



C Piscine

C 00

Preâmbulo: Este documento serve como o subject para o módulo C 00 da C Piscine na 42.

Versão: 8

Sumário

I	Instruções	2
II	Instruções de IA	4
III	Prefácio	7
IV	Exercício 00: ft_putchar	8
V	Exercício 01: ft_print_alphabet	9
VI	Exercício 02: ft_print_reverse_alphabet	10
VII	Exercício 03: ft_print_numbers	11
VIII	Exercício 04: ft_is_negative	12
IX	Exercício 05: ft_print_comb	14
X	Exercício 06: ft_print_comb2	15
XI	Exercício 07: ft_putnbr	16
XII	Exercício 08: ft_print_combn	17
XIII	Entrega e Avaliação por Pares	18

Capítulo I

Instruções

- Somente esta página serve como sua referência, não confie em rumores.
- Cuidado! Este documento pode mudar antes da submissão.
- Certifique-se de que você tenha as permissões apropriadas em seus arquivos e diretórios.
- Você deve seguir os **procedimentos de submissão** para todos os seus exercícios.
- Seus exercícios serão verificados e corrigidos por seus colegas de classe.
- Além disso, seus exercícios serão avaliados por um programa chamado **Moulinette**.
- **Moulinette** é meticulosa e estrita em sua avaliação. Ela é totalmente automatizada, e não há como negociar com ela. Para evitar surpresas desagradáveis, seja o mais completo possível.
- **Moulinette** não é tolerante. Se seu código não aderir à Norma, ela não tentará entendê-lo. **Moulinette** utiliza um programa chamado **norminette** para verificar se seus arquivos estão em conformidade com a Norma. TL;DR: Submeter trabalho que não passa na verificação da **norminette** não faz sentido.
- Esses exercícios são organizados por ordem de dificuldade, do mais fácil para o mais difícil. Nós **não** consideraremos um exercício mais difícil concluído com sucesso se um exercício mais fácil não estiver totalmente funcional.
- Usar uma função proibida é considerado trapaça. Trapaceiros recebem uma nota de **-42**, o que é inegociável.
- Você só precisa submeter uma função **main()** se pedirmos especificamente um **programa**.
- **Moulinette** compila com as seguintes flags: **-Wall -Wextra -Werror**, usando **cc**.
- Se seu programa não compilar, você receberá uma nota **0**.

- Você **não pode** deixar **nenhum** arquivo adicional em seu diretório além daqueles especificados na tarefa.
- Tem uma pergunta? Pergunte ao colega à sua direita. Se não, tente o colega à sua esquerda.
- Seu guia de referência se chama **Google / man / Internet / ...**
- Verifique a seção "Piscine C" do fórum na intranet ou na Piscine no Slack.
- Examine cuidadosamente os exemplos. Eles podem conter detalhes cruciais que não são explicitamente declarados na tarefa...
- Por Odin, por Thor! Use seu cérebro!!!



Não se esqueça de incluir o *cabeçalho padrão 42* em cada um dos seus arquivos `.c` e `.h`. A Norminette verifica sua presença de qualquer forma!



A Norminette deve ser executada com a flag `-R`
`CheckForbiddenSourceHeader`. A Moulinette também a usará.

Capítulo II

Instruções de IA

● Contexto

A C Piscine é intensa. É seu primeiro grande desafio na 42 — um mergulho profundo na resolução de problemas, autonomia e comunidade.

Durante essa fase, seu objetivo principal é construir sua base — através do esforço, da repetição e, especialmente, da troca de **aprendizagem entre pares**.

Na era da IA, atalhos são fáceis de encontrar. No entanto, é importante considerar se o uso da IA está realmente ajudando você a crescer — ou simplesmente atrapalhando o desenvolvimento de habilidades reais.

A Piscine também é uma experiência humana — e, por enquanto, nada pode substituí-la. Nem mesmo a IA.

Para uma visão mais completa de nossa posição sobre a IA — como ferramenta de aprendizagem, como parte do currículo de TIC e como uma expectativa crescente no mercado de trabalho — consulte as perguntas frequentes disponíveis na intranet.

