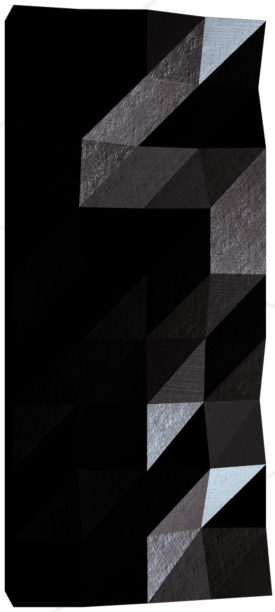


Computor v1

"I'm not a graduate either"

Summary: This project is the first in a series that aims to make you rekindle with maths. They will be quite useful - not to say essential - for numerous projects to come.

Version: 6.2

Contents

Chapter I

Preamble

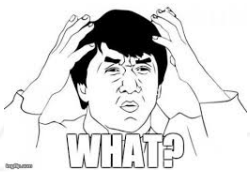
A polynomial is a formal expression of the form:

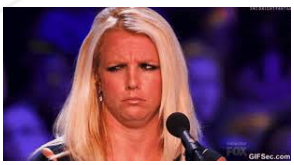
$$P(X) = \sum_{k=0}^{n} a_k X^k \tag{I.1}$$

Where X is called the polynomial indeterminate.

Thus, the product of two polynomials is defined by

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





Chapter II

Introduction

This project aims to make you code a simple equation solving program. It will take polynomial equations into account. These equations will only require exponents. No complex function. The program will have to display its solution(s).

Why polynomials? Just because it's one of the simplest and most powerful tools in mathematics. It is used in every field on every level to simplify and express many things. For instance, they help calculate functions such as `sin`, `cos`, et `tan`.

```
There actually is a result:  the Stone-Weierstrass theorem, that
states that every "current" function (the nice and pretty ones) can
be expressed with a polynomial.
```

Chapter III

Objectives

The idea is to get you (back) in touch with the manipulation of elementary mathematic tools you will be able to use in seveal 42 subjects. The idea is not to "just make maths". This exercise will allow you to approach exercises that will require these skills and knowledge with a restful mind.

Here is a non-exhaustive list of subjects that will require a basic knowledge of the polynomials - what they are, and how to handle them:

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

Besides, this small subject will be completed with others, touching various subjects that will help you understand what you do, rather than just copy a formula you've found on the Internet.

Chapter IV

Mandatory part

Write a program that solves a polynomial second or lower degree equation. You will have to show at least:

- The equation in its reduced form.
- The degree of the equation.
- It's solution(s) and the polarity of the discriminant if it makes sens.

Ex 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 strictly greater than 2, I can't solve.

$>./computor "6 * X^0 = 6 * X^0"
Reduced form: 0 * X^0 = 0
Any real number is a solution.

$>./computor "10 * X^0 = 15 * X^0"
Reduced form: -5 * X^0 = 0
No solution.

$>./computor "1 * X^0 + 2 * X^1 + 5 * X^2 = 0"
Reduced form: 1 * X^0 + 2 * X^1 + 5 * X^2 = 0
Polynomial degree: 2
Discriminant is strictly negative, the two complex solutions are:
-1/5 + 2i/5
-1/5 - 2i/5
```

We will always expect the entry to have the right format, ie. every term respect the form $a * x^p$. Exponents are organized and present. Beware, it doesn't mean the equation

has a solution! If so, your program should detect it and specify it. You should also think of zero, negative or non whole coefficients...

There might be exceptions you will have to manage. In the equation $42 * X^0 = 42 * X^0$, for instance, each real number is a solution...

`Third degree equation resolution is not required. But that will make for an amazing new subject, right? :)`

`If no argument is provided, the program will wait for an equation to be entered on the STDIN.`

Chapter V

Bonus part

Here is a bonus list you might want to implement:

- Manage entry mistakes (vocabulary and syntax).
- Manage free form entrie.

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

- Display solution(s) as an irreducible fraction if it's interesting.
- Display the intermediate steps.
- ...

The bonus part will only be assessed if the mandatory part is PERFECT. Perfect means the mandatory part has been integrally done and works without malfunctioning. If you have not passed ALL the mandatory requirements, your bonus part will not be evaluated at all.

Chapter VI

Submission and peer-evaluation

Turn in your assignment in your `Git` repository as usual. Only the work inside your repository will be evaluated during the defense. Don't hesitate to double check the names of your folders and files to ensure they are correct.

- Think of complex solution when the degree equals 2. ;)
- You're free to pick your favorite language.
- That being said, you obviously cannot use a math lib function (except for subtraction, division, addition and multiplication of real numbers) that you would not have implemented yourself.
- If you work in a compilable language (C/C++ in particular), you will present a Makefile that includes the usual set of rules.