



Piscina C

C 11

Sumário: Este documento é o tema do módulo C 11 da Piscina C da 42.

Versão: 6.2

Conteúdo

I	Instruções	2
II	Preâmbulo	3
III	Exercício 00 : ft_foreach	5
IV	Exercice 01 : ft_map	6
V	Exercício 02 : ft_any	7
VI	Exercício 03 : ft_count_if	8
VII	Exercício 04 : ft_is_sort	9
VIII	Exercício 05 : do-op	10
IX	Exercício 06 : ft_sort_string_tab	12
X	Exercício 07 : ft_advanced_sort_string_tab	13
XI	Submissão e avaliação	14

Capítulo I

Instruções

- Somente este documento servirá de referência; não confie nos boatos.
- Leia bem o enunciado antes de entregar seus exercícios. A qualquer momento pode haver alterações.
- Os exercícios são ordenados precisamente do mais simples ao mais complexo. Em caso algum consideraremos um exercício mais complexo se outro mais simples não tiver sido perfeitamente realizado.
- Tenha atenção aos direitos dos seus ficheiros pastas.
- Deverá seguir o procedimento de entrega para todos os exercícios.
- Os seus exercícios serão corrigidos pelos seus colegas de piscine.
- Além dos seus colegas, a Moulinette também corrigirá os seus exercícios.
- A Moulinette é extremamente rígida na sua avaliação. É completamente automatizada, e é impossível discutir a sua nota com ela. Portanto, sejam rigorosos!
- Os exercícios de shell devem ser executados com `/bin/sh`.
- Não deve deixar no repositório de entrega nenhum outros ficheiros além daqueles explicitamente especificados pelos enunciados dos exercícios.
- Tem alguma dúvida? Pergunte ao seu vizinho da direita. Tente, também, com o seu vizinho da esquerda.
- A bibliografia para consulta chama-se `Google / man / Internet /`
- Considere discutir os exercícios no Slack da sua piscine!
- Leia atentamente os exemplos: podem demonstrar coisas que não estão especificadas no enunciado...

Capítulo II

Preâmbulo

Here's a little story :

(1982, California) Larry Walters of Los Angeles is one of the few to contend for the Darwin Awards and live to tell the tale. "I have fulfilled my 20-year dream," said Walters, a former truck driver for a company that makes TV commercials. "I'm staying on the ground. I've proved the thing works."

Larry's boyhood dream was to fly. But fates conspired to keep him from his dream. He joined the Air Force, but his poor eyesight disqualified him from the job of pilot. After he was discharged from the military, he sat in his backyard watching jets fly overhead.

He hatched his weather balloon scheme while sitting outside in his "extremely comfortable" Sears lawnchair. He purchased 45 weather balloons from an Army-Navy surplus store, tied them to his tethered lawnchair (dubbed the Inspiration I) and filled the four-foot diameter balloons with helium. Then, armed with some sandwiches, Miller Lite, and a pellet gun, he strapped himself into his lawnchair. He figured he would shoot to pop a few of the many balloons when it was time to descend.

Larry planned to sever the anchor and lazily float to a height of about 30 feet above the backyard, where he would enjoy a few hours of flight before coming back down. But things didn't work out quite as Larry planned.

When his friends cut the cord anchoring the lawnchair to his Jeep, he did not float lazily up to 30 feet. Instead he streaked into the LA sky as if shot from a cannon, pulled by the lift of 45 helium balloons, holding 33 cubic feet of helium each.

He didn't level off at 100 feet, nor did he level off at 1000 feet. After climbing and climbing, he leveled off at 16,000 feet.

At that height he felt he couldn't risk shooting any of the balloons, lest he unbalance the load and really find himself in trouble. So he stayed there, drifting cold and frightened with his beer and sandwiches, for more than 14 hours. He crossed the primary approach corridor of LAX, where startled Trans World Airlines and Delta Airlines pilots radioed in reports

of the strange sight.


Eventually he gathered the nerve to shoot a few balloons, and slowly descended. The hanging tethers tangled and caught in a power line, blacking out a Long Beach neighborhood for 20 minutes. Larry climbed to safety, where he was arrested by waiting members of the LAPD. As he was led away in handcuffs, a reporter dispatched to cover the daring rescue asked him why he had done it. Larry replied nonchalantly, "A man can't just sit around."

The Federal Aviation Administration was not amused. Safety Inspector Neal Savoy said, "We know he broke some part of the Federal Aviation Act, and as soon as we decide which part it is, a charge will be filed."

The moral of this story is Larry Walters should have stay on his chair and learn C....