● Mensagem principal

- ✎ Construa bases sólidas sem atalhos.
- ✎ Desenvolva realmente habilidades técnicas e de poder.
- ✎ Experimente a verdadeira aprendizagem entre pares, comece a aprender como aprender e resolver novos problemas.
- ✎ A jornada de aprendizagem é mais importante que o resultado.
- ✎ Aprenda sobre os riscos associados à IA e desenvolva práticas de controle eficazes e contramedidas para evitar armadilhas comuns.

● Regras para o aluno:

- Você deve aplicar o raciocínio às tarefas atribuídas, especialmente antes de recorrer à IA.
- Você não deve pedir respostas diretas à IA.
- Você deve aprender sobre a abordagem global da 42 em relação à IA.

● Resultados da fase:

Dentro desta fase fundamental, você obterá os seguintes resultados:

- Obter bases adequadas em tecnologia e codificação.
- Saber por que e como a IA pode ser perigosa durante esta fase.

● Comentários e exemplo:

- Sim, sabemos que a IA existe — e sim, ela pode resolver seus projetos. Mas você está aqui para aprender, não para provar que a IA aprendeu. Não perca seu tempo (nem o nosso) apenas para demonstrar que a IA pode resolver o problema proposto.
- Aprender na 42 não é sobre saber a resposta — é sobre desenvolver a capacidade de encontrá-la. A IA lhe dá a resposta diretamente, mas isso impede você de construir seu próprio raciocínio. E o raciocínio leva tempo, esforço e envolve falhas. O caminho para o sucesso não deve ser fácil.
- Lembre-se de que durante os exames, a IA não está disponível — sem internet, sem smartphones, etc. Você perceberá rapidamente se confiou muito na IA em seu processo de aprendizagem.
- A aprendizagem entre pares o expõe a diferentes ideias e abordagens, melhorando suas habilidades interpessoais e sua capacidade de pensar de forma divergente. Isso é muito mais valioso do que apenas conversar com um bot. Portanto, não seja tímido — converse, faça perguntas e aprenda juntos!
- Sim, a IA fará parte do currículo — tanto como ferramenta de aprendizagem quanto como tópico em si. Você terá até a chance de construir seu próprio software de IA. Para saber mais sobre nossa abordagem crescente, consulte a documentação disponível na intranet.

✓ Boa prática:

Estou travado em um novo conceito. Pergunto a alguém próximo como ele abordou isso. Conversamos por 10 minutos — e de repente, clica. Entendi.

✗ Má prática:

Usei secretamente a IA, copiei um código que parece certo. Durante a avaliação entre pares, não consigo explicar nada. Eu falho. Durante o exame — sem IA — estou travado novamente. Eu falho.

Capítulo III

Prefácio

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

Assim como a maioria dos óleos de peixe, ele contém altos níveis de ácidos graxos ômega-3, incluindo o ácido eicosapentaenoico (EPA) e o ácido docosahexaenoico (DHA). O óleo de fígado de bacalhau é também rico em vitaminas A e D.


Historicamente, tem sido consumido pelo seu teor de vitamina A e vitamina D.

Antigamente, era comum administrá-la a crianças, pois a vitamina D demonstrou prevenir o raquitismo e outros sintomas de deficiência de vitamina D.

Ao contrário do óleo de fígado de bacalhau, C é bom, certifique-se de consumir um pouco!

Capítulo IV

Exercício 00: ft_putchar

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

- Escreva uma função que exiba o caractere passado como parâmetro.
- A função deve ser prototipada da seguinte forma:

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

- Para exibir o caractere, você deve usar a função **write** da seguinte forma:


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



O primeiro atraso para nova tentativa é curto, então não hesite em fazer uma avaliação intermediária para acompanhar seu progresso.

Capítulo V

Exercício 01: ft_print_alphabet

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

- Crie uma função que exiba o alfabeto em minúsculas, em uma única linha, em ordem ascendente, começando pela letra 'a'.
- A função deve ser prototipada da seguinte forma:


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



Não hesite em perguntar aleatoriamente a alguém em seu cluster se tiver alguma dúvida.

