

## PARTIE 2 : ALGORITHMIQUE & PROGRAMMATION

### CHAPITRES 1 & 2 – INTRODUCTION À L'ALGORITHMIQUE ET À LA PROGRAMMATION

## 1 Typage des variables

Définition

**Variable** : une variable permet de stocker des informations. Elle est définie par un identifiant (un nom), un type, une valeur, une référence et des opérations.

**Référence** : une référence est un alias pointant vers une adresse mémoire.

**Type** : le typage correspond à la nature d'une variable (entier, chaîne de caractères, liste ...). En Python, le typage est **dynamique** ce qu'il signifie que Python reconnaît le type de variable dès son affectation.

**Valeur** : la valeur correspond à la donnée que l'utilisateur veut stocker en mémoire.

Les objets non mutables sont ceux que l'on ne peut modifier après leur création : les entiers (int), les flottants (flt), les chaînes de caractères (str), les booléens, les tuples etc... Si on modifie un de ces types, une nouvelle référence est créée.

Les objets mutables peuvent être modifiés après leur création : les listes, les dictionnaires...

Exemple



On ne peut pas modifier la variable ch sans changer de référence. C'est un type non mutable.

```
>>> ch = 'abcde'
>>> ch[0] = 'a'
TypeError: 'str' object does not support item assignment
```

On peut modifier des éléments variable ch sans changer de référence. C'est un type mutable.

```
>>> ch = [a,b,c,d,e]
>>> ch[0] = 'a'
```

## 2 Instructions et expressions

Définition

**Instruction**

**Expression**

Exemple

## 3 Structures algorithmiques

## 4 Structure d'un programme