



Piscina C

C 01

Sumário: Este documento é o tema do módulo C 01 da Piscina C da 42.

Versão: 5.2

Conteúdo

I	Instruções	2
II	Preâmbulo	4
III	Exercício 00 : ft_ft	5
IV	Exercício 01 : ft_ultimate_ft	6
V	Exercício 02 : ft_swap	7
VI	Exercício 03 : ft_div_mod	8
VII	Exercício 04 : ft_ultimate_div_mod	9
VIII	Exercício 05 : ft_putstr	10
IX	Exercício 06 : ft_strlen	11
X	Exercício 07 : ft_rev_int_tab	12
XI	Exercício 08 : ft_sort_int_tab	13

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 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 tólice 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 não 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

O jogo do Xarope segundo Perceval, extraído da série *Kaamelott* :

"Bom, eu vou ensinar as regras simplificadas, porque as verdadeiras regras são complexas. A única coisa é que normalmente se joga a três.
Mas não tem problema, a gente vai dar um jeito.

O princípio é transformar em valores. Então, vamos dizer que estamos em três e há três valores para distribuir. Vamos dizer, xarope de oito, xarope de catorze e xarope de vinte e um. Não se preocupem com os xaropes agora. Primeiro, é preciso entender os valores. Se você jogar um valor no início da rodada, vamos dizer um xarope de oito, para começar devagar, os outros têm a opção de deixar passar ou lançar um xarope de catorze. O sentido vai de acordo com os valores. É por isso que é preciso entender bem o sistema de valores. Depois disso, tudo fica mais claro.

Bom, vamos dizer que eu começo com um xarope de oito.
Se foi você que bebeu na rodada anterior, é a sua vez. Então ou você deixa passar e diz "passo-xarope", ou você joga e anuncia um xarope de catorze. Como começamos as jogadas, o segundo jogador não tem o direito de passar. Você pode ou jogar um xarope de vinte e um ou abandonar a rodada e dizer "dorme-xarope" ou "xarope Jeannot", isso depende da região. E depois ou a gente joga a partida ou eu faço um "contra-xarope"! E a partir de então, xarope de maçã sobre vinte e um, então jogamos em quatro rodadas até que alguém beba.

Para ganhar, só tem três possibilidades: ou você bebe seu xarope de oito e diz "bom xarope", e nós recontamos, ou você bebe seu xarope de catorze e diz "bom xarope, xarope vencedor", e nós te acrescentamos a metade, ou você bebe seu xarope de vinte e um e diz "bom xarope, meio xarope, bebi, xarope vencedor, xarope-sino, balaclava, xarope gostoso".

Normalmente, joga-se com cartas, mas se você só tiver dados, também pode jogar com eles porque o que importa são os valores."

Pelo menos um dos exercícios seguintes não tem nada a ver com o jogo do Xarope.

Capítulo III

Exercício 00 : ft_ft

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

- Escreva uma função que tenha um ponteiro para int em parâmetro e dê ao int o valor de 42.
- Ela deverá ser prototipada da seguinte maneira:

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

Capítulo IV

Exercício 01 : ft_ultimate_ft

	Exercício : 01
	ft_ultimate_ft
	Pasta de entrega : ex01/
	Arquivos para entregar : <u>ft_ultimate_ft.c</u>
	Funções autorizadas : Nenhuma

- Escreva uma função que tenha um ponteiro para int em parâmetro e dê ao int o valor de 42.
- Ela 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>
	Arquivos para entregar : <i>ft_swap.c</i>
	Funções autorizadas : Nenhuma

- Escreva uma função que troque o conteúdo de dois inteiros cujos endereços são dados em parâmetros.
- Ela 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>
	Arquivos para entregar : ft_div_mod.c
	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);
```

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

Capítulo VII

Exercício 04 : ft_ultimate_div_mod

	Exercício : 04
	ft_ultimate_div_mod
	Pasta de entrega : <i>ex04/</i>
	Arquivos para entregar : <i>ft_ultimate_div_mod.c</i>
	Funções autorizadas : Nenhuma

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

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

- Essa função divide os int apontados por *a* e *b*.
O resultado da divisão está armazenado no int apontado por *a*.
O resultado do resto da divisão está armazenado no int apontado por *b*.

Capítulo VIII

Exercício 05 : ft_putstr

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

- Escreva uma função que mostre um a um os caracteres de uma string na tela.
- O endereço do primeiro caractere da string está contido no ponteiro passado como parâmetro para a função.
- Ela 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>
	Arquivos para entregar : ft_strlen.c
	Funções autorizadas : Nenhuma

- Escreva uma função que contenha o número de caracteres em uma string de caracteres que retorne ao número encontrado.
- Ela 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>
	Arquivos para entregar : <u>ft_rev_int_tab.c</u>
	Funções autorizadas : Nenhuma

- Escreva uma função que inverta a ordem dos elementos de uma matriz de inteiros.
- Os parâmetros são um ponteiro para inteiro e o número de inteiros na matriz.
- 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>
	Arquivos para entregar : <u>ft_sort_int_tab.c</u>
	Funções autorizadas : Nenhuma

- Escreva uma função que ordene uma matriz de inteiros em ordem crescente.
- Os parâmetros são um ponteiro para inteiro e o número de inteiros na matriz.
- A função deverá ser prototipada da seguinte maneira:

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