



UNIVERSITÉ  
DE LORRAINE

# Language C

---

Télécom Nancy

Omar CHIDA

# Chapitre 1

## 1. Introduction

1.1 A propos de moi

1.2 Organisation

1.3 L'objectif du Tutorat

1.4 À propos de C

1.5 Motivation : Pourquoi apprendre le C en 2021?

## 2. Compilation

## 3. La langage C

## 4. Les outils

## 5. Dark Frames

## A propos de moi

- Premier ligne de code à l'âge de 14 ans.
- Hardline C++ Fanboy : 6 ans de C/C++.
- De nombreux projets dont un moteur de rendu, une application mobile entre autres codés C/C++.



# Organisation

*Comment ca va se passer ?*

- Cours, exercices, solutions et projets seront sur [Github](#).
- Serveur [Discord](#) dédié pour les questions, aide et autre.
- TD, TP et Projets seront en présentiel.
- N'hésitez pas à m'interrompre à tout moment pour poser des questions.

# L'objectif du Tutorat

- Vous familiariser avec la Langage C.
- Connaître les bonnes pratiques de programmation en C.
- Réussir les examens mais ça va aussi plus loin que ça.
- Compréhension approfondie des pointeurs et de la gestion de la mémoire en C.
- Bien comprendre l'outillage (Compilateur, Débogueur, autre).

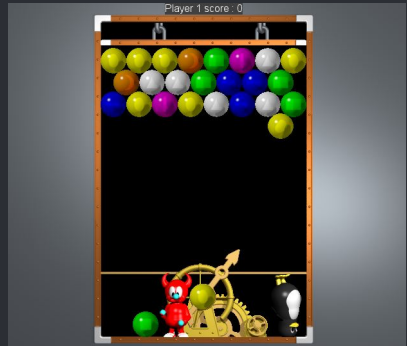
# L'objectif du Tutorat

*Ce que vous pourrez faire à la fin*

```
#include <stdio.h>

int main()
{
    printf("Hello world!\n");
    return 0;
}
```

```
omar@Omar:~$ gcc main.c
omar@Omar:~$ ./a.out
Hello world!
omar@Omar:~$ |
```



# L'objectif du Tutorat

*Ce que nous allons faire ensemble*

- Plein d'exercices (même style que les TD).
  - Exercices liés aux structures de données.
  - Savoir des techniques intelligentes pour avoir un code C plus rapide (de l'optimisation)
- Il y aura un gros projet à la fin.
  - Un jeu vidéo du style (Puzzle Bobble ou Mario).
  - Jeu sur le terminal (style Snake).
  - Émulateur de processeur ARM.
  - Quelque chose de plus simple que ça ? (n'hésitez pas à déposer vos idées).

# Chapitre 2

## 1. Introduction

## 2. Compilation

2.1 Phase 1 : Preprocessing

2.2 Phase 2 : Compiling

2.3 Phase 3 : Assembly

2.4 Phase 4 : Linking

## 3. La langage C

## 4. Les outils

## 5. Dark Frames



# Chapitre 3

1. Introduction

2. Compilation

**3. La langage C**

3.1 Les bases

3.2 Les structs

3.3 Les enums

3.4 Les pointeurs

3.5 Le keyword static

4. Les outils

5. Dark Frames

# Chapitre 4

1. Introduction

2. Compilation

3. La langage C

**4. Les outils**

4.1 Compilateur : GCC/Clang

4.2 Débogueur : GDB

4.3 Valgrind

5. Dark Frames

# Chapitre 5

1. Introduction

2. Compilation

3. La langage C

4. Les outils

**5. Dark Frames**

5.1 Blind Text

5.2 Structuring Elements

5.3 Numerals and Mathematics

5.4 Figures and Code Listings

5.5 Citations and Bibliography

# Jabberwocky

*Lewis Carroll*

'Twas brillig, and the slithy toves  
Did gyre and gimble in the wabe;  
All mimsy were the borogoves,  
And the mome raths outgrabe.

"Beware the Jabberwock, my son!  
The jaws that bite, the claws that catch!  
Beware the Jubjub bird, and shun  
The frumious Bandersnatch!"



## Lists and locales

*Lorem ipsum dolor sit amet*

- Nulla nec lacinia odio.  
Curabitur urna tellus.
  - Fusce id sodales dolor. Sed  
id metus dui.
    - » Cupio virtus licet mi vel  
feugiat.
- 1. Donec porta, risus porttitor  
egestas scelerisque video.
  - 1.1 Nunc non ante fringilla,  
manus potentis cario.
    - 1.1.1 Pellentesque servus  
morbi tristique.

Nechť již hříšné saxofony d'áblů rozzvučí síň úděsnými tóny waltzu, tanga a quickstepu ! Nezvyčajné krdle šťastných figliarskych d'atľov učia pri kótovanom ústí Váhu mĺkveho koňa Waldemara obžierať väčšie kusy exkluzívnej kôry. The quick, brown fox jumps over a lazy dog. DJs flock by when MTV ax quiz prog. "Now fax quiz Jack !"

## Text blocks

*In plain, example, and **alert** flavour*

**This text** is highlighted.

A plain block

This is a plain block containing some **highlighted text**.

An example block

This is an example block containing some **highlighted text**.

An alert block

This is an alert block containing some **highlighted text**.

