



Piscina C

C 05

Sumário: Este documento é o tema do módulo C 05 da Piscina C da 42.

Versão: 6.3

Conteúdo

I	Instruções	2
II	Preâmbulo	4
III	Exercício 00 : ft_iterative_factorial	6
IV	Exercício 01 : ft_recursive_factorial	7
V	Exercício 02 : ft_iterative_power	8
VI	Exercício 03 : ft_recursive_power	9
VII	Exercício 04 : ft_fibonacci	10
VIII	Exercício 05 : ft_sqrt	11
IX	Exercício 06 : ft_is_prime	12
X	Exercício 07 : ft_find_next_prime	13
XI	Exercício 08 : As dez damas	14
XII	Entrega e avaliação entre pares	15

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

A seguir, um trecho do primeiro livro da saga Harry Potter:

Ah, vocês podem me achar pouco atraente,
Mas não me julguem só pela aparência
Engulo a mim mesmo se puderem encontrar
Um chapéu mais inteligente do que o papai aqui.

Podem guardar seus chapéus-coco bem pretos,
Suas cartolas altas de cetim brilhoso
Porque eu sou o Chapéu Seletor de Hogwarts
E dou de dez a zero em qualquer outro chapéu.

Não há nada escondido em sua cabeça
Que o Chapéu Seletor não consiga ver,
Por isso é só me porem na cabeça que vou dizer
Em que casa de Hogwarts deverão ficar.

Quem sabe sua morada é a Grifinória,
Casa onde habitam os corações indomáveis,
Ousadia e sangue-frio e nobreza
Destacam os alunos da Grifinória dos demais.

Quem sabe é na Lufa-Lufa que você vai morar,
Onde seus moradores são justos e leais
Pacientes, sinceros, sem medo da dor;
Ou será a velha e sábia Corvinal,

A casa dos que têm a mente sempre alerta,
Onde os homens de grande espírito e saber
Sempre encontrarão companheiros seus iguais;
Ou quem sabe a Sonserina será a sua casa


E ali fará seus verdadeiros amigos,
Homens de astúcia que usam quaisquer meios
Para atingir os fins que antes colimaram.
Vamos, me experimentem!

Não devem temer! Nem se atrapalhar!
Estarão em boas mãos!
(Mesmo que os chapéus não tenham pés nem mãos)
Porque eu sou único, sou um Chapéu Pensador!

Infelizmente, esse tema não tem nada a ver com a série Harry Potter e é uma pena, porque sua entrega não será feita por magia.

Capítulo III

Exercício 00 : ft_iterative_factorial


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

- Escreva uma função iterativa que retorne um número. Esse número deve ser o resultado da operação fatorial a partir do número passado como parâmetro.
- Se o argumento não for válido, a função deve retornar 0.
- Não se deve lidar com overflows, o retorno da função será indefinido.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_iterative_factorial(int nb);
```

Capítulo IV

Exercício 01 : ft_recursive_factorial


	Exercício : 01
	ft_recursive_factorial
	Pasta de entrega : <i>ex01/</i>
	Ficheiros para entregar : ft_recursive_factorial.c
	Funções autorizadas : Nenhuma

- Escreva uma função recursiva que retorne o fatorial do número passado como parâmetro.
- Se o argumento não for válido, a função deve retornar 0.
- Não se deve lidar com overflows, o retorno da função será indefinido.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_recursive_factorial(int nb);
```


Capítulo V

Exercício 02 : ft_iterative_power


	Exercício : 02
	ft_iterative_power
	Pasta de entrega : <i>ex02/</i>
	Ficheiros para entregar : <code>ft_iterative_power.c</code>
	Funções autorizadas : Nenhuma

- Escreva uma função iterativa que retorne uma potência de um número.
- Uma potência inferior a 0 retornará 0.
- Não se deve lidar com overflows.
- Nós decidimos que 0 elevado a 0 retornará 1.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_iterative_power(int nb, int power);
```

Capítulo VI

Exercício 03 : ft_recursive_power


	Exercício : 03
	ft_recursive_power
	Pasta de entrega : <i>ex03/</i>
	Ficheiros para entregar : <code>ft_recursive_power.c</code>
	Funções autorizadas : Nenhuma

- Escreva uma função recursiva que retorne uma potência de um número.
- Uma potência menor que 0 retorna 0.
- Não se deve lidar com overflows, o retorno da função será indefinido.
- Nós decidimos que 0 elevado a 0 retornará 1.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_recursive_power(int nb, int power);
```

Capítulo VII

Exercício 04 : ft_fibonacci

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


- Escreva uma função **ft_fibonacci** que retorne o **n**-ésimo elemento da sequência de Fibonacci, sendo o primeiro elemento com índice 0. Vamos considerar que a sequência de Fibonacci começa por 0, 1, 1, 2.
- Não se deve lidar com overflows, o retorno da função será indefinido.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_fibonacci(int index);
```

- Evidentemente, **ft_fibonacci** deverá ser recursiva.
- Se **index** for inferior a 0, a função retornará -1.

Capítulo VIII

Exercício 05 : ft_sqrt


	Exercício : 05
ft_sqrt	
Pasta de entrega : <i>ex05/</i>	
Ficheiros para entregar : ft_sqrt.c	
Funções autorizadas : Nenhuma	

- Escreva uma função que retorne a raiz quadrada de um número (se ela existir), ou 0 se a raiz quadrada for um número irracional.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_sqrt(int nb);
```

Capítulo IX

Exercício 06 : ft_is_prime

	Exercício : 06
	ft_is_prime
	Pasta de entrega : <i>ex06/</i>
	Ficheiros para entregar : <code>ft_is_prime.c</code>
	Funções autorizadas : Nenhuma

- Escreva uma função que retorne 1 se o número for primo e 0 se não for.
- Ela deverá ser prototipada da seguinte maneira:


```
int ft_is_prime(int nb);
```



0 e 1 não são números primos.

Capítulo X

Exercício 07 : ft_find_next_prime


	Exercício : 07
ft_find_next_prime	
Pasta de entrega : <i>ex07/</i>	
Ficheiros para entregar : <code>ft_find_next_prime.c</code>	
Funções autorizadas : Nenhuma	

- Escreva uma função que retorne o número primo imediatamente superior ou igual ao número passado como parâmetro.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_find_next_prime(int nb);
```

Capítulo XI

Exercício 08 : As dez damas

	Exercício : 08
As dez damas	
Pasta de entrega : <i>ex08/</i>	
Ficheiros para entregar : <i>ft_ten_queens_puzzle.c</i>	
Funções autorizadas : <i>write</i>	

- Escreva uma função que mostre todas as possibilidades de posicionar dez damas em um tabuleiro de 10x10 sem que elas possam ser atingidas com um único movimento, e retorne o número de possibilidades.
- A recursividade deverá ser utilizada para resolver este problema.
- Ela deverá ser prototipada da seguinte maneira:

```
int ft_ten_queens_puzzle(void);
```

- A exibição deverá ser feita da seguinte maneira:

```
$>./a.out | cat -e
0257948136$
0258693147$
...
4605713829$
4609582731$
...
9742051863$
$>
```

- A sequência é lida da esquerda para a direita. O primeiro dígito corresponde à posição da primeira dama na primeira coluna (o índice começando com 0). O enésimo número corresponde à posição da enésima dama na enésima coluna.

Capítulo XII

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.