

# TD N°2 : Contrôle du flux d'instructions

Ahmed Ammar (ahmed.ammar@fst.utm.tn)

Institut Préparatoire aux Études Scientifiques et Techniques, Université de Carthage.

Nov 6, 2019

## Table des matières

### Exercice 1 : Comparer deux entiers

Écrivez un programme qui vous demande de saisir 2 nombres entiers et affiche la plus petite de ces valeurs.

### Exercice 2 : Comparer deux chaînes

Écrivez un programme qui demande d'entrer 2 chaînes et qui affiche la plus grande des 2 chaînes (celle qui contient le plus de caractères).

### Exercice 3 : Convertir Euro contre Dinar Tunisien | EUR TND

- Écrivez un programme qui convertit l'euro (EUR) en dinar tunisien (TND) :
- Le programme commence par demander à l'utilisateur d'indiquer par une chaîne de caractères 'EUR' ou 'TND' la devise du montant qu'il entrera.
  - Ensuite, le programme exécute une action conditionnelle de la forme :

```
if devise == 'EUR' :  
    # Expression 1  
elif devise == 'TND' :  
    # Expression 2  
else :  
    # affichage d'un message d'erreur
```

## Exercice 4 : Résolution d'une équation du second degré

Soit l'équation du second degré  $ax^2 + bx + c = 0$  où  $a$ ,  $b$  et  $c$  sont des coefficients réels.

a) Écrivez un programme qui demande d'entrer les coefficients et affiche les solutions de l'équation.

### Indication. Solutions analytiques

Des solutions sont recherchées dans le cas général, compte tenu du discriminant  $\Delta = b^2 - 4ac$ , l'équation admet comme solutions analytiques :

$$\begin{cases} \Delta > 0 & \text{deux solutions réelles : } x_1 = \frac{-b - \sqrt{\Delta}}{2a}; \quad x_2 = \frac{-b + \sqrt{\Delta}}{2a} \\ \Delta = 0 & \text{une solution double : } x_0 = \frac{-b}{2a} \\ \Delta < 0 & \text{deux solutions complexes : } z_1 = \frac{-b - i\sqrt{-\Delta}}{2a}; \quad z_2 = \frac{-b + i\sqrt{-\Delta}}{2a} \end{cases}$$

### Algorithme

#### Définition

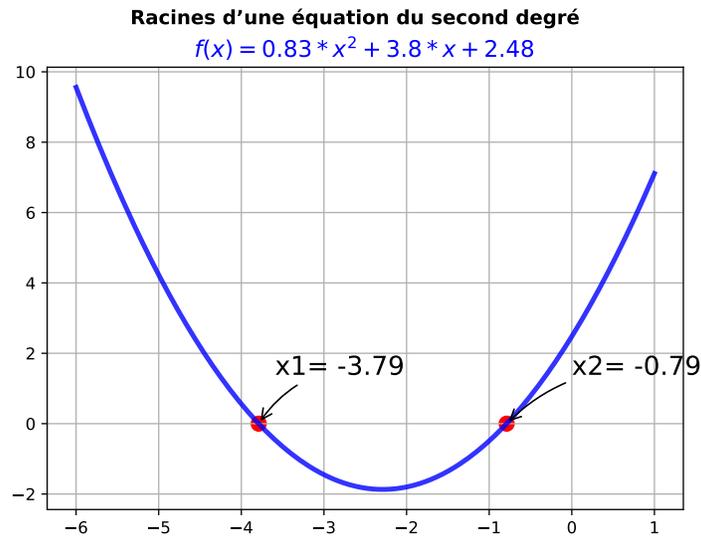
Ensemble de règles opératoires dont l'application permet de résoudre un problème énoncé au moyen d'un nombre fini d'opérations. Un algorithme peut être traduit, grâce à un langage de programmation, en un programme exécutable par un ordinateur. Source: LAROUSSE

#### Pseudo-code de l'algorithme

Présentons tout d'abord un pseudo-code de l'algorithme, c'est-à-dire le détail des opérations à effectuer sans syntaxe propre du langage.

```
# Calcul des racines de l'équation du second degré
a, b et c ← ... # Assignation des variables a, b et c (variables de type réel) en utilisant la fonction
input()
Δ ← b2 - 4ac
si Δ est positive:
... x1 ←  $\frac{-b - \sqrt{\Delta}}{2a}$ 
... x2 ←  $\frac{-b + \sqrt{\Delta}}{2a}$ 
... # Affichez les solutions trouvées
sinon si Δ est nul:
... x0 ←  $\frac{-b}{2a}$ 
... # Affichez la solution trouvée
sinon si Δ est négative:
... z1 ←  $\frac{-b - i\sqrt{-\Delta}}{2a}$ 
... z2 ←  $\frac{-b + i\sqrt{-\Delta}}{2a}$ 
... # Affichez les solutions trouvées
```

- b) Soit la fonction  $f(x) = 0.83x^2 + 3.8x + 2.48$ . En utilisant le programme précédent, trouvez les solutions pour  $f(x) = 0$ .
- c) La représentation graphique de  $f(x)$  est indiquée ci-dessous :



- Nous allons utiliser une fonction `EqSecondDegree(a,b,c)` dans **TD N°3** pour reproduire cette figure en utilisant les bibliothèques `numpy` et `matplotlib`.
- Écrivez la fonction `EqSecondDegree(a,b,c)` qui **renvoie** les solutions de l'équation  $ax^2 + bx + c = 0$ .
  - Enregistrez la fonction `EqSecondDegree(a,b,c)` dans un script Python `racines.py`.
- d) En utilisant la fonction `EqSecondDegree(a,b,c)`, trouvez les solutions de  $f(x) = 0$ .

### Exercice 5 : programmez une boucle `while`

Définir une séquence de nombres :

$$x_n = n^2 + 1$$

pour les entiers  $n = 0, 1, 2, \dots, N$ . Écrivez un programme qui affiche  $x_n$  pour  $n = 0, 1, \dots, 20$  en utilisant une boucle `while`.

### Exercice 6 : Créer une liste avec une boucle `while`

Stockez toutes les valeurs  $x_n$  calculées dans l'exercice 5 dans une liste (à l'aide d'une boucle `while`). Affichez la liste complète (en un seul objet).

### **Exercice 7 : Programmer une boucle for**

Faites l'exercice 6, mais utilisez une boucle `for`.

### **Exercice 8 : Ecrire une fonction Python**

Écrivez une fonction `x(n)` pour calculer un élément dans la séquence  $x_n = n^2 + 1$ . Appelez la fonction pour  $n = 4$  et affichez le résultat.

### **Exercice 9 : Renvoyer trois valeurs d'une fonction Python**

Écrivez une fonction Python qui évalue les fonctions mathématiques  $f(x) = \cos(2x)$ ,  $f'(x) = -2\sin(2x)$  et  $f''(x) = -4\cos(2x)$ . Retourner ces trois valeurs. Écrivez les résultats de ces valeurs pour  $x = \pi$ .