# Definitions, theorems, and proofs

*All integers divide zero*

## Definition

$$\forall a, b \in \mathbb{Z} : a \mid b \iff \exists c \in \mathbb{Z} : a \cdot c = b$$

## Theorem

$$\forall a \in \mathbb{Z} : a \mid 0$$

## Proof

$$\forall a \in \mathbb{Z} : a \cdot 0 = 0$$



# Numerals and Mathematics

*Formulae, equations, and expressions*

$$1234567890 \quad 1234567890 \quad \hat{x}, \check{x}, \tilde{a}, \bar{a}, \dot{y}, \ddot{y} \iiint f(x, y, z) \, dx dy dz$$

$$\frac{1}{1 + \frac{1}{2 + \frac{1}{3 + x}}} + \frac{1}{1 + \frac{1}{2 + \frac{1}{3 + x}}}$$

$$F : \begin{vmatrix} F''_{xx} & F''_{xy} & F'_x \\ F''_{yx} & F''_{yy} & F'_y \\ F'_x & F'_y & 0 \end{vmatrix} = 0$$

$$\iint_{\mathbf{x} \in \mathbb{R}^2} \langle \mathbf{x}, \mathbf{y} \rangle \, d\mathbf{x}$$

$$\overline{\overline{a\alpha^2 + \underline{b\beta} + \overline{\overline{d\delta}}}}$$

$$]0,1[ + \lceil x \rceil - \langle x, y \rangle$$

$$e^x \approx 1 + x + x^2/2! + x^3/3! + x^4/4!$$

$$\binom{n+1}{k} = \binom{n}{k} + \binom{n}{k-1}$$



## Figures

*Tables, graphs, and images*

Faculty	With T <sub>E</sub> X	Total	%
Faculty of Informatics	1 716	2 904	59.09
Faculty of Science	786	5 275	14.90
Faculty of Economics and Administration	64	4 591	1.39
Faculty of Arts	69	10 000	0.69
Faculty of Medicine	8	2 014	0.40
Faculty of Law	15	4 824	0.31
Faculty of Education	19	8 219	0.23
Faculty of Social Studies	12	5 599	0.21
Faculty of Sports Studies	3	2 062	0.15

Table – The distribution of theses written using T<sub>E</sub>X during 2010–15 at MU

# Figures

*Tables, graphs, and images*

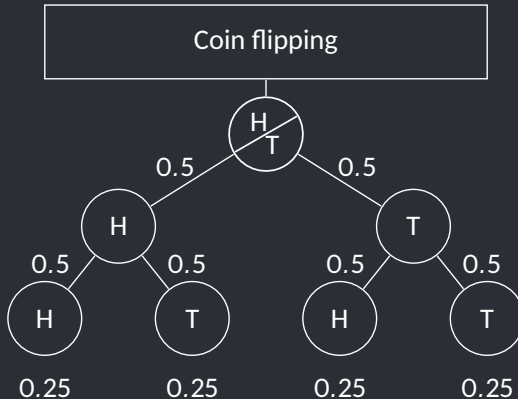


Figure – Tree of probabilities – Flipping a coin<sup>1</sup>

1. A derivative of a diagram from [texample.net](https://texample.net) by cis, CC BY 2.5 licensed

# Code listings

## *An example source code in C*

```
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>

// This is a comment
int main(int argc, char **argv)
{
    while (--c > 1 && !fork());
    sleep(c = atoi(v[c]));
    printf("%d\n", c);
    wait(0);
    return 0;
}
```

# Citations

## $T_{\text{E}}\text{X}$ , $\text{\LaTeX}$ , and Beamer

$T_{\text{E}}\text{X}$  is a programming language for the typesetting of documents. It was created by Donald Erwin Knuth in the late 1970s and it is documented in *The  $T_{\text{E}}\text{X}$ book* [1].

In the early 1980s, Leslie Lamport created the initial version of  $\text{\LaTeX}$ , a high-level language on top of  $T_{\text{E}}\text{X}$ , which is documented in  *$\text{\LaTeX}$  : A Document Preparation System* [2]. There exists a healthy ecosystem of packages that extend the base functionality of  $\text{\LaTeX}$ ; *The  $\text{\LaTeX}$  Companion* [3] acts as a guide through the ecosystem.

In 2003, Till Tantau created the initial version of Beamer, a  $\text{\LaTeX}$  package for the creation of presentations. Beamer is documented in the *User's Guide to the Beamer Class* [4].

# Bibliography

$T_{\text{E}}\text{X}$ ,  $\text{\LaTeX}$ , and Beamer

- [1] Donald E. Knuth. *The  $T_{\text{E}}\text{X}$ book*. Addison-Wesley, 1984.
- [2] Leslie Lamport.  *$\text{\LaTeX}$  : A Document Preparation System*. Addison-Wesley, 1986.
- [3] M. Goossens, F. Mittelbach, and A. Samarin. *The  $\text{\LaTeX}$  Companion*. Addison-Wesley, 1994.
- [4] Till Tantau. *User's Guide to the Beamer Class Version 3.01*. Available at <http://latex-beamer.sourceforge.net>.
- [5] A. Mertz and W. Slough. Edited by B. Beeton and K. Berry. *Beamer by example* In TUGboat, Vol. 26, No. 1., pp. 68-73.