Capítulo VI

Exercício 02: ft_print_reverse_alphabet

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

- Crie uma função que exiba o alfabeto em minúsculas, em uma única linha, em ordem descendente, começando pela letra 'z'.
- A função deve ser prototipada da seguinte forma:


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



Envie seu código para o Git regularmente!

Capítulo VII

Exercício 03: ft_print_numbers

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

- Crie uma função que exiba todos os dígitos em uma única linha, em ordem ascendente.
- A função deve ser prototipada da seguinte forma:


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



Colaboração é a chave para o sucesso.

Capítulo VIII

Exercício 04: ft_is_negative

	Exercício : 04
	ft_is_negative
	Pasta de entrega : <i>ex04/</i>
	Arquivos para entregar : ft_is_negative.c
	Funções ou bibliotecas autorizadas : write

- Crie uma função que exiba 'N' ou 'P' dependendo do sinal do inteiro passado como parâmetro.
 - Se **n** for negativo, exiba 'N'.
 - Se **n** for positivo ou zero, exiba 'P'.
- A função deve ser prototipada da seguinte forma:

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



O fracasso faz parte da sua jornada de aprendizado.

Marco Atingido, Continue em Frente!

Você chegou ao fim dos exercícios obrigatórios necessários para validar este projeto.


Agora, cabe a você decidir se continua com os seguintes exercícios opcionais ou se avança para o seu próximo projeto. Ambos os caminhos o exporão a conceitos valiosos mais cedo ou mais tarde.

Para tomar sua decisão, considere os seguintes pontos:

- O primeiro exame foca em programação C. Se você já trabalhou no primeiro projeto C, essa experiência será útil. O mesmo se aplica ao "rush" no final da semana (você logo aprenderá mais sobre isso).
- Sua excelência nesta Piscina é avaliada com base em múltiplos fatores. Completar cada projeto é importante, mas seu progresso geral em todos os projetos da Piscina também desempenha um papel fundamental. Escolha com sabedoria para otimizar seus resultados.
- Você sempre pode revisitar e tentar novamente o mesmo projeto em alguns dias ou semanas, até o final da Piscina.
- Manter-se sincronizado com seus colegas promove uma melhor colaboração.

Capítulo IX

Exercício 05: ft_print_comb

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

- Crie uma função que exiba todas as combinações únicas de três dígitos distintos, com os dígitos dentro de cada combinação e as próprias combinações em ordem crescente.

- Saída esperada:

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

- **987** não é incluído porque **789** já cobre essa combinação.
- **999** não é incluído porque o dígito **9** aparece mais de uma vez.
- A função deve ser prototipada da seguinte forma:


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



Você verificou com seu vizinho da direita?

Capítulo X

Exercício 06: ft_print_comb2

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

- Crie uma função que exiba todas as diferentes combinações de dois números de dois dígitos distintos (XX XX) entre **00** e **99**, listadas em ordem crescente.
- Saída esperada:

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

- A função deve ser prototipada da seguinte forma:


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



Inspire-se nos outros, mas não deixe que eles façam o trabalho por você!

Capítulo XI

Exercício 07: ft_putnbr

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

- Crie uma função que exiba o número passado como parâmetro. A função deve ser capaz de exibir todos os valores possíveis de uma variável do tipo **int**.
- A função deve ser prototipada da seguinte forma:

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


- Exemplo:
 - **ft_putnbr(42);** deve exibir **42**.



Não confie em uma única fonte de informação, sempre execute seus próprios testes, verificações e validações!

Capítulo XII

Exercício 08: ft_print_combn

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

- Crie uma função que exiba todas as combinações únicas de **n** dígitos distintos em ordem crescente, sem repetição.
- O valor de **n** será tal que: $0 < n < 10$.
- Exemplo de saída para **n** = 2:

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

- A função deve ser prototipada da seguinte forma:

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



Você verificou com seu vizinho da esquerda?

Capítulo XIII

Entrega e Avaliação por Pares

Envie seu trabalho para o seu repositório **Git** como de costume. Apenas o trabalho dentro do seu repositório será avaliado durante a defesa. Certifique-se de verificar novamente os nomes dos seus arquivos para garantir que estão corretos.



Você deve enviar apenas os arquivos especificados nas instruções do projeto.