



Piscina C

C 06

Sumário: Este documento é o tema do módulo C 06 da Piscina C da 42.

Conteúdo

I	Instruções	2
II	Preâmbulo	4
III	Exercício 00 : ft_print_program_name	5
IV	Exercício 01 : ft_print_params	6
V	Exercício 02 : ft_rev_params	7
VI	Exercício 03 : ft_sort_params	8

Capítulo I

Instruções

- Somente esta página servirá de referência, não confie nos boatos.
- Leia 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 `gcc`.
- 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

Diálogo do filme O Grande Lebowski:

Walter: Desculpa, Smokey. Você pisou na linha: falta.

Smokey: Mentira! Marca oito, Cara...

Walter: Me desculpa. Marca zero. Próxima rodada.

Smokey: Deixa de ser chato, Walter. Marca oito, Cara...

Walter: Smokey, não estamos no Vietnã. Estamos no boliche. A gente joga com regras...

O Cara: Vai, Walter, não enche. O pé dele escorregou um pouquinho na linha... É só um jogo!

Walter: Sim. E é um jogo de campeonato, decide quem vai pra próxima fase, não é?

Smokey: É, mas eu não...

Walter: Não é verdade?

Smokey: Sim, mas eu não pisei... Vai! Cara, marca oito.

Walter: [saca sua arma] Smokey, meu amigo, se você nunca sentiu dor, você vai sentir...

Cara: Walter, não seja idiota!

Walter [ameaçador] Vai, marca um oito que você vai ver!

Smokey: Eu não...

Walter: Você vai sentir dor, Smokey!

Smokey: Cara? É o seu parceiro...

Walter: O mundo todo tá FICANDO LOUCO? [Ele se levanta] EU SOU O ÚNICO QUE LIGA PRAS REGRAS? MARCA ZERO!

Cara: Eles vão chamar a polícia, Walter, guarda essa...

Walter: [aponta a arma para Smokey] MARCA ZERO!

Cara: Guarda a arma, Walter...

Smokey: ...

Cara: Walter...


Walter: [engatilha a arma] VOCÊ ACHA QUE EU TÔ BRINCANDO? MARCA ZERO!

Smokey: Tá bom, marquei zero... Tá feliz? Seu louco!

Walter: [se acalma] ... É um jogo de campeonato.

Capítulo III

Exercício 00 : ft_print_program_name


	Exercício : 00
ft_print_program_name	
Pasta de entrega : <i>ex00/</i>	
Arquivos para entregar : <code>ft_print_program_name.c</code>	
Funções autorizadas : <code>write</code>	

- A questão aqui é sobre um programa, você deve, portanto, ter uma função `main` no seu arquivo `.c`.
- Escreva um programa que mostre o nome do programa.
- Exemplo :

```
$> ./a.out  
./a.out  
$>
```

Capítulo IV

Exercício 01 : ft_print_params


	Exercício : 01
	ft_print_params
	Pasta de entrega : <i>ex01/</i>
	Arquivos para entregar : <code>ft_print_params.c</code>
	Funções autorizadas : <code>write</code>

- A questão aqui é sobre um programa, você deve, portanto, ter uma função `main` no seu arquivo arquivo `.c`.
- Escreva um programa que mostre os argumentos recebidos na linha de comando.
- Um por linha na mesma ordem da linha de comando
- Você deve mostrar todos os argumentos, menos `argv[0]`.
- Exemplo :

```
$>./a.out test1 test2 test3
test1
test2
test3
$>
```

Capítulo V


Exercício 02 : ft_rev_params

	Exercício : 02
	ft_rev_params
Pasta de entrega : <i>ex02/</i>	
Arquivos para entregar : ft_rev_params.c	
Funções autorizadas : write	

- A questão aqui é sobre um programa, você deve, portanto, ter uma função **main** no seu arquivo arquivo **.c**.
- Escreva um programa que mostre os argumentos recebidos na linha de comando.
- Um por linha na ordem inversa da linha de comando.
- Você deve mostrar todos os argumentos, menos **argv[0]**.

Capítulo VI

Exercício 03 : ft_sort_params

	Exercício : 03
	ft_sort_params
Pasta de entrega : <i>ex03/</i>	
Arquivos para entregar : <code>ft_sort_params.c</code>	
Funções autorizadas : <code>write</code>	

- A questão aqui é sobre um programa, você deve, portanto, ter uma função `main` no seu arquivo `.c`.
- Escreva um programa que mostre os argumentos recebidos na linha de comando classificados por ordem `ascii`.
- Você deve mostrar todos os argumentos, menos `argv[0]`.
- Um argumento por linha.