Explorer la documentation

Le langage C++ et la bibliothèque standard sont extrêmement riches, il est difficile (et inutile) d'apprendre toutes les subtilités et détails dès le début. Cela viendra avec l'expérience. Par contre, il est important pour votre apprentissage d'acquérir une certaine autonomie, en particulier pour trouver les informations dont vous aurez besoin. Ce cours vous donnera les éléments de compréhension indispensables, mais c'est dans votre intérêt de ne pas hésiter à faire vos propres recherches et lire des sources complémentaires.

L'un des sources les plus importantes est bien sûr la “documentation”.

Le site cppreference.com

La référence officielle du C++ est le document publié par l'organisme international de normalisation (ISO). Ce document est assez austère à lire (plus de 1300 pages), il est avant tout destiné à ceux qui implémentent les compilateurs. Il est payant, mais vous pouvez consulter le document de travail du comité C++, qui contient également les propositions pour le futur standard (Working Draft, Standard for Programming Language C++.

En pratique, vous n'allez donc pas utiliser ce document. Il existe différents sites internet qui présentent les spécifications du C++, de façon plus abordable. Dans ce cours, nous allons utiliser le site [cppreference.com]. Ce n'est pas la documentation officielle à proprement parlé, mais c'est l'une des plus à jour et complète.

La page principale est divisée en deux partie. En haut, la documentation du C++, et celle du C en dessous. Nous allons bien sûr utiliser celle concernant le C++.

* screenshot page principale

La documentation du C++ se décompose en plusieurs parties. En pratique, on peut distinguer trois parties : le langage proprement dit (http://en.cppreference.com/w/cpp/language), les fichiers d'en-tête (http://en.cppreference.com/w/cpp/header) et la bibliothèque standard (tous les autres parties en fait).

Faire une recherche dans la documentation

Vous allez utiliser la documentation dans deux situations. Soit vous recherchez les détails techniques d'une fonctionnalités du C++ à partir d'un mot-clé, soit vous rechercher quelle outils du C++ permet de réaliser une idée que vous avez en tête.

Partir d'un mot-clé est beaucoup plus simple. Il suffit de rechercher dans un moteur de cherche ce mot clé (par exemple en tapant “C++11 mot-clé”) pour trouver des références sur le sujet (éviter de rechercher dans un premier temps “C++ mot-clé”, il existe beaucoup de références obsolètes sur internet. Ne faites cela que dans un second temps, lorsque la recherche sur le C++11 a échouée).

Lorsque vous lirez ce cours ou lorsque vous poserez des questions sur les forums internet, on vous donnera probablement des mots-clés, comme par exemple `string` pour les chaînes de caractères ou `vector` pour les tableaux.

Sur cppreference.com, vous avez un champ de recherche en haut à droit. Il n'est pas nécessaire de préciser “C++” ici, puisque la recherche se focalisera sur le site. Vous pouvez donc taper directement votre mot-clé. Si vous tapez `vector` par exemple, vous allez obtenir la page principale de `std::vector` : http://en.cppreference.com/w/cpp/container/vector. Si vous essayez avec `string`, plusieurs pages peuvent correspondre, vous obtiendrez donc une page listant les différents résultats possible http://en.cppreference.com/mwiki/index.php?title=Special%3ASearch&search=string (pour les chaînes de caractères, c'est la première qui nous intéresse : `std::string` http://en.cppreference.com/w/cpp/string/basic_string).

Trouver la fonctionnalité du C++ correspond à une idée

La documentation du C++ (et la bibliothèque standard) est organisée par thématique. Il suffit généralement d'avoir une idée du module qui sera utilisé pour retrouver facilement le mot-clé correspondant. Mais pour cela, il faut être un peut familier avec l'organisation de la documentation, pour savoir où chercher. Il n'y a pas 36 méthodes pour cela : il faut utiliser la documentation le plus souvent possible, même pour rechercher une fonctionnalité que vous connaissez déjà.

Les titres de modules est suffisamment explicite, nous n'allons pas les détailler maintenant. Ils seront vu tout au long de ce cours, en fonction des besoins. Pour résumer les modules :

- “Utilities library” : différentes fonctionnalités utilitaires, pou gérer les types, la mémoire, les erreurs, les dates et temps, etc. ;
- “Strings library” : les chaînes de caractères ;
- “Containers library” : les collections de données (tableau, liste, etc.) ;
- “Algorithms library” : différents algorithmes génériques ;
- “Iterators library” : fonctionnalités pour manipuler les collections et algorithmes de façon générique ;
- “Numerics library” : tout ce qui concerne la manipulation des nombres, les nombres complexes et les nombres aléatoires ;
- “Input/output library” : entrée et sorties, aussi bien le clavier, que les fichiers ;
- “Localizations library” : prise en charge de l'internationnalisation ;
- “Regular expressions library” : expressions régulières ;
- “Atomic operations library” : les atomiques ;
- “Thread support library” : la programmation multi-threads.

Bien sûr, nombreux concepts cités ici vous sont inconnus. Pas de panique, tout sera détaillé dans la suite de ce cours.