



## Computor v1

"Moi non plus j'ai pas le bac"

Sebastien Julliot [sjulliot@student.42.fr](mailto:sjulliot@student.42.fr)  
42 staff [pedago@staff.42.fr](mailto:pedago@staff.42.fr)

*Résumé: Ce projet est le premier d'une série ayant pour but de vous faire renouer avec les maths, qui vous seront très utiles -voire nécessaires- pour de nombreux autres projets.*

# Table des matières

<b>I</b>	<b>Préambule</b>	<b>2</b>
<b>II</b>	<b>Introduction</b>	<b>3</b>
<b>III</b>	<b>Objectifs</b>	<b>4</b>
<b>IV</b>	<b>Partie obligatoire</b>	<b>5</b>
<b>V</b>	<b>Partie bonus</b>	<b>6</b>
<b>VI</b>	<b>Consignes</b>	<b>7</b>

# Chapitre I

## Préambule

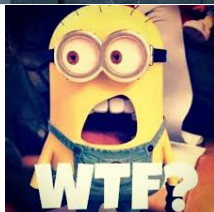
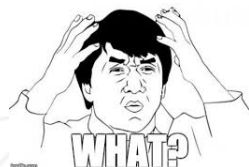
Un polynôme est une expression formelle de la forme :

$$P(X) = \sum_{k=0}^n a_k X^k \quad (\text{I.1})$$

Où X est appelé indéterminée du polynôme.

Le produit de deux polynômes est ainsi défini par

$$\left( \sum_{i=0}^n a_i X^i \right) \left( \sum_{j=0}^m b_j X^j \right) = \sum_{k=0}^{n+m} \left( \sum_{i+j=k} a_i b_j \right) X^k. \quad (\text{I.2})$$



La video offre une explication plus... comprehensible.

# Chapitre II

## Introduction

Le but de ce sujet est de vous faire coder un programme qui résout des équations simples. Le programme prendra en paramètre une équation polynomiale. C'est-à-dire ne faisant intervenir que des puissances, aucune fonction compliquée. Le programme devra afficher sa (ses) solution(s).

Pourquoi des polynômes? Parce que c'est l'un des outils mathématiques les plus simples et les plus puissants. On s'en sert dans tous les domaines et à tous les niveaux pour simplifier et exprimer beaucoup de choses. Par exemple, les fonctions `sin`, `cos`, et `tan` sont calculées à l'aide de polynômes.



En fait, il existe même un résultat : le théorème de Stone-Weierstrass, qui dit que toutes les fonctions "courantes", (celles qui sont bien lisses et jolies), peuvent être exprimées à l'aide de polynômes.

# Chapitre III

## Objectifs

L'idée est de vous faire (re)prendre contact avec la manipulation d'outils mathématiques élémentaires, qui pourront vous être utiles dans de nombreux autres sujets de 42. Il ne s'agit donc pas de "faire des maths pour faire des maths", mais bien de vous permettre d'aborder plus progressivement et sereinement les sujets où il y en a besoin.

Voici une liste non exhaustive des sujets où savoir ce que sont et comment manipuler les polynômes pourrait bien vous être utile :

- Fractol
- RT
- mod1
- Expert System
- Infin Mult

Par ailleurs, ce petit sujet sera complété par d'autres sur des sujets variés, pour comprendre ce que vous faites plutôt que simplement copier une formule sur Internet.

# Chapitre IV

## Partie obligatoire

Ecrivez un programme qui résout une équation polynomiale de degré inférieur ou égal à 2. Vous devrez afficher au moins :

- La forme réduite de l'équation.
- Le degré de l'équation.
- Sa ou ses solution(s), ainsi que le signe du discriminant quand cela a du sens.

Exemples :

```
$>./computer "5 * X^0 + 4 * X^1 - 9.3 * X^2 = 1 * X^0"
Reduced form: 4 * X^0 + 4 * X^1 - 9.3 * X^2 = 0
Polynomial degree: 2
Discriminant is strictly positive, the two solutions are:
0.905239
-0.475131
$>./computer "5 * X^0 + 4 * X^1 = 4 * X^0"
Reduced form: 1 * X^0 + 4 * X^1 = 0
Polynomial degree: 1
The solution is:
-0.25
./computer "8 * X^0 - 6 * X^1 + 0 * X^2 - 5.6 * X^3 = 3 * X^0"
Reduced form: 5 * X^0 - 6 * X^1 + 0 * X^2 - 5.6 * X^3 = 0
Polynomial degree: 3
The polynomial degree is stricly greater than 2, I can't solve.
```

On considèrera toujours que l'entrée est bien formatée, ie. tous les termes sont de la forme  $a * x^p$ . Les puissances sont bien ordonnées et toutes présentes. Attention, cela ne signifie pas forcément que l'équation soit soluble ! Dans ce cas, votre programme doit le détecter et l'indiquer. Pensez aussi aux coefficients nuls, négatifs, pas entiers ...

Il existe peut-être des cas particuliers que vous devez gérer. Par exemple pour l'équation  $42 * X^0 = 42 * X^0$ , tous les nombres réels sont solution...



La résolution des équations de degré trois ou plus n'est pas demandée. Ca ferait un excellent nouveau sujet non ? :)

# Chapitre V

## Partie bonus

Voici une liste de bonus qu'il pourrait être utile d'implémenter :

- Gérer les erreurs sur l'entrée (lexique et syntaxe).
- Gérer les entrées sorties sous forme naturelle.

```
./computer "5 + 4 * X + X^2= X^2"  
Reduced form: 5 + 4 * X = 0  
Polynomial degree: 1  
The solution is:  
-1.25
```

- Afficher la (les) solution sous forme de fraction irréductible quand c'est intéressant.
- Afficher des étapes intermédiaires
- ...

# Chapitre VI

## Consignes

- Pensez aux solutions complexes quand le degré vaut 2 ;)
- Le choix du langage est à votre discrétion.
- Cela dit, vous n'avez évidemment droit à aucune fonction/bibliothèque mathématique (hors addition et multiplication de réels) que vous n'ayez pas implémentée vous-mêmes.
- Si vous travaillez dans un langage compilable (C/C++ notamment), vous rendrez un Makefile contenant les règles habituelles.
- En C, vous respecterez bien sûr la norme.

Bon courage !