

Le langage C

Voici les choses essentielles à connaître sur le langage C.

5TTR

6TTR



Découverte

Objectifs

À la fin de ce chapitre, tu sauras :

- Expliquer la différence entre un langage **compilé** et un langage **interprété**.
- Comprendre pourquoi on apprend encore le C aujourd'hui.
- Connaître les outils nécessaires pour programmer en C.

Le C en quelques mots

Le langage C a été créé dans les années 1970.

- Il a servi à écrire **le système Unix** et plus tard **Linux**.
- Beaucoup de logiciels systèmes (compilateurs, bases de données, microcontrôleurs) utilisent encore le C.
- Le C est un langage **rapide** et "**proche**" de la machine : il permet de comprendre comment l'ordinateur gère la mémoire et exécute les instructions.

Langages compilés et interprétés

Il existe deux grandes familles de langages :

- **Langages interprétés** (ex. Python, JavaScript)
 - Ton code est **lu ligne par ligne** par un interpréteur.
 - Tu dois avoir cet interpréteur installé pour lancer ton programme.
 - Exemple : `python moncode.py` → il faut Python installé.
 - Avantage : facile à tester rapidement.

- Inconvénient : plus lent, dépend toujours de l'interpréteur.
- **Langages compilés** (ex. C, C++)
 - Ton code est **traduit en langage machine** avant d'être exécuté → création d'un fichier exécutable.
 - Une fois compilé, ton programme peut tourner **sans avoir besoin du compilateur**.
 - Exemple : `moncode.c` → compilation → `moncode.exe`, exécutable directement.
 - Avantage : programme **rapide et autonome**.
 - Inconvénient : il faut passer par l'étape de **compilation**.

💡 Comparaison rapide :

- Python → `python moncode.py` (l'interpréteur est indispensable).
- C → `gcc moncode.c -o moncode` puis `moncode.exe` (le programme compilé peut être exécuté seul).

Pourquoi apprendre le C

- Comprendre **comment fonctionne un ordinateur** en profondeur.
- Apprendre la rigueur d'un langage plus « strict » que Python ou JavaScript.
- Utile pour la **programmation système**, l'**embarqué** (microcontrôleurs, robots), et même pour certains moteurs de jeux vidéo.

Les outils à utiliser

Pour écrire et lancer un programme en C, tu as besoin de :

1. **Un éditeur de code / IDE** : Code::Blocks, Visual Studio Code, CLion...
2. **Un compilateur** : le plus courant est `gcc` (GNU Compiler Collection).
3. **Un terminal / console** (éventuellement) pour compiler et exécuter ton programme.

👉 On fera un article spécifique pour t'expliquer **comment installer et utiliser ces outils**.

🔥 **Compilateur**: logiciel/programme qui traduit un code source en langage machine exécutable par l'ordinateur.

Exercices de réflexion

1. Classe ces langages dans la bonne catégorie : C, JavaScript, Python, C++, C#.

2. Pourquoi dit-on qu'un programme compilé est plus rapide qu'un programme interprété ?
3. Imagine : tu veux programmer un petit jeu sur ton navigateur. Choisirais-tu plutôt C ou JavaScript ? Pourquoi ?
4. Explique pourquoi tu ne peux pas lancer directement `moncode.c` comme un fichier exécutable.

À retenir

- Le C est un langage **compilé**, contrairement à Python ou JavaScript qui sont **interprétés**.
- Un langage interprété → nécessite **toujours l'interpréteur installé** pour exécuter ton programme.
- Un langage compilé → produit un **exécutable autonome**, que tu peux lancer directement.
- Le C reste aujourd'hui incontournable pour comprendre **l'informatique au plus près de la machine**.

Vidéo à la une à regarder sur Youtube:  <http://youtu.be/CYvJPra7Ebk>