**Chapitre 1. Introduction**

# Préambule

## - Qu’est-ce qu’un programme informatique ?

Un programme informatique est une liste d'instructions écrites dans un ou plusieurs fichiers, destiné à être exécuté par l'ordinateur afin de réaliser une ou plusieurs tâche(s), de résoudre un problème, de manipuler des [données.](http://www.linux-france.org/prj/jargonf/D/donneaces.html)

## - Qu’est-ce qu’un langage informatique ?

On appelle langage informatique ou langage de programmation, un langage destiné à décrire l'ensemble des actions consécutives qu'un ordinateur ou plus précisément qu’un processeur doit exécuter.

Le langage utilisé par le processeur, c'est-à-dire les données telles qu'elles lui arrivent, est appelé *langage machine*. Il s'agit d'une suite de 0 et de 1 (du binaire). Toutefois le langage machine n'est pas compréhensible facilement par l'humain.

Ainsi il est plus pratique de trouver un langage intermédiaire, compréhensible par l'homme, qui sera ensuite transformé en langage machine pour être exploitable par le processeur.

# Historique

Langage C a été créé en 1972 par Dennie Ritchie aux Laboratoires Bell/AT&T avec un objectif de développer une version portable du système d'exploitation UNIX.

Il provient de deux langages : BPCL développé en 1967 par Martin Richards et B développé en 1970 chez AT&T par Ken Thompson.

Il fut limité à l’usage interne de Bell jusqu’en 1978, date à laquelle Brian Kernighan et Dennie Ritchie publièrent la définition classique du langage C (connue sous le nom de *standard* ***K&R-C***) dans un livre intitulé « The C Programming Language ».

Le succès des années qui suivaient et le développement de compilateurs C par d'autres maisons ont rendu nécessaire la définition d'un standard actualisé et plus précis. En 1983, l'American National Standards Institute' (ANSI) chargeait une commission de mettre au point une définition explicite et indépendante de la machine pour le langage C. Le résultat était le *standard ANSI-C*. La seconde édition du livre « The C Programming Language », parue en 1988, respecte tout à fait le standard ANSI-C et elle est devenue par la suite, la 'bible' des programmeurs en C.

# Intérêts du langage C

* polyvalent : il permet le développement de systèmes d'exploitation, de programmes applicatifs scientifiques et de gestion.
* Il a donné naissance à de nombreux langages dérivés, comme le [C++,](http://fr.wikipedia.org/wiki/C%2B%2B) l['Objective C,](http://fr.wikipedia.org/wiki/Objective_C) le [Java](http://fr.wikipedia.org/wiki/Java_%28langage%29) et le [C#](http://fr.wikipedia.org/wiki/C_sharp).
* Vaste ensemble de bibliothèques prédéfinies, fournies avec le compilateur.
* Portabilité : En respectant la norme ANSI-C, il est possible d'utiliser le même programme sur tout autre système (autre hardware, autre système d'exploitation), simplement en le recompilant.
* Permet un accès à toutes les ressources de la machine (Mémoire, processeur, périphérique, etc.).
* Le C a une grande popularité. Vous trouverez très facilement de l'aide sur celui-ci (forums, sites, newsgroups ...).
* Etc.

# 4. De l’édition à l’exécution

Les étapes menant de l’édition à l’exécution d’un programme en C sont : l’édition, la compilation et l’édition de liens.

## – L’édition du programme

L’édition, c’est la rédaction du programme à l’aide de l’éditeur de texte de CodeBlocks ou d’un autre traitement de texte : on parle alors de « programme source ».

En générale, ce texte sera conservé dans un fichier que l’on nommera « fichier source » dont l’extension est « **.C** ».

## – La compilation

Elle consiste à traduire le programme source en langage machine, en faisant appel à un programme nommé compilateur. Le fichier généré, appelé fichier objet, possède le même nom que le fichier source, mais son extension est « **.OBJ** ».

## – L’édition de liens

Il permet d'intégrer dans le fichier final tous les éléments annexes (fonctions ou librairies) auquel le programme fait référence mais qui ne sont pas stockés dans le fichier source. Le fichier généré est un **fichier exécutable** qui contient tout ce dont il a besoin pour fonctionner de façon autonome.

Ces étapes sont illustrées dans la figure suivante :



**exécution**

**éditeur de liens**

**compilateur**

Résultat

fichier exécutable

fichier objet

fichier source

**Exercice** : Editer, compiler et exécuter le programme suivant: #include <stdio.h>

void main()

{

printf("Soyez les bienvenues dans le monde du développement informatique !");

}