

# 1. Introduction

## 1 But du cours et des TD

L'intitulé général du cours et des TD est « Informatique » mais les sujets abordés ne représentent qu'un aspect bien particulier de ce domaine : l'apprentissage du langage C en vue d'application à la physique. On abordera les sujets suivants :

- éléments du langage C ANSI
- notions de *Linux* pour utilisateur
- règles générales de programmation, indépendantes du langage utilisé, à observer pour écrire des programmes justes, efficaces et lisibles donc faciles à faire évoluer.

Liste des documents mis à la disposition des étudiants :

- transparents de cours
- polycopié de cours
- polycopié de TD comprenant :
  - énoncés d'exercices
  - résumé des commandes de base de l'éditeur *emacs*
  - résumé des commandes *Linux* de base

Tous les documents sont disponibles sur le site : <http://hebergement.u-psud.fr/mpo-informatique/>

## 2 Le système d'exploitation Linux

Le système d'exploitation est *Linux (Debian)*. Pour l'utilisateur il apparaît comme un langage de commande (complètement indépendant du C) qui permet de faire effectuer à la machine toutes sortes d'opérations : création et manipulation des fichiers, communication avec différents périphériques, entre utilisateurs, avec le réseau *Internet*, installation, création, compilation et exécution de logiciels, etc.

*Linux* est un système d'exploitation stable, gratuit et libre. Ceux qui, au prix d'un peu de travail, peuvent l'installer sur leur ordinateur personnel, retrouvent un environnement identique à celui du Magistère et de la plupart des laboratoires de recherche et de nombreuses entreprises. Ils ont accès gratuitement à de nombreux logiciels, en particulier scientifiques, de très haute qualité. Son inconvénient est qu'il ne permet pas toujours aisément l'utilisation de certains périphériques ou l'installation de nouveaux logiciels. La documentation sur *Linux* abonde aussi bien dans les librairies que sur Internet. Ce qu'on en apprendra, le strict nécessaire pour nos besoins, n'est qu'une toute petite partie de ses possibilités.

Pour l'écriture des programmes il est proposé d'utiliser l'éditeur<sup>1</sup> *emacs*. D'autres éditeurs pourront être installés si nécessaire.

## 3 Le langage C

*C is not just another programming language, it's the lingua franca of programming and the bedrock of modern computing; most operating systems, networks, web browsers and many other programming languages such as Python are written in C.*

Né au début des années 70, c'est un langage polyvalent et universellement répandu, particulièrement utilisé par les informaticiens.

Ses avantages sont, en particulier les suivants :

- il est à la fois pas trop éloigné du langage de base de l'ordinateur, ce qui permet une communication assez directe et rapide avec lui, et évolué, ce qui permet une communication pas trop laborieuse avec le cerveau humain
- il est aisé de faire interagir les programmes C avec le système d'exploitation
- il est la base du C++
- il existe des compilateurs<sup>2</sup> gratuits d'excellente qualité, ainsi qu'une documentation abondante, gratuite ou bon marché.

---

1. Un éditeur est un logiciel permettant d'écrire du texte brut (par opposition au texte mis en forme par un traitement de texte) dans un fichier.

2. Le compilateur est le programme qui traduit le C en instructions exécutables par la machine.

Exemple des temps d'exécution (en ms) d'un programme test (tri) selon le langage :

Langage	C	Java	PHP	Python
Temps d'exécution	2.7	47.8	89.2	92.

Ses inconvénients :

- pour le calcul scientifique il n'a pas l'élégance et la clarté du Fortran (mais ce dernier n'a pas la même polyvalence et est principalement cantonné aux laboratoires de recherche alors que le C est largement utilisé dans les entreprises et, d'autre part, le C++ permet de compenser les lacunes du C)
- il se prête bien à l'écriture de programmes remarquablement obscurs. Il existe d'ailleurs un concours international annuel du programme le plus incompréhensible.

C'est donc le C qui est enseigné, en vue du C++ étudié en seconde année de Magistère. Quelques éléments mineurs de C++<sup>3</sup> sont cependant introduits dès la première année, quand ils permettent de faire plus simplement que leurs équivalents en C.

Remarque

Dans le domaine scientifique *Maple* et *Mathematica* sont très intéressants pour le calcul formel mais trop lents pour les calculs numériques.

Ce cours s'adresse non seulement à ceux qui n'ont jamais programmé en C, mais aussi à ceux qui n'ont jamais programmé du tout. Seuls des rudiments y sont présentés. Des notions importantes sont à peine abordées (préprocesseur, fichiers, chaînes de caractères, structures). De nombreuses subtilités ne sont pas mentionnées. Cependant ce qui est présenté permet déjà de faire des calculs intéressants, en particulier en physique. De plus ce cours voudrait être une base suffisante à partir de laquelle il est aisé d'étendre ses connaissances à l'aide de documents écrits.

## 4 Liste de livres de référence pour Linux et le C

En principe l'essentiel de ce qui est au programme doit se trouver dans le cours et il n'est pas nécessaire d'acheter un livre. Cependant, il peut être très utile d'en consulter un pour vérifier une syntaxe ou un point délicat, obtenir un complément d'explication ou étudier des éléments du C qui ne sont que peu ou pas abordés dans le cours.

Pour débiter :

- Le livre du C premier langage, Claude Delannoy, (s'adresse en particulier à des lecteurs qui ne connaissent aucun langage de programmation mais peut convenir aussi aux autres pour démarrer le C)
- Programmer en langage C, Claude Delannoy, (plus complet que le précédent, reste abordable pour un lecteur commençant l'étude du C)

Pour approfondir les points délicats ou les notions complexes, des livres tendant vers l'exhaustivité mais difficiles pour un débutant :

- La référence du C norme ANSI/ISO, Claude Delannoy
- Le langage C Norme ANSI, Brian W. Kernighan et Denis M. Ritchie, (écrit par les fondateurs du C)
- Le langage C ANSI, Philippe Drix, Dunod

Il existe aussi de très bons sites pour le C sur Internet, voir en particulier : <http://cpp.developpez.com/cours/cpp/> et, plus généralement <http://www.developpez.com/>

Pour *Linux* le cours oral et le polycopié suffisent largement. Pour ceux qui veulent en apprendre plus, plusieurs livres très utiles sont disponibles à la bibliothèque universitaire, tel :

- *Linux in a Nutshell*, J.P. Hekman, etc. O'REILLY

## 5 Les logiciels mis à disposition des étudiants

- compilateurs C, C++
- *Python*, *Matplotlib* (langage interprété, graphisme)
- *Gnuplot* (tracé de courbes et surfaces, traitement de données)

---

3. Signalés explicitement comme tels.

- bibliothèque mathématique *gsl*
- *Xfig*, *Inkscape* (dessin vectoriel)
- *Latex* (traitement de texte, en particulier scientifique)
- *LibreOffice* (analogue à *Office* de *Microsoft*, lancer avec *loffice*)
- *Gimp* (traitement d'images)
- *OpenGL* (animation)

et tous les logiciels courants d'une distribution *Linux*.