**Introduction :**

**Algorithmique**

Un Algorithme :

* Méthode détaillée [*Séquence d’instructions*] permettant de résoudre un problème [*Une classe de problèmes*].
* En pratique, on désigne souvent par algorithmes les parties difficiles d’un logiciel.

But :

* Trouver (par un algorithme travaillant avec des **intervalles**) les valeurs des variables qui minimisent une fonction tout en respectant des contraintes.

**Développement Initiatique**

* Les algorithmes (*Et les langages de programmation*) offrent un jeu très restreint d’instructions de base. L’algorithmique est l’art de savoir manier ce petit vocabulaire pour exprimer toutes les méthodes imaginables. « Vocabulaires » :
  + Entrées/Sorties ;
  + Variables et **structures de données** plus sophistiquées (*dont les tableaux*) ;
  + Affectations de variables ;
  + Instructions conditionnelles (*si alors sinon*) ;
  + Répétition d’un ensemble d’instructions (*boucles*) ;
  + Appel à des sous-programmes (*procédures, fonctions, méthodes ; une infinité de sous-programmes existent*).

Communication avec l’être humain :

* Syntaxe :

Entrée au clavier :

* Syntaxe :
  + 1/ Le programme se bloque et attend une valeur saisie au clavier, suivie d’entrée.
  + 2/ La valeur saisie est affectée à la variable en argument.
* Par défaut, on suppose que l’utilisateur saisit une valeur du bon type.
* *Exemple :*

*Définition :*

* Une **variable** est un symbole, un nom associé à une information.  
    
  A une variable correspond un emplacement mémoire (RAM) permettant de stocker le résultat de calculs (intermédiaires) d’un algorithme/programme.
* Une variable a généralement un **type**, défini **statiquement** (*Une fois pour toutes, dans le fichier source*) ou **dynamiquement** (*pendant l’exécution de l’algorithme : une variable typée dynamiquement peut donc changer de type au cours de l’exécution du programme …*), selon le langage.
* Types principaux : entier (*relatifs*), réel (*flottant, double*), booléen, caractère, chaînes de caractères.