

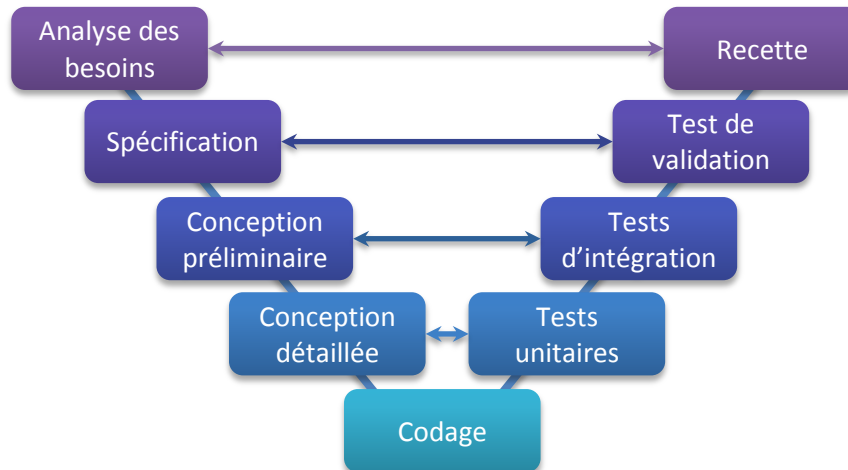
# Notions de programmation

I3 – Chapitre 1

## I. Critères de qualité d'un bon programme

1. lisible
2. fiable
3. maintenable
4. réutilisable
5. portable
6. correct
7. efficace (complexité)
8. faire face à des contraintes économiques

## II. Méthodologie et cycle en V



- **Spécification** : Analyse descendante : abstraire, décomposer, combiner.
- **Conception préliminaire** : Choix d'un paradigme, traduction des fonctions.
- **Conception détaillée** : Détail des fonctions en pseudo-code.

## III. Complexité

### 1. Notations

Taille du problème :  $n$

Nombre d'opérations :  $T(n)$

Taille mémoire :  $M(n)$

$O(f(n))$  : au pire

$\Omega(f(n))$  : au mieux

$\theta(f(n))$  : en moyenne

### 2. Calcul

Schéma A	$O(A^O)$	$\Omega(A^\Omega)$
Instruction de base	$O(1)$	$\Omega(1)$
Séquence $A_1, A_2, \dots, A_n$	$O(\max(A_1^O, A_2^O, \dots, A_n^O))$	$\Omega(\max(A_1^\Omega, A_2^\Omega, \dots, A_n^\Omega))$
$A_1$ ou $A_2$	$O(\max(A_1^O, A_2^O))$	$\Omega(\min(A_1^\Omega, A_2^\Omega))$
<i>pour</i> avec $n$ itérations	$O(n \times A^O)$	$\Omega(n \times A^\Omega)$
<i>tant que</i> $n$ itérations max.		$\Omega(1)$
<i>répéter</i> $n$ itérations max.		$\Omega(A^\Omega)$