



# Piscina C

## C 01

*Sumário: ESTE documento é o enunciado do módulo C 01 da Piscina C da 42.*

*Versão: 5.4*

# Conteúdo

|      |                                    |    |
|------|------------------------------------|----|
| I    | Instruções                         | 2  |
| II   | Preâmbulo                          | 4  |
| III  | Exercício 00 : ft_ft               | 6  |
| IV   | Exercício 01 : ft_ultimate_ft      | 7  |
| V    | Exercício 02 : ft_swap             | 8  |
| VI   | Exercício 03 : ft_div_mod          | 9  |
| VII  | Exercício 04 : ft_ultimate_div_mod | 10 |
| VIII | Exercício 05 : ft_putstr           | 11 |
| IX   | Exercício 06 : ft_strlen           | 12 |
| X    | Exercício 07 : ft_rev_int_tab      | 13 |
| XI   | Exercício 08 : ft_sort_int_tab     | 14 |
| XII  | Submissão e avaliação              | 15 |

# Capítulo I

## Instruções

- Somente este documento servirá de referência; não confie nos boatos.
- Leia bem o enunciado antes de entregar os seus exercícios. A qualquer momento pode haver alterações.
- Tenha atenção aos direitos dos seus ficheiros e pastas.
- Deverá seguir o procedimento de entrega para todos os exercícios.
- Os seus exercícios serão corrigidos pelos seus colegas de piscine.
- Além dos seus colegas, a Moulinette também corrigirá os seus exercícios.
- A Moulinette é extremamente rígida na sua avaliação. É completamente automatizada, e é impossível discutir a sua nota com ela. Portanto, seja rigoroso!
- A Moulinette não tem uma mente muito aberta: não tenta entender código que não respeita a Norma. A Moulinette utiliza o programa `norminette` para verificar a norma dos ficheiros. Seria uma tontice entregar código que não passa pela `norminette`...
- Os exercícios são ordenados precisamente do mais simples ao mais complexo. Em caso algum consideraremos um exercício mais complexo se outro mais simples não tiver sido perfeitamente realizado.
- A utilização de qualquer função proibida é um caso de fraude. Qualquer fraude é punida com nota de -42.
- Deve entregar uma função `main()` se for pedido um programa.
- A Moulinette compila com as flags `-Wall -Wextra -Werror`, e utiliza `cc`.
- Se o seu programa não compila, terá 0.
- Você não deve deixar em sua pasta nenhum outro arquivo além daqueles explicitamente especificados pelos enunciados dos exercícios.

- Você tem alguma dúvida? Pergunte ao seu vizinho da direita. Ou tente também perguntar ao seu vizinho da esquerda.
- Seu manual de referência se chama `Google / man / Internet / ....`
- Considere discutir no fórum Piscina do seu Intra, assim como no slack da sua Piscina!
- Leia atentamente os exemplos. Eles podem muito bem pedir coisas que não estão especificadas no tema...
- Não deve deixar no repositório de entrega nenhum outro ficheiro além daqueles explicitamente especificados pelo enunciado dos exercícios.
- Tem alguma dúvida? Pergunte ao seu vizinho da direita. Tente, também, com o seu vizinho da esquerda.



Não se esqueça de adicionar o *standard 42 header* nos seus ficheiros `.c/.h`. A norminette verifica a sua existência de qualquer forma!



A Norminette deve ser lançada com a *flag* `-R CheckForbiddenSourceHeader`. A Moulinette também a utilizará.

# Capítulo II

## Preâmbulo

Vincent: And you know what they call a... a... a Quarter Pounder with Cheese in Paris?

Jules: They don't call it a Quarter Pounder with cheese?

Vincent: No man, they got the metric system. They wouldn't know what the fuck a Quarter Pounder is.

Jules: Then what do they call it?

Vincent: They call it a Royale with cheese.

Jules: A Royale with cheese. What do they call a Big Mac?

Vincent: Well, a Big Mac's a Big Mac, but they call it le Big-Mac.

Jules: Le Big-Mac. Ha ha ha ha. What do they call a Whopper?

Vincent: I dunno, I didn't go into Burger King.

At least one of the following exercises has nothing to do you with a Royale with cheese.


## Limite Mínimo

O limiar de validação para este projeto é de 50%.

Cabe a ti determinar qual exercício te permite atingir esse limiar e se desejas completar mais exercícios.

