Association 42
96, Boulevard Bessières
75017 Paris
FRANCE

ACADEMIC RESULTS FOR WOOLIM PARK

I, the undersigned Grégoire MARTINEZ, Director of 42 Paris located at 96, Boulevard Bessières, 75017 Paris, FRANCE, hereby certify that:

Woolim Park, born on May 17, 1991 in Gwangmyeong (South Korea)

obtained the grades detailed below as of January 11, 2026.

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

Selected in: **July 2019**
Curriculum started on: **October 07, 2019**
Curriculum ended on: **January 04, 2026**

Founded in 2013, 42 is a worldwide network of ICT schools. We are a non-traditional educator offering high-quality and scalable software engineering education to anyone who wants to learn.
It is our mission to prepare the next generation for the jobs of today and tomorrow. We do so using an innovative educational model, which relies on peer-to-peer learning, project-based and hands-on approach to programming. Our innovative model, allowing individual pace and path, has proven that our students become industry-ready software engineers within 2 to 5 years.

The progression of the student inside the curriculum is represented by its level, over 21.
The current level of the student is: **21.04.**

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: **alumni.**
See details below.

Made in Paris, on January 11, 2026

G. Martinez

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:

Woolim Park : 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: 2 Internships

Details of the validated projects in appendix 2.

SPECIAL

A student can eventually benefit from special programs or projects valuable for their personal skill set, and thus included in their curriculum. They are mentioned here:

Name	Equivalent workload
–	

APPENDIX 1

Projects covered during the common core:

Name	Estimated workload	Result	Associated skills	Validation date
Libft	70H	Pass with bonus	Rigor, Imperative programming, Algorithms & AI	October 15, 2019
get_next_line	55H	Pass with bonus	Rigor, Unix, Algorithms & AI	October 28, 2019
ft_printf	55H	Pass	Rigor, Algorithms & AI	November 09, 2019
ft_server	84H	Pass	Rigor, Network & system administration	November 25, 2019
cub3d	280H	Pass	Rigor, Imperative programming, Algorithms & AI, Graphics	December 04, 2019
Exam Rank 02	0H	Pass		December 19, 2019
ft_services	210H	Pass	Rigor, Network & system administration	February 11, 2020
Exam Rank 03	0H	Pass		February 21, 2020
minishell	210H	Pass	Rigor, Imperative programming, Unix	March 10, 2020
Philosophers	70H	Pass	Rigor, Imperative programming, Unix	March 13, 2020
CPP Module 00	22H	Pass	Object-oriented programming, Rigor, Imperative programming	April 13, 2020
CPP Module 01	12H	Pass	Object-oriented programming, Rigor, Imperative programming	April 13, 2020
CPP Module 02	12H	Pass	Object-oriented programming, Rigor, Imperative programming	April 22, 2020
CPP Module 03	12H	Pass	Object-oriented programming, Rigor, Imperative programming	April 25, 2020
CPP Module 04	12H	Pass	Object-oriented programming, Rigor, Imperative programming	April 26, 2020
CPP Module 05	25H	Pass	Object-oriented programming, Rigor, Imperative programming	April 27, 2020

CPP Module 06	25H	Pass	Object-oriented programming, Rigor, Imperative programming	April 27, 2020
CPP Module 07	25H	Pass	Object-oriented programming, Rigor, Imperative programming	April 28, 2020
CPP Module 08	25H	Pass	Object-oriented programming, Rigor, Imperative programming	April 29, 2020
ft_containers	140H	Pass with bonus	Object-oriented programming, Rigor	May 14, 2020
webserv	175H	Pass	Object-oriented programming, Rigor, Unix, Network & system administration	June 26, 2020
Exam Rank 04	0H	Pass		July 17, 2020
Exam Rank 05	0H	Pass		July 30, 2020
Exam Rank 06	0H	Failed		August 07, 2020
ft_transcendence	245H	Pass	Rigor, Web, Group & interpersonal	September 14, 2020

APPENDIX 2

Projects covered during the 42 advanced:

Name	Estimated workload	Result	Associated skills	Validation date
netwhat	28H	Pass	Network & system administration	October 24, 2019
libasm	20H	Pass	Rigor, Imperative programming	January 27, 2020
darkly	98H	Pass with bonus	Web, Adaptation & creativity, Security	October 05, 2020
ft_linear_regression	70H	Pass with bonus	Rigor, Algorithms & AI, DB & Data	November 26, 2020
red-tetris	147H	Pass with bonus	Object-oriented programming, Functional programming, Web, Technology integration	December 08, 2020
matcha	98H	Pass with bonus	Web, DB & Data, Security	March 10, 2021
Python - 0 - Starting	7H	Pass	Object-oriented programming, Rigor, Algorithms & AI	February 01, 2024
Python - 1 - Array	7H	Pass	Object-oriented programming, Rigor, Algorithms & AI	February 02, 2024
Python - 2 - DataTable	7H	Pass	Object-oriented programming, Rigor, Algorithms & AI	February 02, 2024
Python - 3 - OOP	7H	Pass	Object-oriented programming, Rigor, Algorithms & AI	February 03, 2024
Python - 4 - Dod	7H	Pass	Object-oriented programming, Rigor, Algorithms & AI	February 04, 2024
Python for Data Science	35H	Pass	Object-oriented programming, Rigor, Algorithms & AI	February 04, 2024
ready set boole	110H	Pass	Rigor, Algorithms & AI, Adaptation & creativity	February 12, 2024
hypertube	196H	Pass with bonus	Web, DB & Data, Group & interpersonal	March 09, 2024
dslr	98H	Pass with bonus	Algorithms & AI, DB & Data	April 04, 2024
Tokenizer	98H	Pass	Rigor, Adaptation & creativity, Technology integration	July 04, 2024
matrix	110H	Pass with bonus	Rigor, Algorithms & AI, Adaptation & creativity	July 23, 2024
Leaffliction	294H	Pass	Rigor, Algorithms & AI, Group & interpersonal	September 05, 2024
multilayer-perceptron	98H	Pass with bonus	Rigor, Algorithms & AI, DB & Data	September 06, 2024

