1337 coding school
Mail central, Bâtiment B
25000 Khouribga,
MOROCCO

## ACADEMIC RESULTS FOR ISMAIL BENAAITONA

I, the undersigned Larbi EL HILALI, Managing Director of 1337 coding school located at Mail central, Bâtiment B 25000 Khouribga, Morocco, hereby certify that:

**Ismail Benaaitona, born on February 23, 1998 in souk sebt (Morocco)**

obtained the grades detailed below as of February 16, 2025.

This certificate is delivered upon request for all legal intents and purposes.

***Selected in:*** **August 2022**
***Curriculum started on:*** **October 05, 2022**
***Curriculum ended on:*** **-**

The progression of the student inside the curriculum is represented by its level, over 21.
***The current level of the student is:*** **11.42.**

The 42 curriculum is divided into two halves: the common core and the 42 advanced part. Once students complete the first half (the common core), they have the option to either continue their journey in the 42 advanced part, or conclude their progression and become an alumni at any point during this second part.
***The current situation of the student is:*** **in the 42 advanced part.**
See details below.

Made in Khouribga, on February 16, 2025

## DETAILS

Here is a description of each part of the curriculum and the current position of the student:

### The Common Core

The common core of the 42 curriculum represents the minimum set of skills to be ready for a first professional experience. It provides basic and standard coding skills, as well as a fruitful range of soft skills.  The delay of the CC is approximately between 1 and 2 years. The following information represent the skills developed during this part of the curriculum and the current progression of the student:

***Ismail Benaaitona : Common core achieved at:*** **100%.**

Developed skills during the entire common core:

- ***Algorithms & AI:*** Standards algorithms on standards structures: searching, sorting, insertion, deletion, balance, on: arrays, linked lists, trees. State machine and asynchronous management.

- ***Graphics:*** Image management, RGB structure of an image, manipulating areas, drawing into an image, interacting with the window management system and getting user events and inputs from keyboard and mouse, programming with callbacks and event loop.

- ***Group & interpersonal:*** Collaboration, relationships and group management situations, including different kinds of interactions between people (friendly, tensions ...)

- ***Imperative programming:*** Basics of coding in C : the C syntax, variable, loops, conditional branches, functions, recursivity, instructions, calculus and expressions, comparisons operators, standard and advanced types, strings processing, structures, includes and libraries, memory allocation and release, linked lists, trees, the C standard library

- ***Network & system administration:*** Basics of computer networking : IP addresses, subnets, default routing, local network structure, host to host connectivity to network services; Basics of system administration : operating system installation with Linux, setting up security, access, users, storage, installing network services like mail, dns, web server, ...

- ***Object-oriented programming:*** Object programming principles in C++, classes, namespaces, constructors and destructors, memory management in C++, inheritance, abstraction, overloading, templates, standard C++ library types and tools

- ***Rigor:*** The need to fulfill administrative and technical constraints. The need for a wide and deep testing process to eliminate failure.

- ***System programming:*** Classic Unix system interactions : system calls, filesystem access and management, process creation, execution, management; inter-process communications : pipes and signals; device management and ioctl, terminal capabilities; network communication : TCP & UDP sockets, DNS resolution, endianness

- ***Web:*** The client-server architecture involved in the web, role and actions of the web server, role and actions of the web browser; The HTTP protocol; Web technologies involved : HTML, CSS, Javascript, images and videos; Backend language and framework for dynamic websites: one among php, ruby, python, go, javascript, Rails, Symfony, Django, Node, ... ; MVC model; users web services : web sessions, authentification, cookies, search, caddie, backoffice configuration, ... ; Basics of user experience, user interface, and design.

Details of each validated project in appendix 1.

## The 42 Advanced Part

The 42 Advanced offers a choice of path among various ICT specialisations: each student can select the topic(s) she/he wants to develop and improve. This part of the curriculum also contains several professional experiences (internships, part-time jobs, ...).

No projects completed yet

Professional experience: no professional experience yet

Details of the validated projects in appendix 2.

## APPENDIX 1

Projects covered during the common core:

| Name | Estimated workload | Result | Associated skills | Validation date |
|---|---|---|---|---|
| Libft | 70H | Pass with bonus | Rigor, Algorithms & AI, Imperative programming | November 08, 2022 |
| get_next_line | 70H | Pass with bonus | Rigor, Algorithms & AI, Unix | November 17, 2022 |
| ft_printf | 70H | Pass | Rigor, Algorithms & AI | November 27, 2022 |
| Born2beroot | 40H | Pass with bonus | Rigor, Network & system administration | December 30, 2022 |
| push_swap | 60H | Pass with bonus | Rigor, Algorithms & AI, Unix, Imperative programming | February 08, 2023 |
| fract-ol | 60H | Pass with bonus | Imperative programming, Graphics | March 08, 2023 |
| pipex | 50H | Pass with bonus | Unix, Imperative programming | April 06, 2023 |
| Exam Rank 02 | 0H | Pass | | May 04, 2023 |
| Philosophers | 70H | Pass with bonus | Rigor, Unix, Imperative programming | May 29, 2023 |
| minishell | 210H | Pass with bonus | Rigor, Unix, Imperative programming | August 12, 2023 |
| Exam Rank 03 | 0H | Pass | | August 23, 2023 |
| NetPractice | 50H | Pass | Rigor, Network & system administration | August 29, 2023 |
| cub3d | 280H | Pass with bonus | Rigor, Algorithms & AI, Imperative programming, Graphics | October 08, 2023 |
| CPP Module 00 | 22H | Pass | Rigor, Object-oriented programming, Imperative programming | October 23, 2023 |
| CPP Module 01 | 12H | Pass | Rigor, Object-oriented programming, Imperative programming | November 04, 2023 |
| CPP Module 02 | 12H | Pass | Rigor, Object-oriented programming, Imperative programming | November 11, 2023 |
| CPP Module 03 | 12H | Pass | Rigor, Object-oriented programming, Imperative programming | November 18, 2023 |
| CPP Module 04 | 12H | Pass | Rigor, Object-oriented programming, Imperative programming | November 27, 2023 |
| Exam Rank 04 | 0H | Pass | | December 18, 2023 |
| CPP Module 05 | 25H | Pass | Rigor, Object-oriented programming, Imperative programming | December 20, 2023 |
| CPP Module 06 | 25H | Pass | Rigor, Object-oriented programming, Imperative programming | December 22, 2023 |
| CPP Module 07 | 25H | Pass | Rigor, Object-oriented programming, Imperative programming | December 27, 2023 |
| CPP Module 08 | 25H | Pass | Rigor, Object-oriented programming, Imperative programming | January 07, 2024 |

