



C Piscine

C 08

*Summary:* このドキュメントは、C Piscine @ 42の C 08モジュール用の課題である。

# Contents

<b>I</b>	<b>Instructions</b>	<b>2</b>
<b>II</b>	<b>Foreword</b>	<b>4</b>
<b>III</b>	<b>Exercise 00 : ft.h</b>	<b>5</b>
<b>IV</b>	<b>Exercise 01 : ft_boolean.h</b>	<b>6</b>
<b>V</b>	<b>Exercise 02 : ft_abs.h</b>	<b>8</b>
<b>VI</b>	<b>Exercise 03 : ft_point.h</b>	<b>9</b>
<b>VII</b>	<b>Exercise 04 : ft_strs_to_tab</b>	<b>10</b>
<b>VIII</b>	<b>Exercise 05 : ft_show_tab</b>	<b>12</b>

# Chapter I

## Instructions

- 課題に関する噂に惑わされないよう気をつけ、信用しないこと。
- この書類は、提出前に変更になる可能性があるため、気をつけること。
- ファイルとディレクトリへの権限があることを、あらかじめ確認すること。
- すべての課題は、提出手順に従い行うこと。
- 課題の確認と評価は、あなたの周りにいるPiscine受験者により行われる。
- 課題の確認と評価は、Piscine受験者に加えて、Moulinetteと呼ばれるプログラムによっても行われる。
- Moulinetteは、大変細かい評価を行う。これはすべて自動で行われるため、交渉の余地はない。
- Moulinetteは、コーディング規範（Norm）を遵守しないコードを解釈することができない。そのため、Moulinetteはnorminetteと呼ばれるプログラムを使用し、あなたのファイルがコーディング規範を遵守しているか確認を行う。せっかくの取り組みが、norminetteの確認により無駄にならないよう、気をつけること。
- 問題は、簡単なものから徐々に難しくなるように並べられている。簡単な問題が解けていない場合は、難しい問題が解けていたとしても 加点されることはない。
- 使用が禁止されている関数を使用した場合は、不正とみなされる。不正者は-42の評価をつけられ、この評価に対する交渉の余地はない。
- 課題がプログラムの提出を要求する場合のみ、main()関数を提出すること。
- Moulinetteは以下のフラグを用いて、gccでコンパイルする。 -Wall -Wextra -Werror
- プログラムがコンパイルされなかった場合、評価は0になる。
- 課題で指定されていないものは、どんなファイルもディレクトリ内に置かないこと。

- 質問がある場合は、隣の人に聞くこと。それでも分からない場合は、反対側の席の人に聞くこと。
- 助けてくれるのは、Google / 人間 / インターネット / ...と呼ばれているものたちである。
- 出力例には、問題文に明記されていない細部まで表示されている場合があるため、入念に確認すること。

# Chapter II

## Foreword

Here's what Wikipedia have to say about Platypus :

The platypus (*Ornithorhynchus anatinus*), also known as the duck-billed platypus, is a semiaquatic egg-laying mammal endemic to eastern Australia, including Tasmania. Together with the four species of echidna, it is one of the five extant species of monotremes, the only mammals that lay eggs instead of giving birth. The animal is the sole living representative of its family (*Ornithorhynchidae*) and genus (*Ornithorhynchus*), though a number of related species have been found in the fossil record.


The unusual appearance of this egg-laying, duck-billed, beaver-tailed, otter-footed mammal baffled European naturalists when they first encountered it, with some considering it an elaborate hoax. It is one of the few venomous mammals, the male platypus having a spur on the hind foot that delivers a venom capable of causing severe pain to humans. The unique features of the platypus make it an important subject in the study of evolutionary biology and a recognisable and iconic symbol of Australia; it has appeared as a mascot at national events and is featured on the reverse of its 20-cent coin. The platypus is the animal emblem of the state of New South Wales.

Until the early 20th century, it was hunted for its fur, but it is now protected throughout its range. Although captive breeding programs have had only limited success and the platypus is vulnerable to the effects of pollution, it is not under any immediate threat.

This subject is absolutly not talking about platypus.

