



Piscina C

C 09

Sumário: Este documento é o tema do módulo C 09 da Piscina C da 42.

Conteúdo

I	Instruções	2
II	Preâmbulo	4
III	Exercício 00 : libft	5
IV	Exercício 01 : Makefile	6
V	Exercício 02 : ft_split	8

Capítulo I

Instruções

- Somente esta página servirá de referência, não confie nos boatos.
- Releia bem o tema antes de entregar seus exercícios. A qualquer momento o tema pode mudar.
- Atenção aos direitos de seus arquivos e suas pastas.
- Você deve seguir o 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 por 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 tólice 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 pedirmos um programa.
- A Moulinette compila com as sinalizações -Wall -Wextra -Werror, e utiliza **gcc**.
- 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 Piscina do seu Intra, assim como no slack 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.



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

Capítulo II

Preâmbulo

Diálogo do filme O Grande Lebowski:

Walter: Desculpa, Smokey. Você pisou na linha: falta.
Smokey: Mentira! Marca oito, Cara...
Walter: Me desculpa. Marca zero. Próxima rodada.
Smokey: Deixa de ser chato, Walter. Marca oito, Cara...
Walter: Smokey, não estamos no Vietnã. Estamos no boliche. A gente joga com regras...
O Cara: Vai, Walter, não enche. O pé dele escorregou um pouquinho na linha... É só um jogo!
Walter: Sim. E é um jogo de campeonato, decide quem vai pra próxima fase, não é?
Smokey: É, mas eu não...
Walter: Não é verdade?
Smokey: Sim, mas eu não pisei... Vai! Cara, marca oito.
Walter: [saca sua arma] Smokey, meu amigo, se você nunca sentiu dor, você vai sentir...
Cara: Walter, não seja idiota!
Walter [ameaçador] Vai, marca um oito que você vai ver!
Smokey: Eu não...
Walter: Você vai sentir dor, Smokey!
Smokey: Cara? É o seu parceiro...
Walter: O mundo todo tá FICANDO LOUCO? [Ele se levanta] EU SOU O ÚNICO QUE LIGA PRAS REGRAS? MARCA ZERO!
Cara: Eles vão chamar a polícia, Walter, guarda essa...
Walter: [aponta a arma para Smokey] MARCA ZERO!
Cara: Guarda a arma, Walter...
Smokey: ...
Cara: Walter...
Walter: [engatilha a arma] VOCÊ ACHA QUE EU TÔ BRINCANDO?
MARCA ZERO!
Smokey: Tá bom, marquei zero... Tá feliz? Seu louco!
Walter: [se acalma] ... É um jogo de campeonato.

Capítulo III

Exercício 00 : libft

	Exercício : 00
	libft
Pasta de entrega : <i>ex00/</i>	
Arquivos para entregar : <i>libft_creator.sh</i> , <i>ft_putchar.c</i> , <i>ft_swap.c</i> , <i>ft_putstr.c</i> , <i>ft_strlen.c</i> , <i>ft_strcmp.c</i>	
Funções autorizadas : <i>write</i>	

- Crie sua biblioteca **ft**. Ela vai se chamar **libft.a**.
- Um shell script chamado **libft_creator.sh** vai compilar adequadamente os arquivos de código-fonte e criar sua biblioteca.
- Essa biblioteca deve conter todas as seguintes funções:

```
void    ft_putchar(char c);
void    ft_swap(int *a, int *b);
void    ft_putstr(char *str);
int     ft_strlen(char *str);
int     ft_strcmp(char *s1, char *s2);
```

- Vamos executar o seguinte comando:

```
sh libft_creator.sh
```

Capítulo IV

Exercício 01 : Makefile

	Exercício : 01
	Makefile
Pasta de entrega :	<i>ex01/</i>
Arquivos para entregar :	Makefile
Funções autorizadas :	Nenhuma

- Escreva o **Makefile** que compile uma biblioteca **libft.a**.
- Seu makefile deve mostrar claramente cada comando que efetuar sem enrolação.
- Seu makefile não deve efetuar comandos inúteis.
- O **Makefile** vai procurar os arquivos de código-fonte na pasta **srcs**.
- Esses arquivos de código-fonte serão: **ft_putchar.c**, **ft_swap.c**, **ft_putstr.c**, **ft_strlen.c**, **ft_strcmp.c**
- O **Makefile** vai procurar os arquivo de cabeçalhos na pasta **includes**.
- Esses arquivos de cabeçalhos serão: **ft.h**
- Ele deverá compilar seus arquivos c utilizando **gcc** e as opções de sinalizações **-Wall** -**Wextra** -**Werror** nessa ordem.
- A lib estará na origem do exercício.
- Os arquivos **.o** deverão estar ao lado do seu respectivo arquivo **.c**.
- O **Makefile** deverá também implementar as regras **clean**, **fclean**, **re**, a regra **all** e, claro, **libft.a**.
- Executar apenas **make** deve ser equivalente a **make all**.

- A regra `all` deverá agir como `make libft.a`.
- A regra `clean` deverá eliminar todos os arquivos temporários gerados.
- A regra `fclean` é o equivalente a um `make clean` e também apaga o binário criado no `make all`.
- A regra `re` é o equivalente a um `make fclean` então um `make all`.
- Seu makefile não deve recompilar os arquivos desnecessariamente.
- Nós vamos pegar só seu Makefile e testar com nossos arquivos.



Cuidado com os wildcards!

Capítulo V

Exercício 02 : ft_split

	Exercício : 02
	ft_split
Pasta de entrega :	<i>ex02/</i>
Arquivos para entregar :	ft_split.c
Funções autorizadas :	malloc

- Escreva uma função que divida uma cadeia de caracteres em função de uma outra cadeia de caracteres.
- Será necessário utilizar cada caractere da cadeia **charset** como separador.
- A função retorna uma matriz na qual cada célula contém o endereço de uma cadeia de caracteres compreendida entre dois separadores. O último elemento da matriz deverá ser igual a 0 para marcar o fim da tableau.
- Não deve haver cadeias vazias na sua matriz. Tire as conclusões necessárias.
- A cadeia que será transmitida não será editável.
- Ela deverá ser prototipada da seguinte maneira:

```
char **ft_split(char *str, char *charset);
```