

C++

- [À propos de ce livre](#)
- [1 - Histoire, philosophie et modèle de compilation](#)
- [Types fondamentaux et représentation mémoire](#)
 - [Endianness](#)

À propos de ce livre

Objectifs

Ce livre numérique a pour but de constituer une référence courte du C++, en proposant un lien vers des sources plus complètes quand nécessaire. L'idée est de pouvoir un aperçu rapide de la majorité des fonctionnalités offertes par le langage.

Conventions

Nous nous fixerons très peu de convention dans cet ouvrage, si ce n'est l'usage de certains termes anglais, afin de faciliter le lien avec le C++, bien que l'on mentionnera les traductions éventuelles, qui peuvent servir durant la rédaction des commentaires ou d'une documentation en langue française. Cela ne sera pas forcément systématique, on pourra aisément remplacer *vector* par *vecteur*, mais il vous sera épargné, en autres, le remplacement de *endianness* par *boutisme*. Les notions du langage seront présentées **en gras** lors de leur première apparition, puis *en italique*.

1 - Histoire, philosophie et modèle de compilation

Types fondamentaux et représentation mémoire

Endianness

Jusqu'au-boutisme

En informatique, un nombre entier est représenté sur des *octets*. L'**endianness** (*endianisme* ou *boutisme* en français), désigne l'ordre dans lequel ces octets sont placés. Il existe deux conventions opposées :

- **big-endian** : octets de poids fort en premiers (en français, *gros-boutiste* ou *gros-boutienne*)
- **little-endian** : octets de poids faible en premiers (en français, *petit-boutiste* ou *petit-boutienne*)

Le choix de l'*endianness* peut être fixée par l'architecture du processeur cible, ou par le format de données d'un fichier ou d'un protocole.