



Piscina C

C 06

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

Versão: 7.3

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
VII	Entrega e avaliação entre pares	9

Capítulo I

Instruções

- Somente esta página servirá de referência, não confie nos boatos.
- Cuidado! Este documento pode mudar até uma hora antes da entrega.
- Certifique-se que você tenha as permissões apropriadas nos seus arquivos e pastas.
- Você deverá seguir o procedimento de entrega para todos os exercícios.
- Os seus exercícios serão corrigidos por seus colegas de Piscina.
- Além disso, seus exercícios serão verificados e corrigidos por um programa chamado Moulinette
- A Moulinette é extremamente rígida em sua avaliação. Ela é completamente automatizada e é impossível discutir sua nota com ela. Se você quiser evitar surpresas, tenha muita atenção.
- A Moulinette não tem a mente muito aberta. Ela não tentará entender seu código se ele não respeitar a Norma. A Moulinette conta com um programa chamado **norminette** para verificar se seus arquivos respeitam a Norma. TL;DR: seria idiota enviar um trabalho que não passa na verificação da **norminette**.
- Os exercícios estão arranjados cuidadosamente em ordem de dificuldade - do mais fácil para o mais difícil. Nós **não** levaremos em conta um exercício mais difícil completo se um mais fácil não estiver perfeitamente funcional.
- Usar uma função proibida é considerado trapaça. Trapaceiros levam -42, e esta nota não é negociável.
- Você somente deverá enviar uma função `main()` se nós pedirmos um **programa**.
- A Moulinette compila com as flags: `-Wall -Wextra -Werror`, e usa `cc`.
- Se o seu programa não compila, você receberá um 0.

- Você não pode deixar **nenhum** arquivo adicional na sua pasta além dos especificados no subject.
- Tem uma dúvida? Pergunte para o seu parceiro da direita. Ou então tente o seu parceiro da esquerda.
- Seu guia de referência chama-se `Google / man / a Internet /`
- Verifique o fórum "C Piscine" na intranet, ou o servidor do Discord da Piscina.
- Examine os exemplos atentamente. Pode haver detalhes neles que não estão explícitos no subject...
- Por Odin, por Thor! Use seu cérebro!!!



Não se esqueça de adicionar o cabeçalho padrão da 42 em cada um dos seus arquivos `.c/.h`. A norminette verifica se ele está lá!



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

Capítulo II

Preâmbulo

Diálogo do filme O Grande Lebowski:

O Cara: Walter, você sabe, é o Smokey, o dedão dele escapou um pouquinho, grande coisa. É só um jogo, cara.

Walter Sobchak: Cara, é um jogo da liga, isso determina quem entra na próxima rodada.

Eu tô errado? Eu tô errado?

Smokey: É, mas não tinha terminado. Me dá o marcador, Cara, eu tô marcando um 8.

Walter Sobchak: [puxa uma arma] Smokey, meu amigo, você tá entrando num mundo de dor.

O Cara: Walter...

Walter Sobchak: Se você marcar um 8 nesse quadro, você tá entrando num mundo de dor.

Smokey: Não tô...

Walter Sobchak: Um mundo de dor.

Smokey: Cara, ele é seu parceiro...

Walter Sobchak: [gritando] O mundo inteiro ficou louco? Eu sou o único aqui que se importa com as regras? Marca zero!

O Cara: Eles tão chamando a polícia, tira a peça daí.

Walter Sobchak: Marca zero!

[aponta a arma na cara do Smokey]

O Cara: Walter...


Walter Sobchak: [gritando] Você acha que eu tô de brincadeira? Marca zero!

Smokey: Ok, é a merda de um zero. Tá feliz, seu louco de merda?

Walter Sobchak: ...É um jogo da liga, Smokey.

Capítulo III

Exercício 00 : ft_print_program_name


	Exercício : 00
ft_print_program_name	
Pasta de entrega : <i>ex00/</i>	
Ficheiros 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 exiba 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>
	Ficheiros 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 exiba os argumentos recebidos na linha de comando.
- Um por linha na mesma ordem da linha de comando
- Você deve exibir 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>
	Ficheiros 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 exiba os argumentos recebidos na linha de comando.
- Um por linha na ordem inversa da linha de comando.
- Você deve exibir todos os argumentos, menos **argv[0]**.

Capítulo VI

Exercício 03 : ft_sort_params

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

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

Capítulo VII

Entrega e avaliação entre pares

Entregue seu projeto em seu repositório `Git` como de costume. Somente o trabalho contido em seu repositório será avaliado durante a defesa. Não hesite em verificar mais de uma vez os nomes dos seus arquivos para ter certeza de que eles estão corretos.



Você deve submeter somente os arquivos solicitados pelo subject deste projeto.