cloud-1	100H	Pass	Web, DB & Data, Network & system administration, Technology integration	October 13, 2024
TokenizeArt	98H	Pass	Rigor, Adaptation & creativity, Technology integration	February 11, 2025
OCAML - Basic syntax and semantics - 0	7H	Pass	Rigor, Functional programming, Adaptation & creativity	February 14, 2025
computorv1	49H	Pass with bonus	Rigor, Algorithms & AI	February 14, 2025
OCAML - Recursion and higher-order functions - 0	7H	Pass	Rigor, Functional programming, Adaptation & creativity	February 15, 2025
OCAML - Pattern Matching and Data Types - 0	7H	Pass	Rigor, Functional programming, Adaptation & creativity	April 01, 2025
ft_turing	98H	Pass with bonus	Rigor, Imperative programming, Functional programming, Algorithms & AI	April 02, 2025
OCAML - OCaml's modules language - 1	7H	Pass	Rigor, Functional programming, Adaptation & creativity	April 02, 2025
OCAML - Imperative features - 1	7H	Pass	Rigor, Functional programming, Adaptation & creativity	April 03, 2025
OCAML - Functor - 1	7H	Pass	Rigor, Functional programming, Adaptation & creativity	April 30, 2025
OCAML - Object Oriented Programming - 1	7H	Pass	Rigor, Functional programming, Adaptation & creativity	April 30, 2025
Inception-of-Things	200H	Pass	Rigor, Network & system administration	May 01, 2025
OCAML - Object Oriented Programming - 2	7H	Pass	Rigor, Functional programming, Adaptation & creativity	May 01, 2025
OCAML - Monoids and Monads - 3	7H	Pass	Rigor, Functional programming, Adaptation & creativity	May 02, 2025
gomoku	196H	Pass with bonus	Rigor, Algorithms & AI, Group & interpersonal	May 04, 2025
Piscine ocaml	63H	Pass	Rigor, Functional programming, Adaptation & creativity	May 06, 2025

Internship and professional experiences

Company name	Duration	Validation	Skills	Validation date
école42	6 months	Pass with bonus	Company experience, Group & interpersonal	May 07, 2021
Luko Assurance (Demain ES)	6 months	Pass with bonus	Company experience, Group & interpersonal	January 27, 2024

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 from the C standard library, as well as some other utility functions that you will use throughout your whole curriculum.
netwhat	Netwhat will allow you to discover the network and to understand how it works. This will allow you to understand how some things you already use in your everyday life, without even knowing it. For that you will answer a questionnaire on the website netwhat.42.fr. If you unregister this project, you will need to wait three days before you can retry the subject.
get_next_line	Whether it's a file, stdin, or even later a network connection, you'll always need a way to read content line by line. It's 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.
ft_server	This project is intended to introduce you to the basics of system and network administration. It will allow you to install a