Capítulo III

Exercício 00 : ft_foreach

	Exercício : 00
	ft_foreach
	Pasta de entrega : ex00/
	Ficheiros para entregar : ft_foreach.c
	Funções autorizadas : Nenhuma

- Escreve uma função `ft_foreach` que, para uma array de inteiros, aplique uma função sobre todos os elementos do array. Essa função será aplicada na ordem do array.
- A função deve ser prototipada da seguinte maneira:


```
void      ft_foreach(int *tab, int length, void(*f)(int));
```

- Por exemplo, a função `ft_foreach` poderá ser chamada da seguinte forma para mostrar o conjunto de inteiros do array:

```
ft_foreach(tab, 1337, &ft_putnbr);
```

Capítulo IV

Exercice 01 : ft_map


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

- Escreve uma função `ft_map` que, para uma array de inteiros, aplicará uma função sobre todos os elementos do array (em sequência) e retornará um array de todos os valores retornados.
- A função será aplicada na ordem do array.
- A função deve ser prototipada da seguinte maneira:

```
int      *ft_map(int *tab, int length, int(*f)(int));
```

Capítulo V

Exercício 02 : ft_any

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


- Escreve uma função **ft_any** que retornará 1 se pelo menos um elemento do array retornar algo diferente de 0 ao ser passado para a função **f**. Caso contrário, retorna 0.
- A função será aplicada na ordem do array.
- A função deve ser prototipada da seguinte maneira:

```
int ft_any(char **tab, int(*f)(char*));
```

- O array termina em nulo.

Capítulo VI

Exercício 03 : ft_count_if


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

- Escreve uma função `ft_count_if` que retornará o número de elementos do array que, ao serem passados para a função `f`, não retornam 0.
- A função será aplicada na ordem do array.
- A função deve ser prototipada da seguinte maneira:

```
int ft_count_if(char **tab, int length, int(*f)(char*));
```

Capítulo VII

Exercício 04 : ft_is_sort


	Exercício : 04
	ft_is_sort
	Pasta de entrega : ex04/
	Ficheiros para entregar : ft_is_sort.c
	Funções autorizadas : Nenhuma

- Escreve uma função `ft_is_sort` que retornará 1 se o array estiver ordenado e 0 no caso contrário.
- A função passada como parâmetro retornará um inteiro negativo se o primeiro argumento for inferior ao segundo, 0 se forem iguais e um inteiro positivo caso contrário.
- A função deve ser prototipada da seguinte maneira:

```
int ft_is_sort(int *tab, int length, int(*f)(int, int));
```

Capítulo VIII

Exercício 05 : do-op

	Exercício : 05
do-op	
Pasta de entrega : <i>ex05/</i>	
Ficheiros para entregar : Todos os arquivos necessários para o seu programa	
Funções autorizadas : <code>write</code>	

- Escreve um programa chamado `do-op`.
- O programa deve ser executado com três argumentos: `do-op valor1 operador valor2`
- Exemplo:

```
$>./do-op 42 "+" 21
63
$>
```

- Deves utilizar um array de ponteiros para função de forma a chamar a função correspondente ao **operador**.
- Em caso de operador desconhecido, seu programa deverá mostrar 0.
- Se o número de argumentos não estiver correto, **do-op** não mostra nada.
- Seu programa deve aceitar e mostrar o resultado com os seguintes operadores: `'+' '-' '/' '*'` e `'%'`
- Em caso de divisão por 0, seu programa deve mostrar:

```
Stop : division by zero
```

- Em caso de módulo por 0, seu programa deve mostrar:


```
Stop : modulo by zero
```

- Veja um exemplo de testes da Moulinette :

```
$> make clean
$> make
$> ./do-op
$> ./do-op 1 + 1
2
$> ./do-op 42amis - ---20toto12
62
$> ./do-op 1 p 1
0
$> ./do-op 1 + toto3
1
$>
$> ./do-op toto3 + 4
4
$> ./do-op foo plus bar
0
$> ./do-op 25 / 0
Stop : division by zero
$> ./do-op 25 % 0
Stop : modulo by zero
$>
```

Capítulo IX

Exercício 06 : ft_sort_string_tab


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

- Escreve uma função `ft_sort_string_tab` que classifica por ordem `ascii` as `strings`.
- `tab` termina em nulo
- A ordenação será realizada ao trocar os ponteiros do array.
- Deve ser prototipada da seguinte maneira:

```
void ft_sort_string_tab(char **tab);
```

Capítulo X

Exercício 07 : ft_advanced_sort_string_tab

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

- Escreve a função `ft_advanced_sort_string_tab` que ordena conforme o retorno da função passada como parâmetro
- A ordenação será realizada ao trocar os ponteiros da array.
- `tab` termina em nulo
- Deve ser prototipada da seguinte maneira:

```
void ft_advanced_sort_string_tab(char **tab, int(*cmp)(char *, char *));
```



Se chamares o `ft_advanced_sort_string_tab()` como segundo parâmetro `ft_strcmp` retornará o mesmo resultado que `ft_sort_string_tab()`.

Capítulo XI

Submissão e avaliação

Entrega o teu trabalho no teu repositório `Git`, como habitual. Apenas o trabalho dentro do teu repositório será avaliado durante a defesa. Não hesites em confirmar os nomes dos teus ficheiros para ter a certeza que estão corretos.



Apenas precisas de entregar os ficheiros pedidos no enunciado deste projeto.