



# 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 este documento servirá de referência; não confie nos boatos.
- Releia 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.
- 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.
- A bibliografia para consulta chama-se `Google / man / Internet / ....`
- Considere discutir os exercícios no Slack da sua piscine!
- Leia atentamente os exemplos: podem demonstrar coisas que não estão especificadas no enunciado...
- Reflita. Por favor, por Odin! Por tudo que é mais sagrado.

# 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>	
Ficheiros para entregar : <b>ft.h</b>	
Funções autorizadas : Nenhuma	

- Crie um ficheiro `ft.h`
- Ele deve conter todos os protótipos das funções seguintes:

```
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/
	Ficheiros para entregar : ft_boolean.h
	Funções autorizadas : Nenhuma

- Escreva um ficheiro 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 imprimir

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

- ou

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

- seguido de uma quebra de linha, quando necessário.




A Norminette deve ser lançada com a sinalização `-R CheckDefine`. A Moulinette também a utilizará.



# Capítulo V

## Exercício 02 : ft\_abs.h

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

- Crie uma macro ABS que substitua o parâmetro pelo seu valor absoluto:


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



A Norminette deve ser lançada com a sinalização `-R CheckDefine`. A Moulinette também a utilizará.

# Capítulo VI

## Exercício 03 : ft\_point.h

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

- Crie um ficheiro **ft\_point.h** que compile a 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 : ex04/
	Ficheiros para entregar : ft_strs_to_tab.c
	Funções autorizadas : malloc, free

- Escreva uma função que tenha como parâmetros, uma matriz de *string* e o tamanho dessa matriz.
- 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 *string* numa estrutura.
- A estrutura será definida no ficheiro `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 *string*;
  - `str` sendo a *string*;
  - `copy` sendo uma cópia da *string*;
- Ela deve manter a ordem dos elementos de `av`.

- A matriz de estruturas deverá ser alocada, 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 *pointer* NULL.
- Vamos testar sua função com o 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 : ex05/
	Ficheiros para entregar : ft_show_tab.c
	Funções autorizadas : write

- 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 ficheiro `ft_stock_str.h` que vamos fornecer:
- Para cada elemento da matriz:
  - a *string* deve ser seguida de uma quebra de linha
  - o tamanho seguido de uma quebra de linha
  - a cópia da *string* (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!