INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

Introduction

Université de Toulouse

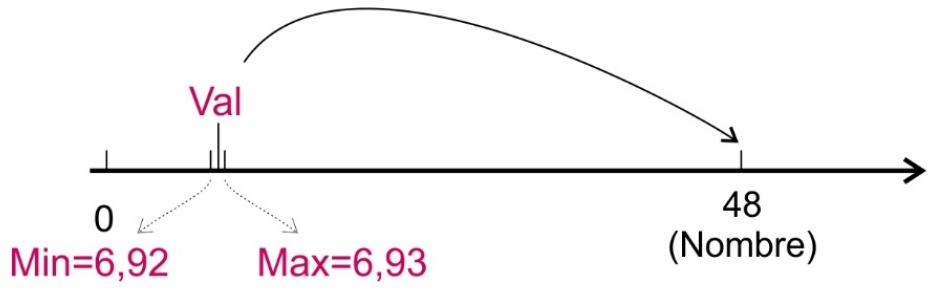
Algorithme de la raciné carré

```
Min ← 0

Max ← Nombre

Precision ←  0,0001

Val = Nombre

Tant que ( |Val*Val - Nombre| < Precision)
Faire

        Val ← (Max + Min)/2

        Si (Val*Val > Nombre)

                Max ← Val

        Sinon

                Min ← Val

        FinSi

FinTantque
```

Algorithme de la raciné carré

```
Min ← 0
Max ← Nombre
Precision ←  0,0001
Val = Nombre
Tant que ( |Val*Val - Nombre| < Precision)
Faire
        Val ← (Max + Min)/2
        Si (Val*Val > Nombre)
                Max ← Val
        Sinon
                Min ← Val
        FinSi
FinTantque
```

On continue la procédure jusqu'à ce que l'on obtienne le résultat avec une précision désirée



Questions / angoisses…

- Est-ce dur ?
- Comment apprendre « toutes les choses » sur le code
- J'aime pas bosser sur un PC…
- Ce qui touche à internet….
- J'angoisse….J'y connais rien… J'ai du mal…Aucune compétence…pas doué
- Excel ? Arduino ?
- Faire du code sur papier
- Google est mon ami…

L'algorithmique

- ***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*** **» (Larousse)**
  - Algorithme ⇒ analyse, réflexion, choix
  - Codage (langage) = mise en œuvre
- **Objectif premier de cet enseignement : savoir construire des algorithmes**
- **Objectif secondaire : savoir les mettre en œuvre**

Algorithme vs organigramme

- **Organigramme**

Je n'utiliserai pas d'organigrammes et ne vous encourage pas à le faire….
Je ne vous demanderai pas de formaliser systématiquement votre solution en algorithme « pur » mais je vous encourage à le faire…..

- Graphique, visuel, intuitif
- Mais : limité, difficile à faire évoluer, inadapté pour les « gros problèmes »

- **Algorithme**

```
Variable
                                    rs
                                    ecter à x : 2x

                                    ecter à x : x + 1

Fin
```

- Textuel, formel
- Permet la notion de fonction
- Facilement modifiable
- Proche du codage

Construire un algorithme

- Comprendre un algorithme (bien écrit) ne pose généralement pas trop de soucis
- Trouver l'algorithme qui répond à une problématique donnée (ex de la racine carrée) est plus difficile
- Nécessite de l'entrainement pour acquérir l'agilité cognitive
- Si mal à l'aise : faire et refaire plutôt que lire et relire.
- => Installer l'environnement de développement chez vous

Pour le premier TD :
avoir installé Python Anaconda sur votre ordinateur
https://www.anaconda.com/download/

Python

- Nouveauté 2018 : des « vrais » cours…Cours différentiés au second semestre
- Python depuis l'an passé
- Le langage à la mode (il date pourtant de 1990)
- Langage dédié aux calculs scientifiques (mais pas que)
- Prisé des mathématiciens
- GRATUIT
- De nombreuses bibliothèques préexistantes
- Langage interprété….Nous y reviendrons..