

Chapitre 1 : algorithme et programme

"Objectifs"

À la fin du chapitre, on doit savoir :

- distinguer algorithme et programme
- donner au moins trois exemples de langages de programmation
- distinguer langage compilé et langage interprété

Algorithme et ordinateur

"Définition"

Un **algorithme** est une suite finie d'instructions non ambiguës, exécutables mécaniquement, qui transforme des données d'entrée en un résultat de sortie après un nombre fini d'étapes.

"Exercice 1 : algorithme d'addition"

1. Effectuer l'addition des nombres 9870756 et 4321807.
2. Décrire l'algorithme que vous avez utilisé

Langage de programmation



"Définition"

Un **langage** de programmation traduit un algorithme exprimé en langage humain dans un **programme** compréhensible par un ordinateur. Un programme est au départ un texte ou **code source** qui est traduit en **langage machine** par un autre programme appelé **compilateur**. Il existe une multitude de langages de programmation adaptés à des usages différents.

Un langage de programmation est intermédiaire entre :

- le **langage humain**, pratique mais souvent ambigu ;
- le **langage machine**, directement exécutable par le processeur de l'ordinateur mais très difficile à écrire directement.



"Exemple : diversité des langages de programmation"

Les trois programmes ci-dessous permettent tous d'afficher 'Bonjour' mais sont écrits dans trois langages différents.

En Python :

```
print("Bonjour")
```

En JavaScript :

```
console.log("Bonjour");
```

En C :

```
#include <stdio.h>

int main() {
    printf("Bonjour\n");
    return 0;
}
```

Compilation et interprétation



"Définition"

Un **compilateur** est un programme qui traduit un code source en un autre langage, souvent en langage machine, avant l'exécution du programme.

Dans un langage compilé, la traduction est généralement faite avant l'exécution.

Un **interpréteur** est un programme qui prend en charge l'exécution d'un programme source, instruction après instruction ou après une traduction intermédiaire.

Dans un langage interprété, on lance généralement le programme à l'aide d'un interpréteur.



"Exemples de langages compilés ou interprétés"

Langage	Compilé	Interprété
C	oui	non
Rust	oui	non
Python	en partie, vers du bytecode	oui
Java	en partie, vers du bytecode	oui, par la machine virtuelle Java
JavaScript	parfois, à la volée	oui



"Attention"

La distinction entre langage compilé et langage interprété est utile, mais simplifiée. En réalité, beaucoup de langages modernes utilisent des mécanismes mixtes : traduction intermédiaire, compilation à la volée, machine virtuelle ... Par exemple l'interpréteur Python compile le code source Python en `bytecode` qui est exécuté sur une machine virtuelle.

