



Piscina C

C 00

Sumário: Este documento é o tema do módulo C 00 da Piscina C da 42.

Versão: 7.2

Conteúdo

I	Instruções	2
II	Preâmbulo	4
III	Exercício 00 : ft_putchar	5
IV	Exercício 01 : ft_print_alphabet	6
V	Exercício 02 : ft_print_reverse_alphabet	7
VI	Exercício 03 : ft_print_numbers	8
VII	Exercício 04 : ft_is_negative	9
VIII	Exercício 05 : ft_print_comb	10
IX	Exercício 06 : ft_print_comb2	11
X	Exercício 07 : ft_putnbr	12
XI	Exercício 08 : ft_print_combn	13
XII	Entrega e avaliação entre pares	14

Capítulo I

Instruções

- Somente esta página servirá de referência, não confie nos boatos.
- Fique atento! Este documento pode mudar antes da sua entrega. Releia antes de entregar.
- 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 de 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 não pedirmos um programa.
- A Moulinette compila com as marcações -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 "C Piscine" na intranet, assim como no Discord 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.



Para esse módulo, a Norminette deve ser lançada com a sinalização `-R CheckForbiddenSourceHeader`. A Moulinette também a utilizará.

Capítulo II

Preâmbulo

O óleo de fígado de bacalhau é um suplemento nutricional derivado do fígado do bacalhau.

Assim como a maioria dos óleos de peixe, ele tem níveis altos de ácidos graxos ômega-3, incluindo ácido eicosapentaenóico (EPA) e ácido docosahexaenóico (DHA).

Óleo de fígado de bacalhau também contém vitaminas A e D.


Ele tem sido historicamente tomado por causa de seu conteúdo de vitamina A e vitamina D.

Já foi comumente administrado a crianças, porque a vitamina D demonstrou prevenir o raquitismo.

Ao contrário do óleo de fígado de bacalhau, o C é bom. Devore!

Capítulo III

Exercício 00 : ft_putchar

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

- Escreva uma função que mostre o caractere passado como parâmetro.
- Ela deve ser prototipada da seguinte maneira:


```
void ft_putchar(char c);
```

Para indicar o caractere, você deve usar a função **write** da seguinte maneira.

```
write(1, &c, 1);
```

Capítulo IV

Exercício 01 : ft_print_alphabet


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

- Escreva uma função que mostre o alfabeto em minúsculo em apenas uma linha, em ordem crescente, começando pela letra 'a'.
- Ela deve ser prototipada da seguinte maneira:

```
void ft_print_alphabet(void);
```

Capítulo V

Exercício 02 : ft_print_reverse_alphabet


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

- Escreva uma função que mostre o alfabeto em minúsculo em apenas uma linha, em ordem decrescente, começando pela letra 'z'.
- Ela deve ser prototipada da seguinte maneira:

```
void ft_print_reverse_alphabet(void);
```


Capítulo VI

Exercício 03 : ft_print_numbers


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

- Escreva uma função que mostre todos os números em uma única linha em ordem crescente.
- Ela deve ser prototipada da seguinte maneira:

```
void ft_print_numbers(void);
```

Capítulo VII

Exercício 04 : ft_is_negative


	Exercício : 04
ft_is_negative	
Pasta de entrega : <i>ex04/</i>	
Ficheiros para entregar : <code>ft_is_negative.c</code>	
Funções autorizadas : <code>write</code>	

- Escreva uma função que mostre 'N' ou 'P' segundo o sinal do inteiro passado como parâmetro. Se `n` for negativo, indique 'N'. Se `n` for positivo ou nulo, indique 'P'.
- Ela deve ser prototipada da seguinte maneira:

```
void ft_is_negative(int n);
```

Capítulo VIII

Exercício 05 : ft_print_comb

	Exercício : 05
	ft_print_comb
	Pasta de entrega : <i>ex05/</i>
	Ficheiros para entregar : <code>ft_print_comb.c</code>
	Funções autorizadas : <code>write</code>

- Escreva uma função que mostre, em ordem crescente, todas as diferentes combinações de três dígitos diferentes em ordem crescente - sim, a repetição é proposital.
- Isso resulta em algo do tipo:


```
$>./a.out | cat -e
012, 013, 014, 015, 016, 017, 018, 019, 023, ..., 789$>
```

- 987 fica de fora porque o 789 já está presente.
- 999 fica de fora porque o dígito 9 está presente mais de uma vez.
- Ela deve ser prototipada da seguinte maneira:

```
void ft_print_comb(void);
```

Capítulo IX

Exercício 06 : ft_print_comb2

	Exercício : 06
ft_print_comb2	
Pasta de entrega : <i>ex06/</i>	
Ficheiros para entregar : <code>ft_print_comb2.c</code>	
Funções autorizadas : <code>write</code>	

- Escreva uma função que mostre todas as diferentes combinações de dois dígitos entre 0 e 99, em ordem crescente.

- Isso resulta em algo do tipo:


```
$>./a.out | cat -e
00 01, 00 02, 00 03, 00 04, 00 05, ..., 00 99, 01 02, ..., 97 99, 98 99$>
```

- Ela deve ser prototipada da seguinte maneira:

```
void ft_print_comb2(void);
```

Capítulo X

Exercício 07 : ft_putnbr

	Exercício : 07
ft_putnbr	
Pasta de entrega : <i>ex07/</i>	
Ficheiros para entregar : ft_putnbr.c	
Funções autorizadas : write	


- Escreva uma função que mostre um número passado como parâmetro. A função deverá ser capaz de representar a totalidade dos valores possíveis em uma variável do tipo `int`.
- Ela deve ser prototipada da seguinte maneira:

```
void ft_putnbr(int nb);
```

- Por exemplo:
 - `ft_putnbr(42)` mostra "42".

Capítulo XI

Exercício 08 : ft_print_combn

	Exercício : 08
ft_print_combn	
Pasta de entrega : <i>ex08/</i>	
Ficheiros para entregar : <code>ft_print_combn.c</code>	
Funções autorizadas : <code>write</code>	

- Escreva uma função que mostre todas as diferentes combinações de `n` números em ordem crescente.
- `n` será tal que: $0 < n < 10$.
- Se `n = 2`, isso resulta em algo do tipo:

```
$>./a.out | cat -e
01, 02, 03, ..., 09, 12, ..., 79, 89$>
```

- Ela deve ser prototipada da seguinte maneira:

```
void ft_print_combn(int n);
```

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.