# Chapter III

## Exercise 00 : ft.h


	Exercise 00
	ft.h
	提出するディレクトリ : <i>ex00/</i>
	提出するファイル : <i>ft.h</i>
	使用可能な関数 : None

- *ft.h* ファイルを作成せよ。
- 以下の関数のプロトタイプをファイルに入れること。

```
void    ft_putchar(char c);
void    ft_swap(int *a, int *b);
void    ft_putstr(char *str);
int     ft_strlen(char *str);
int     ft_strcmp(char *s1, char *s2);
```

# Chapter IV

## Exercise 01 : ft\_boolean.h

	Exercise 01
	ft_boolean.h
	提出するディレクトリ : ex01/
	提出するファイル : ft_boolean.h
	使用可能な関数 : None

- 以下のプログラムをコンパイルして、動作するようなft\_boolean.h を作成せよ。
- プログラムの出力例)

```
#include "ft_boolean.h"

void      ft_putstr(char *str)
{
    while (*str)
        write(1, str++, 1);
}

t_bool     ft_is_even(int nbr)
{
    return ((EVEN(nbr)) ? TRUE : FALSE);
}

int        main(int argc, char **argv)
{
    (void)argv;
    if (ft_is_even(argc - 1) == TRUE)
        ft_putstr(EVEN_MSG);
    else
        ft_putstr(ODD_MSG);
    return (SUCCESS);
}
```

- プログラムの出力例)

```
I have an even number of arguments.
```

- 上記、もしくは、以下のように標準出力に出力される。

```
I have an odd number of arguments.
```

- 必要に応じて、改行を行うこと。




`-R CheckDefine` オプションを使用して、`norminette`を実行すること。  
Moulinetteも、このオプションを使用する。



# Chapter V

## Exercise 02 : ft\_abs.h

	Exercise 02
	ft_abs.h
	提出するディレクトリ : ex02/
	提出するファイル : ft_abs.h
	使用可能な関数 : None

- 引数を絶対値に変換するマクロの ABS を作成せよ。


```
#define ABS(Value)
```



-R *CheckDefine* オプションを使用して、*norminette* を実行すること。  
Moulinette も、このオプションを使用する。

# Chapter VI

## Exercise 03 : ft\_point.h

	Exercise 03
	ft_point.h
	提出するディレクトリ : ex03/
	提出するファイル : ft_point.h
	使用可能な関数 : None

- 以下のファイルをコンパイルするために必要な、ft\_point.hを作成せよ。

```
#include "ft_point.h"


void      set_point(t_point *point)
{
    point->x = 42;
    point->y = 21;
}

int      main(void)
{
    t_point      point;

    set_point(&point);
    return (0);
}
```

# Chapter VII

## Exercise 04 : ft\_strs\_to\_tab

	Exercise 04
	ft_strs_to_tab
	提出するディレクトリ : ex04/
	提出するファイル : ft_strs_to_tab.c
	使用可能な関数 : malloc, free

- 文字列の配列と、その配列のサイズを引数として受け取る関数を作成せよ。
- プロトタイプ例)

```
struct s_stock_str *ft_strs_to_tab(int ac, char **av);
```

- avの各要素を構造体に変換すること。
- 構造体は、ft\_stock\_str.hファイルの中で定義されている。


```
typedef struct s_stock_str
{
    int size;
    char *str;
    char *copy;
} t_stock_str;
```

- sizeは、文字列の長さである。
- strは、文字列である。
- copyは、文字列のコピーである。
- avの順序を維持すること。
- 配列は、確保されたメモリに動的に配置され、最後の要素のstrは0に設定されていること。

- エラーが発生した場合は、NULLを返すこと。
- この関数は、次の問題のft\_show\_tab関数を使用してテストする。これに従って動作するように実装すること。

# Chapter VIII

## Exercise 05 : ft\_show\_tab

	Exercise 05
	ft_show_tab
	提出するディレクトリ : <i>ex05/</i>
	提出するファイル : <i>ft_show_tab.c</i>
	使用可能な関数 : <i>write</i>

- 前問の関数で作成された配列の要素を、標準出力に出力する関数を作成せよ。
- プロトタイプ例)

```
void ft_show_tab(struct s_stock_str *par);
```

- 構造体は、前問と同様に *ft\_stock\_str.h* ファイルの中で定義されている。
- 各要素の出力例)
  - 文字列と “\n”
  - 文字列のサイズと “\n”
  - 文字列のコピー（修正されている可能性があるもの）と “\n”
- この関数は、前問の *ft\_strs\_to\_tab* 関数を使用してテストする。これに従って動作するように実装すること。