



# C Bootcamp

## 24h Challenge

Staff WeThinkCode\_ [staff@wethinkcode.co.za](mailto:staff@wethinkcode.co.za)

*Summary: THE FOLLOWING TAKES PLACE BETWEEN 4.00 P.M. AND 5.00 P.M.*

# Contents

<b>I</b>	<b>Instructions</b>	<b>2</b>
<b>II</b>	<b>Foreword</b>	<b>4</b>
<b>III</b>	<b>ft_takes_place.c</b>	<b>5</b>

# Chapter I

## Instructions

- Only this page will serve as reference: do not trust rumors.
- Watch out! This document could potentially change up to an hour before submission.
- Make sure you have the appropriate permissions on your files and directories.
- You have to follow the submission procedures for every exercise.
- Your exercises will be checked and graded by your fellow classmates.
- On top of that, your exercises will be checked and graded by a program called Moulinette.
- Moulinette is very meticulous and strict in its evaluation of your work. It is entirely automated and there is no way to negotiate with it. So if you want to avoid bad surprises, be as thorough as possible.
- Moulinette is not very open-minded. It won't try and understand your code if it doesn't respect the Norm. Moulinette relies on a program called **Norminator** to check if your files respect the norm. TL;DR: it would be idiotic to submit a piece of work that doesn't pass **Norminator**'s check.
- These exercises are carefully laid out by order of difficulty - from easiest to hardest. We **will not** take into account a successfully completed harder exercise if an easier one is not perfectly functional.
- Using a forbidden function is considered cheating. Cheaters get -42, and this grade is non-negotiable.
- If `ft_putchar()` is an authorized function, we will compile your code with our `ft_putchar.c`.
- You'll only have to submit a `main()` function if we ask for a program.

- Moulinette compiles with these flags: -Wall -Wextra -Werror, and uses gcc.
- If your program doesn't compile, you'll get 0.
- You cannot leave any additional file in your directory than those specified in the subject.
- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.
- Your reference guide is called `Google / man / the Internet / ....`
- Check out the "C Bootcamp" part of the forum on the intranet.
- Examine the examples thoroughly. They could very well call for details that are not explicitly mentioned in the subject...
- By Odin, by Thor ! Use your brain !!!

# Chapter II

## Foreword

It was an usual evening for Nicolas « Nick » Bauer.

The face still hangovered from the night before, going as best as he could to the bathroom, not without stumbling twice on the stairs and avoiding a deadly fall. Despite a revigorating shower, it was difficult to know which was foggiest of the two: his brain or the bathroom.

Since he was wanted by Russia, China, USA, his life was less spicy. It must be said that when one is on 2 international superpower as well as the first MacDonald's exportator country black list, you need to keep a low profile.

Nick knew what he was talking about when it comes to discretion. His last neighbors, a little too curious, disappeared after a fatale excursion to a shopping mall. When the authorities were looking for clues about the two lovers whereabouts, Nick could peacefully empty his automatic guns on sunday afternoons « to stay sharp ».

In every aspects, this day would end up the most common triviality fashion of Seaside Heights, New Jersey.

« Tut... Tut... »

« My phone, thought Nick. It's not the usual ringtone. It's the harrasment ringtone. »


While answering, Nick looked at the small kitten shaped clock in the kitchen. This trophy was a present from Vic Soontobedead, his soulmate. Vic replaced in his heart his past murdered partner, Renée.

A very strange clock operating on Windows 8, which never worked till now...

# Chapter III

## ft\_takes\_place.c

Even in the future, watches provide the time.  
You just have to know how to set them up.

	Exercise 01
ft_takes_place.c	
Turn-in directory : <i>ex01/</i>	
Files to turn in : <code>ft_takes_place.c</code>	
Allowed functions : <code>write</code> , <code>printf</code>	
Notes : n/a	

- Create a function `ft_takes_place` which displays the time on the standard output like this : THE FOLLOWING TAKES PLACE BETWEEN X.00 A.M. AND Y.00 A.M.. Followed by a line break. Example :

```
THE FOLLOWING TAKES PLACE BETWEEN 10.00 A.M. AND 11.00 A.M.
```

- Here's how it should be prototyped :

```
void ft_takes_place(int hour);
```

- The `hour` argument will be a valid `int`, in a 24-hour clock format. The output will be in the am/pm format.



Be careful with noon and midnight !