Association 42
Muallimköy Mahallesi, Deniz Cd. No:143-5,
41400 Gebze / Kocaeli
TÜRKİYE

ACADEMIC RESULTS FOR MEHMET ALI YILDIRIM

I, the undersigned Meltem Yıldırımış, Managing Director of 42 Kocaeli located at Muallimköy Mahallesi, Deniz Cd. No:143-5, 41400 Gebze / Kocaeli TÜRKİYE, hereby certify that:

Mehmet Ali Yıldırım, born on May 02, 1997 in Adana (Turkey)

obtained the grades detailed below as of June 14, 2024.

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

Selected in: **February 2023**
Curriculum started on: **July 03, 2023**
Curriculum ended on: **-**

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: **8.06.**

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 Common Core.**
See details below.

Made in Kocaeli, on June 14, 2024

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:

***Mehmet Ali Yıldırım* : Common core achieved at: 57%.**

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.

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
Exam Rank 04	0H	Pass		May 17, 2024
CPP Module 00	22H	Pass	Rigor, Object-oriented programming, Imperative programming	March 26, 2024
CPP Module 01	12H	Pass	Rigor, Object-oriented programming, Imperative programming	April 29, 2024
CPP Module 02	12H	Pass	Rigor, Object-oriented programming, Imperative programming	May 10, 2024
CPP Module 03	12H	Pass	Rigor, Object-oriented programming, Imperative programming	May 10, 2024
CPP Module 05	25H	Pass	Rigor, Object-oriented programming, Imperative programming	May 28, 2024
CPP Module 06	25H	Pass	Rigor, Object-oriented programming, Imperative programming	June 01, 2024
CPP Module 07	25H	Pass	Rigor, Object-oriented programming, Imperative programming	June 02, 2024
CPP Module 08	25H	Pass	Rigor, Object-oriented programming, Imperative programming	June 05, 2024
Libft	70H	Pass with bonus	Algorithms & AI, Rigor, Imperative programming	July 11, 2023
Born2beroot	40H	Pass with bonus	Rigor, Network & system administration	August 09, 2023
ft_printf	70H	Pass	Algorithms & AI, Rigor	July 18, 2023
get_next_line	70H	Pass with bonus	Algorithms & AI, Rigor, Unix	July 29, 2023
so_long	60H	Pass	Imperative programming, Graphics	October 08, 2023
minitalk	50H	Pass	Rigor, Unix	August 31, 2023

push_swap	60H	Pass with bonus	Algorithms & AI, Rigor, Unix, Imperative programming	October 27, 2023
minishell	210H	Pass with bonus	Rigor, Unix, Imperative programming	March 24, 2024
NetPractice	50H	Pass	Rigor, Network & system administration	April 27, 2024
Philosophers	70H	Pass	Rigor, Unix, Imperative programming	November 14, 2023
cub3d	280H	Pass	Algorithms & AI, Rigor, Imperative programming, Graphics	March 28, 2024
CPP Module 04	12H	Pass	Rigor, Object-oriented programming, Imperative programming	May 14, 2024
Inception	210H	Pass	Rigor, Network & system administration	May 24, 2024
CPP Module 09	40H	Pass	Rigor, Object-oriented programming, Imperative programming	June 09, 2024
ft_irc	175H	Pass	Rigor, Unix, Network & system administration, Object-oriented programming	June 13, 2024
Exam Rank 02	0H	Pass		August 21, 2023
Exam Rank 03	0H	Pass		November 03, 2023

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.
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.
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.
Born2beroot	This project aims to introduce you to the wonderful world of virtualization.
Exam Rank 02	Exam Rank 02
minitalk	The purpose of this project is to code a small data exchange program using UNIX signals. It is an introductory project for the bigger UNIX projects that will appear later on in the cursus.
so_long	This project is a small 2D game with minilibx. You'll learn about textures, sprites and tiles.
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.

Exam Rank 03	Exam Rank 03
Philosophers	Eat, Sleep, Spaghetti, repeat. This project is about learning how threads work by precisely timing a group of philosophers on when to pick up forks and eat spaghetti without dying from hunger.
minishell	The objective of this project is for you to create a simple shell.
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!
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.
NetPractice	NetPractice is a general practical exercise to let you discover networking.
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	EXAM RANK 04
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.
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.
ft_irc	Create your own IRC server in C++, fully compatible with an official client.