1x^3)/( -1 +  -2x^2
  +  1x^3)- : unit = ()
```

  Donc $S_n=a_n\frac{n-n^2+n^3}{-1-2n^2+n^3}=\frac{n!}{1+n+n^2}$

- $\Sigma_{k=0}^{n}\frac{1}{k!}$
  $\frac{a_n}{a_{n-1}}=\frac{1}{n}$

```
# print $ solve (Q.normalise ([((1, 1), 0)],
   [((1, 1), 1)]));;
p =  1
q =  1
r =  1x
majoration du degre : d = -1
pas de solution.
```

Il n'existe donc pas d'expression "fermée" de $\Sigma_{k=0}^{n}\frac{1}{k!}$.

# Calcul des racines entières d'un polynome rationnel

$a_0 + a_1X + ... + a_dX^d \in \mathbb{Z}[\mathbb{X}]$
Soit p $\in \mathbb{N}$ tel que $a_0 + a_1p + ... + a_dp^d = 0$
Alors $a_0 = -p(a_1 + ... + a_np^{d-1})$
Donc **p divise** $a_0$

# Preuve de la proposition 1 :

$\frac{S_n}{S_{n-1}}$ est une fraction rationelle en n ssi il existe deux fractions rationelles $R(X)$ et $\alpha(X)$ telles que :

- $\forall n \in \mathbb{N}$, $\frac{a_n}{a_{n-1}} = R(n)$
- $\forall n \in \mathbb{N}$, $S_n = \alpha(n)a_n$

**Preuve :**

- C'est suffisant : $\frac{S_n}{S_{n-1}} = \frac{\alpha(n)a_n}{\alpha(n-1)a_{n-1}} = \frac{\alpha(n)R(n)}{\alpha(n-1)}$ fraction rationelle en n.
- C'est nécessaire : supposons que $\frac{S_n}{S_{n-1}} = \sigma(n)$ est une fraction rationelle en n.
  On a $\frac{a_n}{a_{n-1}} = \frac{S_n - S_{n-1}}{S_{n-1} - S_{n-2}} = \frac{S_n - S_{n-1}}{S_{n-1}(1 - \frac{S_{n-2}}{S_{n-1}})} = \frac{\sigma(n) - 1}{1 - \frac{1}{\sigma(n-1)}}$
  Donc $R(X) = \frac{\sigma(X) - 1}{1 - \frac{1}{\sigma(X-1)}}$
  Et $a_n = S_n - S_{n-1} = S_n(1 - \frac{1}{\sigma(n)})$ donc $S_n = \frac{1}{1 - \frac{1}{\sigma(n)}}a_n$
  Donc $\alpha(X) = \frac{1}{1 - \frac{1}{\sigma(X)}}$

# Preuve de la proposition 3

- On a $1 = \frac{S_n - S_{n-1}}{a_n} = \frac{\alpha(n)a_n - \alpha(n-1)a_{n-1}}{a_n} = \alpha(n) - \frac{\alpha(n-1)}{R(n)}$
  Donc $1 = \alpha(n) - \frac{\alpha(n-1)p(n-1)r(n)}{p(n)q(n)}$
  Donc $\forall n \in \mathbb{N}, p(n)q(n) = \alpha(n)p(n)q(n) - \alpha(n-1)p(n-1)r(n)$

- On pose $\beta(X) = \alpha(X)p(X)$
  On a donc $p(X)q(X) = \beta(X)q(X) - \beta(X-1)r(X)$

- $\beta$ est un polynôme !
  Supposons par l'absurde que ça ne soit pas le cas. Soit $z$ un pôle (complexe) de $\beta$.
  - $p(X)q(X) = \beta(X)q(X) - \beta(X-1)r(X)$ donc :
    $z$ pôle complexe de $\beta(X-1)$ ou $q(z) = 0$
  - $p(X+1)q(X+1) = \beta(X+1)q(X+1) - \beta(X)r(X+1)$ donc :
    $z$ pôle complexe de $\beta(X+1)$ ou $r(z+1) = 0$
  - $\beta$ a un nombre fini de pôles, donc il existe $z$ : $z$ pôle de $\beta$ et $z-1$ n'est pas pôle de $\beta$.
  - $z-1$ n'est pas pôle de $\beta$ donc $z$ n'est pas pôle de $\beta(X-1)$.
    Donc $q(z) = 0$.
    $\forall k \in \mathbb{N}$, $q(X) \wedge r(X+k) = 1$, donc $\forall k \in \mathbb{N}, r(z+k) \neq 0$.
    Donc $z+1$ pôle de $\beta$ et par récurrence : $\forall k \in \mathbb{N}$ : $z+k$ pôle de $\beta$.
  - Donc $\beta$ a un nombre infini de pôles, c'est absurde, donc $\beta$ est un polynôme.

- $p(X+1)q(X+1) = \beta(X+1)q(X+1) - \beta(X)r(X+1)$ et $q(X+1) \wedge r(X+1) = 1$ donc $\beta(X) = f(X)q(X+1)$, f polynôme. On a alors :
  $$p(X) = q(X+1)f(X) - f(X-1)r(X)$$
  $$\alpha(X) = \frac{q(X+1)}{p(X)}f(X)$$