

C Piscine C 03

Summary: This document is the subject for the C 03 module of the C Piscine @ 42.

Version: 4.1

Contents

1	Instructions	2
II	Foreword	4
III	Exercise 00 : ft_strcmp	5
IV	Exercise 01 : ft_strncmp	6
V	Exercise 02 : ft_streat	7
VI	Exercise 03 : ft_strncat	8
VII	Exercise 04 : ft_strstr	9
VIII	Exercise 05 : ft_strlcat	10
\mathbf{IX}	Submission and peer-evaluation	11

Chapter I

Instructions

- Only this page serves as your reference, do not trust rumors.
- Watch out! This document may change before submission.
- Ensure you have the appropriate permissions on your files and directories.
- You must follow the **submission procedures** for all your exercises.
- Your exercises will be checked and graded by your fellow classmates.
- Additionally, your exercises will be evaluated by a program called **Moulinette**.
- Moulinette is meticulous and strict in its assessment. It is fully automated, and there is no way to negotiate with it. To avoid unpleasant surprises, be as thorough as possible.
- Moulinette is not open-minded. If your code does not adhere to the Norm, it won't attempt to understand it. Moulinette relies on a program called norminette to check if your files comply with the Norm. TL;DR: Submitting work that doesn't pass norminette's check makes no sense.
- These exercises are arranged in order of difficulty, from easiest to hardest. We will not consider a successfully completed harder exercise if an easier one is not fully functional.
- Using a forbidden function is considered cheating. Cheaters receive a grade of **-42**, which is non-negotiable.
- You only need to submit a main() function if we specifically ask for a program.
- Moulinette compiles with the following flags: -Wall -Wextra -Werror, using cc.
- If your program does not compile, you will receive a grade of **0**.
- You **cannot** leave **any** additional file in your directory beyond those specified in the assignment.
- Have a question? Ask the peer on your right. If not, try the peer on your left.

- \bullet Your reference guide is called **Google / man / the Internet / ...**
- Check the "C Piscine" section of the forum on the intranet or the Piscine on Slack.
- Carefully examine the examples. They may contain crucial details that are not explicitly stated in the assignment...
- By Odin, by Thor! Use your brain!!!



Norminette will be launched with the -R CheckForbiddenSourceHeader flag. Moulinette will use it too.

Chapter II

Foreword

The earliest known mention of the game Rock-Paper-Scissors (RPS) appears in the book Wuzazu, written by the Chinese Ming-dynasty author Xie Zhaozhi. He noted that the game dates back to the Chinese Han dynasty (206 BC - 220 AD). In Wuzazu, the game was called shoushiling.

Li Rihua's book *Note of Liuyanzhai* also references this game, calling it *shoushiling*, *huozhitou*, or *huoquan*.

Throughout Japanese history, there are frequent mentions of *sansukumi-ken*, meaning "ken" (fist) games with a three-way *sukumi* (deadlock). In these games, A defeats B, B defeats C, and C defeats A. The games originated in China before being introduced to Japan, where they became widely popular.

By the early 20th century, Rock-Paper-Scissors had spread beyond Asia, largely due to increased Japanese interactions with the West. Its English name is derived from a translation of the three Japanese hand gestures representing rock, paper, and scissors. Elsewhere in Asia, the open-palm gesture typically represents "cloth" rather than "paper". The depiction of scissors also follows the Japanese style.

In 1927, La Vie au Patronage, a French children's magazine, described the game in detail, referring to it as a jeu japonais ("Japanese game"). Its French name, Chi-fou-mi, is based on the Old Japanese words for "one, two, three" (hi, fu, mi).

A 1932 New York Times article on Tokyo's rush hour explained the game's rules for American readers, suggesting that it was not yet widely known in the U.S. at the time. The 1933 edition of Compton's Pictured Encyclopedia described it as a common method for children to settle disputes in Japan, calling it "John Kem Po." The article even noted, "This is such a good way of deciding an argument that American boys and girls might like to practice it too."

Chapter III

Exercise 00: ft_strcmp

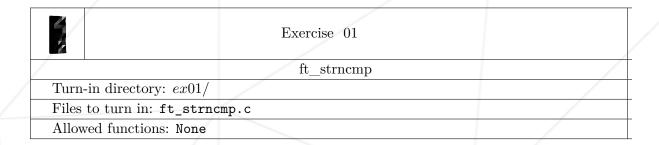
Exercis	e 00
ft_s	stremp
Turn-in directory: $ex00/$	
Files to turn in: ft_strcmp.c	
Allowed functions: None	

- Reproduce the behavior of the function strcmp (man strcmp).
- \bullet The function should be prototyped as follows:

int ft_strcmp(char *s1, char *s2);

Chapter IV

Exercise 01: ft_strncmp

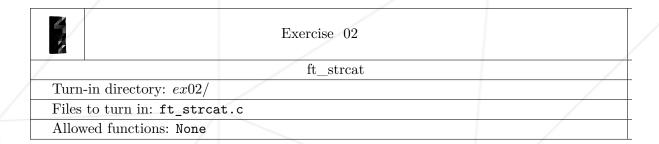


- Reproduce the behavior of the function strncmp (man strncmp).
- The function should be prototyped as follows:

int ft_strncmp(char *s1, char *s2, unsigned int n);

Chapter V

Exercise 02 : ft_strcat

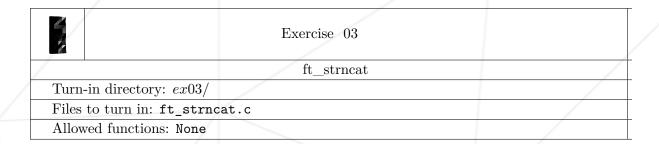


- Reproduce the behavior of the function strcat (man strcat).
- The function should be prototyped as follows:

char *ft_strcat(char *dest, char *src);

Chapter VI

Exercise 03: ft_strncat

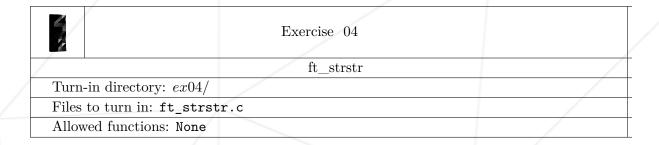


- Reproduce the behavior of the function strncat (man strncat).
- The function should be prototyped as follows:

char *ft_strncat(char *dest, char *src, unsigned int nb);

Chapter VII

Exercise 04: ft_strstr

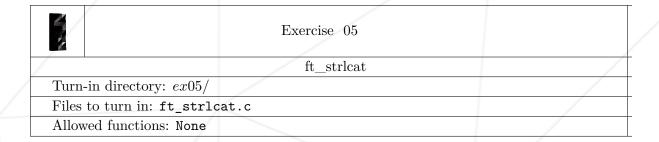


- Reproduce the behavior of the function strstr (man strstr).
- The function should be prototyped as follows:

char *ft_strstr(char *str, char *to_find);

Chapter VIII

Exercise 05: ft_strlcat



- Reproduce the behavior of the function strlcat (man strlcat).
- The function should be prototyped as follows:

unsigned int ft_strlcat(char *dest, char *src, unsigned int size);

Chapter IX

Submission and peer-evaluation

Submit your assignment in your Git repository as usual. Only the work inside your repository will be evaluated during the defense. Make sure to double-check the names of your files to ensure they are correct.



You must submit only the files explicitly requested by the project requirements.