



C Piscine

C 01

Preâmbulo: Este documento é o tema do Módulo C 01 da C Piscine na 42

Versão: 6

Sumário

I	Instruções	2
II	Instruções de IA	4
III	Prefácio	7
IV	Exercício 00 : ft_ft	9
V	Exercício 01 : ft_ultimate_ft	10
VI	Exercício 02 : ft_swap	11
VII	Exercício 03 : ft_div_mod	12
VIII	Exercício 04 : ft_ultimate_div_mod	13
IX	Exercício 05 : ft_putstr	14
X	Exercício 06 : ft_strlen	15
XI	Exercício 07 : ft_rev_int_tab	16
XII	Exercício 08 : ft_sort_int_tab	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 adicionar o *cabeçalho padrão 42* em cada um dos seus arquivos `.c/.h`. A norminette verifica a sua existência 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

Vincent: E sabe como eles chamam um... um... um Quarterão com Queijo em Paris?

Jules: Eles não chamam de Quarterão com Queijo?

Vincent: Não, cara, eles têm o sistema métrico. Eles não saberiam o que diabos é um Quarterão.

Jules: Então como eles chamam?

Vincent: Eles chamam de Royale com queijo.

Jules: Um Royale com queijo. Como eles chamam um Big Mac?

Vincent: Bem, um Big Mac é um Big Mac, mas eles chamam de "Le Big Mac".

Jules: "Le Big-Mac." Ha ha ha ha. Como eles chamam um Whopper?

Vincent: Não sei, não fui ao Burger King.

Pelo menos um dos seguintes exercícios não tem nada a ver com um Royale com queijo.

Limite de hoje

O limite de validação para este projeto é de 50%.

Cabe a você determinar quais exercícios permitem que você atinja esse limite, e se você deseja completar mais.

Capítulo IV

Exercício 00 : ft_ft

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

- Crie uma função que receba um ponteiro para um inteiro como parâmetro e defina o valor desse inteiro como "42".
- A função deve ser prototipada da seguinte forma:

```
void        ft_ft(int *nbr);
```

Capítulo V

Exercício 01 : ft_ultimate_ft

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

- Crie uma função que receba um ponteiro para um inteiro como parâmetro e defina o valor desse inteiro como “42”.
- A função deve ser prototipada da seguinte forma:

```
void      ft_ultimate_ft(int *****nbr);
```

Capítulo VI

Exercício 02 : ft_swap

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

- Crie uma função que troque os valores de dois inteiros usando seus endereços recebidos como parâmetros.
- A função deve ser prototipada da seguinte forma:

```
void    ft_swap(int *a, int *b);
```

Capítulo VII

Exercício 03 : ft_div_mod

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

- Crie uma função **ft_div_mod** com o seguinte protótipo:

```
void      ft_div_mod(int a, int b, int *div, int *mod);
```

- Esta função divide ‘a’ por ‘b’ e armazena o resultado no inteiro apontado por ‘div’. Ela também armazena o resto da divisão de ‘a’ por ‘b’ no inteiro apontado por ‘mod’.

Capítulo VIII

Exercício 04 : ft_ultimate_div_mod

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

- Crie uma função **ft_ultimate_div_mod** com o seguinte protótipo:

```
void      ft_ultimate_div_mod(int *a, int *b);
```

- Esta função divide o valor apontado por ‘a’ pelo valor apontado por ‘b’. O resultado da divisão é armazenado no inteiro apontado por ‘a’, enquanto o resto é armazenado no inteiro apontado por ‘b’.

Capítulo IX

Exercício 05 : ft_putstr

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

- Crie uma função que exiba uma string de caracteres na saída padrão.
- A função deve ser prototipada da seguinte forma:

```
void      ft_putstr(char *str);
```

Capítulo X

Exercício 06 : ft_strlen

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

- Crie uma função que conte e retorne o número de caracteres em uma string.
- A função deve ser prototipada da seguinte forma:

```
int      ft_strlen(char *str);
```

Capítulo XI

Exercício 07 : ft_rev_int_tab

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

- Crie uma função que inverta um array de inteiros dado (o primeiro elemento se torna o último, e assim por diante).
- A função recebe dois argumentos: um ponteiro para um int e o número de elementos no array.
- A função deve ser prototipada da seguinte forma:

```
void    ft_rev_int_tab(int *tab, int size);
```

Capítulo XII

Exercício 08 : ft_sort_int_tab

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

- Crie uma função que ordene um array de inteiros em ordem crescente.
- A função recebe dois argumentos: um ponteiro para um int e o número de elementos no array.
- A função deve ser prototipada da seguinte forma:

```
void    ft_sort_int_tab(int *tab, int size);
```

Capítulo XIII

Entrega e avaliação por pares

Envie sua tarefa 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 arquivos para garantir que estão corretos.



Você deve enviar apenas os arquivos explicitamente exigidos pelas instruções do projeto.