



# Piscina C

## C 08

*Sumário: Este documento é o tema do módulo C 08 da Piscina C da 42.*

*Versão: 12.2*

# Conteúdo

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

# Capítulo I

## Instruções

- Somente esta página servirá de referência, não confie nos boatos.
- Releia bem o tema antes de entregar seus exercícios. A qualquer momento o tema pode mudar.
- Atenção aos direitos de seus arquivos e suas pastas.
- Você deve seguir o procedimento de entrega para todos os seus exercícios.
- Os seus exercícios serão corrigidos por seus colegas de piscina.
- Além dos seus colegas, haverá a correção por um programa chamado Moulinette.
- A Moulinette é muito rigorosa na sua avaliação. Ela é completamente automatizada. É impossível discutir sua nota com ela. Tenha um rigor exemplar para evitar surpresas.
- A Moulinette não tem a mente muito aberta. Ela não tenta entender o código que não respeita a Norma. A Moulinette utiliza o programa **norminette** para verificar a norma dos seus arquivos. Então é uma tolice entregar um código que não passa pela **norminette**.
- Os exercícios estão rigorosamente ordenados do mais simples ao mais complexo. Em nenhum caso daremos atenção, nem levaremos em conta um exercício complexo se outro mais simples não tiver sido perfeitamente realizado.
- A utilização de uma função proibida é um caso de fraude. Qualquer fraude é punida com nota de -42.
- Você não deve entregar uma função `main()` se nós pedirmos um programa.
- A Moulinette compila com as sinalizações -Wall -Wextra -Werror, e utiliza `cc`.
- Se o seu programa não compila, você 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...
- Reflita. Por favor, por Odin! Por tudo que é mais sagrado.



Hoje, a Norminette deve ser lançada com a sinalização `-R CheckForbiddenSourceHeader`. A Moulinette também a utilizará.

# Capítulo II

## Preâmbulo

É isto que a enciclopédia colaborativa *Wikipédia* tem a dizer sobre o ornitorrinco:

O ornitorrinco (*Ornithorhynchus anatinus*) é uma espécie de pequeno mamífero semiaquático endêmico do leste da Austrália, incluindo a Tasmânia. É uma das cinco espécies da ordem dos monotremados, única ordem de mamíferos que põem ovos em vez de dar à luz filhotes completamente formados (as quatro outras espécies são equidnas). É a única espécie sobrevivente da família Ornithorhynchidae e do gênero *Ornithorhynchus*, embora um grande número de fragmentos de espécies fósseis dessa família e desse gênero tenha sido descoberto.


A aparência estranha desse mamífero, que põe ovos, tem esporões venenosos, uma mandíbula achatada como um bico de pato, uma cauda que lembra a de um castor, que serve ao mesmo tempo como leme na água e de reserva de gordura, e patas de lontra, realmente surpreendeu os primeiros exploradores que o encontraram. Muitos naturalistas europeus acreditaram que se tratava de uma brincadeira. É um dos raros mamíferos venenosos: o macho tem nas patas traseiras um esporão que pode liberar um veneno capaz de provocar fortes dores a um ser humano. As características incomuns do ornitorrinco o tornaram um objeto importante de estudo para compreender melhor a evolução das espécies animais e o tornaram também um dos símbolos da Austrália: ele foi utilizado como mascote de vários eventos nacionais e está no verso da moeda de 20 centavos australianos.

Até o início do século XX, ele foi caçado por sua pele, mas atualmente é uma espécie protegida. Apesar de os programas de reprodução em cativeiro terem um sucesso muito limitado e o animal ser sensível aos efeitos da poluição, não se considera que a espécie esteja em perigo.

Esse tema não tem a ver com os ornitorrincos.

# Capítulo III

## Exercício 00 : ft.h


	Exercício : 00
ft.h	
Pasta de entrega : <i>ex00/</i>	
Arquivos para entregar : <i>ft.h</i>	
Funções autorizadas : Nenhuma	

- Escreva seu arquivo *ft.h*
- Ele contém todos os protótipos das funções:

```
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);
```

# Capítulo IV

## Exercício 01 : ft\_boolean.h

	Exercício : 01
	ft_boolean.h
	Pasta de entrega : ex01/
	Arquivos para entregar : ft_boolean.h
	Funções autorizadas : Nenhuma

- Escreva um arquivo `ft_boolean.h` que compile e faça funcionar corretamente o seguinte main:

```
#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);
}
```

- Esse programa deverá mostrar

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

- ou


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

- seguido de uma quebra de linha, no caso adequado.



# Capítulo V

## Exercício 02 : ft\_abs.h


	Exercício : 02
	ft_abs.h
	Pasta de entrega : <i>ex02/</i>
	Arquivos para entregar : <b>ft_abs.h</b>
	Funções autorizadas : <b>Nenhuma</b>

- Escreva uma macro ABS que substitua seu parâmetro por seu valor absoluto:

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

# Capítulo VI

## Exercício 03 : ft\_point.h

	Exercício : 03
ft_point.h	
Pasta de entrega : <i>ex03/</i>	
Arquivos para entregar : <b>ft_point.h</b>	
Funções autorizadas : Nenhuma	

- Escreva um arquivo **ft\_point.h** que compile o seguinte main:

```
#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);
}
```

# Capítulo VII

## Exercício 04 : ft\_strs\_to\_tab

	Exercício : 04
	ft_strs_to_tab
	Pasta de entrega : <i>ex04/</i>
	Arquivos para entregar : <code>ft_strs_to_tab.c</code>
	Funções autorizadas : <code>malloc</code> , <code>free</code>

- Escreva uma função que tenha como parâmetro uma matriz de cadeia de caracteres, assim como o tamanho dessa matriz e retorne uma matriz de estrutura.
- Ela deverá ser prototipada da seguinte maneira:

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

- Ela deve transformar cada elemento da matriz de cadeia de caracteres em estrutura.
- A estrutura será definida no arquivo `ft_stock_str.h` da seguinte forma:


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

- `size` sendo o tamanho da cadeia de caracteres;
  - `str` sendo a cadeia de caracteres;
  - `copy` sendo uma cópia da cadeia de caracteres;
- Ela deve manter a ordem dos elementos de `av`.

- A matriz de estruturas deverá ser designada, e o último elemento terá 0 como valor de `str` a fim de indicar o fim da matriz.
- Se houver um erro de alocação, ela deverá retornar um ponteiro NULL.
- Vamos testar sua função com nosso `ft_show_tab` (próximo exercício). Tome as medidas necessárias para que funcione!

# Capítulo VIII

## Exercício 05 : ft\_show\_tab

	Exercício : 05
	ft_show_tab
	Pasta de entrega : <i>ex05/</i>
	Arquivos para entregar : <b>ft_show_tab.c</b>
	Funções autorizadas : <b>write</b>

- Escreva uma função que mostre o conteúdo de uma matriz criada pela função anterior.
- Ela deverá ser prototipada da seguinte maneira:

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

- A estrutura é a mesma do exercício anterior e estará no arquivo **ft\_stock\_str.h** que vamos fornecer:
- Para cada elemento da matriz:
  - a cadeia de caracteres deve ser seguida de uma quebra de linha
  - o tamanho seguido de uma quebra de linha
  - a cópia da cadeia de caracteres (que poderá ter sido modificada) seguida de uma quebra de linha
- Vamos testar sua função com nosso **ft\_strs\_to\_tab** (exercício anterior). Tome as medidas necessárias para que funcione!