	complete web server, using a deployment technology named Docker.
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.
Exam Rank 02	
libasm	The aim of this project is to become familiar with assembly language.
ft_services	This project consist to clusturing an docker-compose application and deploy it with Kubernetes.
Exam Rank 03	advanced string manipulation, backtracking
minishell	The objective of this project is for you to create a simple shell.
Philosophers	This project aims to teach concurrent programming, focusing on multithreading and multiprocessing.
CPP Module 00	This first module of C++ is designed to help you understand the specificities 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 memory allocation, references, pointers to members, and the usage of the switch statement in C++.
CPP Module 02	This module is designed to help you understand ad-hoc polymorphism, function overloading, and orthodox canonical classes in C++.
CPP Module 03	This module is designed to help you understand inheritance in C++.
CPP Module 04	This module is designed to help you understand subtype polymorphism, abstract classes, and interfaces in C++.
CPP Module 05	This module is designed to help you understand try/catch and exceptions in C++.
CPP Module 06	This module is designed to help you understand the different types of casting in C++.
CPP Module 07	This module is designed to help you understand templates in C++.
CPP Module 08	This module is designed to help you understand templated containers, iterators, and algorithms in C++.
ft_containers	The multiple available containers in C++ all have a very different usage. To make sure you understand them all, let's re-implement them!
webserv	This project aims to create your own HTTP server. You will be able to test it with a real web browser. HTTP is one of the most used protocols on the internet. Knowing its intricacies will be useful, even if web development is not on your carreer path.
Exam Rank 04	
Exam Rank 05	
ft_transcendence	Design, develop, and organize a full-stack web application with complete creative freedom. Choose your project concept, select from a wide range of technical modules, and make key architectural decisions. This highly flexible project allows you to explore modern web development while demonstrating your technical skills and creativity through a modular approach.
darkly	Introductory project to computer security in the specific field of the web, this project will have you dissect a vulnerable website. In doing so, you will develop your own approach to thinking about security in a web application and become aware of issues related to simple development errors, both from a programming and a design perspective.
ft_linear_regression	This project will be your first steps into AI and Machine Learning. You're going to start with a simple, basic machine learning algorithm. You will have to create a program that predicts the price of a car by using a linear function train with a gradient descent algorithm.
red-tetris	The objective of this project is to develop a networked multiplayer Tetris game using a software stack that is exclusively Full Stack JavaScript.
matcha	This second project introduces a more advanced tool for building web applications: the micro-framework. You are invited to create a dating site in the programming language of your choice. User interaction is at the heart of this project!
Work Experience I	Your first step in a company is an important milestone in your 42 training. This employment project is designed to help you discover the professional world and apply your work ethic and adaptability in a real-world context. It represents a first asset for your resume and offers an opportunity to identify your future field of interest.
Work Experience II	Your course 42 can be continued with a second period of professional integration: the second internship. This internship is the accomplishment of your journey, and will have you demonstrate your expertise professionally in a company. This will be the springboard for your career.
Python - 0 - Starting	Introduction about the basics of the Python Programming Language.
Python - 1 - Array	Discovery of arrays, their manipulations, and work on images.
Python - 2 - DataTable	Load, manipulate and display datatable.
Python - 3 - OOP	Classes and the heritage in Python.
Python - 4 - Dod	Structure Design
Python for Data Science	Piscines are an important time in your cursus, during which you will have the occasion to learn a new language, or even a new paradigm!
ready set boole	Discover the basics of computer-related mathematics with Boolean Algebra and Sets Theory!

hypertube	Hypertube introduces you to a powerful category of tools: MVC frameworks. You will learn how to work with an MVC framework—using the programming language of your choice—to build a video streaming website with content downloaded via the BitTorrent protocol.
dslr	Discover Data Science through this project by recreating the Hogwarts Sorting Hat using logistic regression!
Tokenizer	This project allows you to learn the basics in web3. You will have to create your personal token!
matrix	You've probably encountered vectors and vector spaces before. Now it's time to formalize them through Linear Algebra and to learn how matrices and linear transformations work.
Leaffliction	An innovative computer vision project utilizing leaf image analysis for disease recognition.
multilayer-perceptron	This project is an introduction to artificial neural networks through the implementation of a multilayer perceptron.
cloud-1	This project is an introduction to cloud servers
TokenizeArt	This project allows you to learn the basics in web3. You will have to create your non-fungible token!
OCAML - Basic syntax and semantics - 0	In this first OCaml module, you will discover the basic syntax and semantics of the language: values, types, operators, let bindings, functions and recursion.
computorv1	The goal of this project is to get acquainted with handling elementary math tools that may be helpful for other 42 projects. You will not "do math for doing math", but to develop a progressive and relaxed approach to projects where these tools are needed. You can choose the language of your choice for this subject.
OCAML - Recursion and higher-order functions - 0	In this second OCaml module, you will discover the recurion and the higher-order functions.
OCAML - Pattern Matching and Data Types - 0	The main theme of this module is the pattern matching usages and the manipulation of the many constructed types available in OCaml.
ft_turing	ft_turing will help you discover the turing model using imperative programming through an implementation of this model with OCaml
OCAML - OCaml's modules language - 1	The main theme of this module is related to the ocaml modules.
OCAML - Imperative features - 1	The main theme of this module is related to features linked to imperative programming in ocaml.
OCAML - Functor - 1	The main theme of this module is related to the functor in ocaml.
OCAML - Object Oriented Programming - 1	The main theme of this module is to introduce the object oriented programming style with OCaml.
Inception-of-Things	This project aims to introduce you to Kubernetes from a developer's perspective. You will have to set up small clusters and discover the mechanics of continuous integration. At the end of this project, you will be able to set up a working cluster in Docker and have a usable continuous integration pipeline for your applications.
OCAML - Object Oriented Programming - 2	The main theme of this module is to introduce the object oriented programming style with OCaml.
OCAML - Monoids and Monads - 3	The main theme of this module is to introduce the Monoids and the Monads. This is hard but not as hard as it seems.
gomoku	This project involves creating, in the language of your choice, a Gomoku game integrating an AI player capable of beating a human player the fastest way possible. To do this, you will implement a min-max algorithm but also do research, trial and error to find the most adapted heuristics. This will not be as easy as checkers.
Piscine ocaml	Piscines are an important time in your cursus, during which you will have the occasion to learn a new language, or even a new paradigm!