# Capítulo III

## Exercício 00 : ft\_ft

|   |  |
|---|--|
|  | Exercício : 00                           |
|   | ft_ft                                    |
|   | Pasta de entrega : <i>ex00/</i>          |
|   | Ficheiros para entregar : <b>ft_ft.c</b> |
|   | Funções autorizadas : <b>Nenhuma</b>     |

- Escreva uma função que receba um ponteiro para inteiro como parâmetro e coloque no inteiro o valor "42".
- Deverá ser prototipada da seguinte maneira:

```
void      ft_ft(int *nbr);
```

## Exercício 01 : ft\_ultimate\_ft




- Deverá ser prototipada da seguinte maneira:

```
void ft_ultimate_ft(int *****nbr);
```



# Capítulo V

## Exercício 02 : ft\_swap


|   |                |
|---|----------------|
|  | Exercício : 02 |
| ft_swap   |                |
| Pasta de entrega : <i>ex02/</i>   |                |
| Ficheiros para entregar : <b>ft_swap.c</b>  |                |
| Funções autorizadas : Nenhuma   |                |

- Escreva uma função que troque o conteúdo de dois inteiros cujos endereços são passados como parâmetros.
- Deverá ser prototipada da seguinte maneira:

```
void    ft_swap(int *a, int *b);
```

# Capítulo VI

## Exercício 03 : ft\_div\_mod

|   |   |
|---|---|
|  | Exercício : 03                                |
|   | ft_div_mod                                    |
|   | Pasta de entrega : <i>ex03/</i>               |
|   | Ficheiros para entregar : <i>ft_div_mod.c</i> |
|   | Funções autorizadas : Nenhuma                 |


- Escreva uma função `ft_div_mod` que tenha a seguinte prototipagem:

```
void ft_div_mod(int a, int b, int *div, int *mod);
```

- A função divide os dois parâmetros `a` e `b` e armazena o resultado no inteiro apontado por `div`. Ela também armazena o resto da divisão de `a` e `b` no inteiro apontado por `mod`.

# Capítulo VII

## Exercício 04 : ft\_ultimate\_div\_mod

|   |                |
|---|----------------|
|  | Exercício : 04 |
| ft_ultimate_div_mod   |                |
| Pasta de entrega : <i>ex04/</i>   |                |
| Ficheiros para entregar : <code>ft_ultimate_div_mod.c</code>                      |                |
| Funções autorizadas : Nenhuma   |                |


- Escreva uma função `ft_ultimate_div_mod` que tenha a seguinte prototipagem:

```
void    ft_ultimate_div_mod(int *a, int *b);
```

- A função divide os inteiros apontados por **a** e **b**.  
O resultado da divisão fica armazenado no inteiro apontado por **a**.  
O resto da divisão fica armazenado no inteiro apontado por **b**.

# Capítulo VIII

## Exercício 05 : ft\_putstr


|   |                |
|---|----------------|
|  | Exercício : 05 |
| ft_putstr   |                |
| Pasta de entrega : <i>ex05/</i>   |                |
| Ficheiros para entregar : <b>ft_putstr.c</b>                                      |                |
| Funções autorizadas : <b>write</b>  |                |

- Escreva uma função que mostre um a um os caracteres de uma string na tela.
- Deverá ser prototipada da seguinte maneira:

```
void    ft_putstr(char *str);
```

# Capítulo IX

## Exercício 06 : ft\_strlen


|   |  |
|---|--|
|  | Exercício : 06                               |
|   | ft_strlen                                    |
|   | Pasta de entrega : <i>ex06/</i>              |
|   | Ficheiros para entregar : <b>ft_strlen.c</b> |
|   | Funções autorizadas : <b>Nenhuma</b>         |

- Escreva uma função que conte o número de caracteres de uma string e que retorne o valor encontrado.
- Deverá ser prototipada da seguinte maneira:

```
int ft_strlen(char *str);
```

# Capítulo X

## Exercício 07 : ft\_rev\_int\_tab


|   |   |
|---|---|
|  | Exercício : 07                                    |
|   | ft_rev_int_tab                                    |
|   | Pasta de entrega : <i>ex07/</i>                   |
|   | Ficheiros para entregar : <b>ft_rev_int_tab.c</b> |
|   | Funções autorizadas : <b>Nenhuma</b>              |

- Escreva uma função que inverta a ordem dos elementos de um array de inteiros.
- Os parâmetros são um ponteiro para inteiro e o número de inteiros no array.
- A função deverá ser prototipada da seguinte maneira:

```
void ft_rev_int_tab(int *tab, int size);
```

# Capítulo XI

## Exercício 08 : ft\_sort\_int\_tab

|   |  |
|---|--|
|  | Exercício : 08                                     |
|   | ft_sort_int_tab                                    |
|   | Pasta de entrega : <i>ex08/</i>                    |
|   | Ficheiros para entregar : <b>ft_sort_int_tab.c</b> |
|   | Funções autorizadas : <b>Nenhuma</b>               |

- Escreva uma função que ordene um array de inteiros em ordem crescente.
- Os parâmetros são um ponteiro para inteiro e o número de inteiros no array.
- A função deverá ser prototipada da seguinte maneira:

```
void ft_sort_int_tab(int *tab, int size);
```

# Capítulo XII

## Submissão e avaliação

Entrega o teu trabalho no teu repositório `Git`, como habitual. Apenas o trabalho dentro do teu repositório será avaliado durante a defesa. Não hesites em confirmar os nomes dos teus ficheiros para ter a certeza que estão corretos.



Apenas precisas de entregar os ficheiros pedidos no enunciado deste projeto.