|  |  |  |  |  |
|---|---|---|---|---|
| CPP Module 09 | 40H | Pass | Rigor, Object-oriented programming, Imperative programming | January 14, 2024 |
| Exam Rank 05 | 0H | Pass |  | September 18, 2024 |
| ft_irc | 175H | Pass with bonus | Rigor, Unix, Network & system administration, Object-oriented programming | October 17, 2024 |
| Inception | 210H | Pass | Rigor, Network & system administration | October 25, 2024 |
| ft_transcendence | 245H | Pass with bonus | Rigor, Web, Group & interpersonal | January 21, 2025 |
| Exam Rank 06 | 0H | Pass |  | January 30, 2025 |

## APPENDIX 2

Projects covered during the 42 advanced:

| Name | Estimated workload | Result | Associated skills | Validation date |
|---|---|---|---|---|
| – |  |  |  |  |

**Internship and professional experiences**

| Company name | Duration | Validation | Skills | Validation date |
|---|---|---|---|---|
| – |  |  |  |  |

## APPENDIX 3

Description of each covered project:

| Name | Description |
|---|---|
| Libft | This project is your very first project as a student at 42. You will need to recode a few functions of the C standard library as well as some other utility functions that you will use during your whole cursus. |
| get_next_line | May it be a file, stdin, or even later a network connection, you will always need a way to read content line by line. It is time to start working on this function, which will be essential for your future projects. |
| ft_printf | This project is pretty straightforward, you have to recode printf. You will learn what is and how to implement variadic functions. Once you validate it, you will reuse this function in your future projects. |
| Born2beroot | This project aims to introduce you to the wonderful world of virtualization. |
| push_swap | This project involves sorting data on a stack, with a limited set of instructions, and the smallest number of moves. To make this happen, you will have to manipulate various sorting algorithms and choose the most appropriate solution(s) for optimized data sorting. |
| fract-ol | Discover 2D programming and the psychedelic universe of fractals in this project, using minilibX. |
| pipex | This project aims to deepen your understanding of the two concepts that you already know: Redirections and Pipes. It is an introductory project for the bigger UNIX projects that will appear later on in the cursus. |
| Exam Rank 02 | This project will evaluate your abilities and knowledge about programming. |
| Philosophers | This project aims to teach concurrent programming, focusing on multithreading and multiprocessing. |
| minishell | The objective of this project is for you to create a simple shell. |
| Exam Rank 03 |  |
| NetPractice | NetPractice is a general practical exercise to let you discover networking. |
| cub3d | This project is inspired by the world-famous eponymous 90's game, which was the first FPS ever. It will enable you to explore ray-casting. Your goal will be to make a dynamic view inside a maze, in which you'll have to find your way. |

| | |
|---|---|
| CPP Module 00 | This first module of C++ is designed to help you understand the specifities of the language when compared to C. Time to dive into Object Oriented Programming! |
| CPP Module 01 | This module is designed to help you understand the memory allocation, reference, pointers to members and the usage of the switch in CPP. |
| CPP Module 02 | This module is designed to help you understand Ad-hoc polymorphism, overloads and orthodox canonical classes in CPP. |
| CPP Module 03 | This module is designed to help you understand Inheritance in CPP. |
| CPP Module 04 | This module is designed to help you understand Subtype polymorphism, abstract classes and interfaces in CPP. |
| Exam Rank 04 | |
| CPP Module 05 | This module is designed to help you understand Try/Catch and Exceptions in CPP. |
| CPP Module 06 | This module is designed to help you understand the different casts in CPP. |
| CPP Module 07 | This module is designed to help you understand Templates in CPP. |
| CPP Module 08 | This module is designed to help you understand templated containers, iterators and algorithms in CPP. |
| CPP Module 09 | This module is designed to help you understand the containers in CPP. |
| Exam Rank 05 | |
| ft_irc | Create your own IRC server in C++, fully compatible with an official client. |
| Inception | This project aims to broaden your knowledge of system administration by using Docker. You will virtualize several Docker images, creating them in your new personal virtual machine. |
| ft_transcendence | This project is centered around the design, development, and organization of a full-stack web application. |
| Exam Rank 06 | |