MOOC Init. Prog. C++
Exercices supplémentaires facultatifs semaine 2

Résolution d'une équation du 3e degré (niveau 3)

Cet exercice correspond à l'exercice n°10 (pages 23 et 207) de l'ouvrage *C++ par la pratique* (3e édition, PPUR).

On veut écrire un programme qui demande trois valeurs (a0, a1, a2) à l'utilisateur et affiche la (ou les) solution(s) réelle(s) *z* de l'équation du troisième degré :

$$z^3 + a_2 z^2 + a_1 z + a_0 = 0$$

Indications - commencer par calculer :

$$Q = (3a_1 - a_2 a_2)/9$$
$$R = (9a_2 a_1 - 27a_0 - 2{a_2}^3)/54$$
$$D = Q^3 + R^2$$

Démonstration des formules à la page : http://mathworld.wolfram.com/CubicFormula.html

Si D < 0, on calcule les trois solutions réelles ainsi :

$$\theta = \mathrm{acos}\left(\frac{R}{\sqrt{(-Q)^3}}\right)$$

(la fonction `acos` existe en C++ dans la bibliothèque `<cmath>` - i.e. il faut inclure ce fichier en début de programme)

$$z_1 = 2\sqrt{-Q}\,\cos(\theta/3) - a_2/3$$

Pour π, vous pouvez utiliser la constante `M_PI` souvent présente dans `<cmath>` (ce n'est pas rigoureusement une constante officielle, mais elle est présente sur la plupart des compilateurs. Une alternative consiste à utiliser `acos(-1.0)` : `const double pi(acos(-1.0));`)

$$z_2 = 2\sqrt{-Q}\,\cos((\theta + 2\pi)/3) - a_2/3$$

$$z_3 = 2\sqrt{-Q}\,\cos((\theta + 4\pi)/3) - a_2/3$$

Sinon, on calcule :

$$S = (R + \sqrt{D})^{1/3}$$

La racine cubique de x est obtenue en C++ par «`pow(x,1.0/3.0)`». Notez que la racine cubique de (`-x`) est l'opposé de la racine cubique de `x`.
Il faut en effet traiter séparément le cas où `x < 0` du cas `x >= 0`, car

C++ ne l'accepte pas dans la fonction `pow`.

T = $(R - \sqrt{D})^{1/3}$

<u>Si D=0 et S+T != 0</u>, il y a 2 racines :

$z_1 = S + T - a_2/3$
$z_2 = -(S + T)/2 - a_2/3$ (racine double)

<u>Sinon</u>, il y a une racine unique : z_1 ci-dessus.
