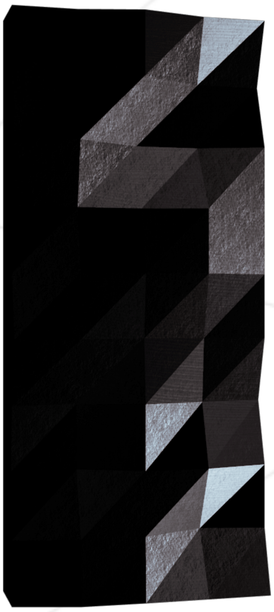

Computor v1

"I don't have my high school diploma either"

Sebastien Julliot sjulliot@student.42.fr
42 staff pedago@staff.42.fr

Summary: This project is the first of a serie to renew your relationship to mathematics, it will be very useful, essential even, for many more projects.

Contents

Chapter I

Foreword

A polynomial is the formal expression of the following form:

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

Where X is called indeterminates of the polynomial.

The product of two polynomials is defined by

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

The video offers an more... understandable explanation.

Chapter II

Introduction

The purpose of this project is to make you code a program that solves simple equations. The program takes a polynomial equation. That is to say, involving only powers, no complicated functions. The program should display its solution(s).

Why polynomials? Because it is one of the simplest and most powerful mathematical tools. It is used in all domains and at all levels to simplify and express many things. For example, the functions `sin`, `cos`, and `tan` are computed using polynomials.

> In fact, there is even one result: the theorem of Stone-Weierstrass, who says that all "current" functions, (those which are very smooth and pretty), can be expressed at using polynomials.

Chapter III

Objectives

The idea is to make you (re) get in touch with the manipulation of basic mathematical tools, which can be useful to you in many other subjects of the cursus. It is therefore not a question of "doing maths to do maths", but to allow you to approach more progressively and serenely the subjects where there are needed.

Here is a non-exhaustive list of subjects where to know what are and how to manipulate the polynomials might be useful to you::

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

Actually, this small project will be complemented by others on various subjects, to understand what you do rather than simply copy a formula on the Internet.

Chapter IV

General Instructions

- Think about complex solutions when the degree is worth 2. ;)
- The language is up to you.
- That said, you are not allowed any mathematic function/library (beside additions and multiplications of real numbers) that you did not implement yourself.
- If you are working in a compilable language (C/C++ for ex) you will submit a Makefile with all the usual rules.

Good luck !

Chapter V

Mandatory part

Write a program that solves a polynomial equation of degree less than or equal to 2. You will need to display at least:

- The reduced form of the equation.
- The degree of the equation.
- Its solution(s), as well as the sign of the discriminant when it makes sense..

Examples:

```
$>./computor "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
$>./computor "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
./computor "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.
```

It will always be considered that the input is well formatted, ie. all terms are of the form $a * x^p$. The powers are well ordered and all present. Attention, this does not necessarily mean that the equation is solvable! In this case, your program must detect it and indicate it. Think also of the coefficients that are null, negative, not integers...

There may be special cases that you need to manage. For example, for the following equation $42 * X^0 = 42 * X^0$, all the real numbers are solution...

The resolution of equations of degree three or more is not required. It would make an excellent new subject, right? :)

Chapter VI

Bonus part

Here is a list of bonuses that could prove useful to implement:

- Manage errors on the input (lexicon and syntax).
- Manage inputs outputs in a natural form.

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

- Display the solution(s) under the form of irrational fractions when it is interesting..
- Display intermediate steps.
- ...