



# Piscina C

## Shell 01

*Sumário: Este documento é o enunciado do módulo Shell 01 da Piscina C da 42.*

*Versão: 7.5*

# Conteúdo

<b>I</b>	<b>Instruções</b>	<b>2</b>
<b>II</b>	<b>Preâmbulo</b>	<b>3</b>
<b>III</b>	<b>Exercício 00: Exame</b>	<b>4</b>
<b>IV</b>	<b>Exercício 01: print_groups</b>	<b>5</b>
<b>V</b>	<b>Exercício 02: find_sh</b>	<b>6</b>
<b>VI</b>	<b>Exercício 03: count_files</b>	<b>7</b>
<b>VII</b>	<b>Exercício 04: MAC</b>	<b>8</b>
<b>VIII</b>	<b>Exercício 05: Consegue criar?</b>	<b>9</b>
<b>IX</b>	<b>Exercício 06: Skip</b>	<b>11</b>
<b>X</b>	<b>Exercício 07: r_dwssap</b>	<b>12</b>
<b>XI</b>	<b>Exercício 08: add_chelou</b>	<b>13</b>
<b>XII</b>	<b>Submissão e avaliação</b>	<b>14</b>

# Capítulo I

## Instruções

- Somente este documento servirá de referência; não confie nos boatos.
- Leia bem o enunciado antes de entregar os 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 e 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, seja rigoroso!
- Os exercícios de shell devem ser executados com `/bin/sh`.
- Não deve deixar no repositório de entrega nenhum outro ficheiro além daqueles explicitamente especificados pelo enunciado 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

Eis o que a Wikipédia tem a dizer sobre a lontra:

A lontra europeia (*Lutra lutra*), também conhecida como lontra comum nos países europeus onde está presente, é um mamífero carnívoro semiaquático e predominantemente noturno, da família dos Mustelídeos (subfamília Lutrinae). Ela é uma das três espécies de lontras pertencente ao gênero *Lutra*. Na França, não se encontra outra espécie de lontra.

Sua altura é em média 30 cm. Sua pelagem, castanho-escura, é composta de duas camadas: a primeira de pelos curtos, muito finos, densos e lanosos; e a segunda de pelos longos, lisos, brilhantes e impermeáveis.

Excelente nadadora, ela tem patas palmadas, corpo alongado (60 a 80 cm em média, com uma cauda de 30 a 40 cm de comprimento grossa na base e mais fina nas extremidades), e um peso que varia de 5 a 15 kg.


Ela vive à margem de cursos de água (córregos, riachos e até rios), até uma altitude de 1.300 m, nos pântanos e às vezes nas costas marinhas. Ela geralmente é solitária, ocupando um território de 5 a 15 km de margem ao longo de um curso de água (às vezes maior) ou de 20 a 30 km<sup>2</sup> em zonas de pântanos. Ela pega regularmente as mesmas passagens nas margens do rio para entrar na água: os "fluxos". Quando sai da água, ela rola na relva para limpar o pelo em áreas reconhecíveis de grama baixa, chamadas "ninhos".

Ela faz sua toca entre as raízes das árvores nas margens dos cursos de água ou em outras cavidades (cavidades rochosas, troncos ocos, tocas de outros tipos). A toca geralmente contém uma entrada mais ou menos escondida abaixo do nível da água e uma abertura para ventilação.

A lontra é bonitinha!

# Capítulo III


## Exercício 00: Exame

	Exercice : 00
Exam machine	

- Durante esta semana, poderá inscrever-se para o exame de sexta na Agenda da intra; reflita sobre isso.
- Depois da reflexão e eventual inscrição na agenda, verifique que foi feita a inscrição no projeto C Piscine Exam 00.
- Verifique, já agora, que está inscrito para o exame de sexta-feira (tanto no evento como no projeto!).
- Verifique que verificou que está inscrito (no evento, e no projeto! Sim, nos dois!).

# Capítulo IV

## Exercício 01: print\_groups

	Exercício : 01
	print_groups.sh
	Pasta de entrega : ex01/
	Ficheiros para entregar : print_groups.sh
	Funções autorizadas : Nenhuma

- Escreva uma linha de comando que determine e mostre a lista de grupos dos quais o login especificado na *environment variable* FT\_USER é membro, separando-os por vírgulas sem espaços.

- Exemplos:

- com FT\_USER=nours, o resultado é "god,root,admin,master,nours,bocal"(sem aspas)

```
$>./print_groups.sh  
god,root,admin,master,nours,bocal$>
```

- com FT\_USER=daemon, o resultado é "daemon,bin"(sem aspas)

```
$>./print_groups.sh  
daemon,bin$>
```




man id



Inspira-te nos outros, mas não os deixes fazer o teu trabalho.

# Capítulo V

## Exercício 02: find\_sh

	Exercício : 02
	find_sh.sh
	Pasta de entrega : <i>ex02/</i>
	Ficheiros para entregar : <b>find_sh.sh</b>
	Funções autorizadas : <b>Nenhuma</b>

- Escreva uma linha de comando que procure, na pasta atual e em todas as suas subpastas, todos os ficheiros cujos nomes terminam com `.sh` (sem as aspas) e que só exiba os seus nomes, sem o `.sh`.
- Exemplo:


```
$>./find_sh.sh | cat -e
find_sh$
file1$
file2$
file3$
$>
```



Não acredites em qualquer fonte de informação: faz sempre os teus próprios testes, controlos e verificações.

# Capítulo VI

## Exercício 03: count\_files

	Exercício : 03
	count_files.sh
	Pasta de entrega : <i>ex03/</i>
	Ficheiros para entregar : <b>count_files.sh</b>
	Funções autorizadas : <b>Nenhuma</b>

- Escreva uma linha de comando que mostre o número de ficheiros e de pastas dentro da pasta atual e todas as suas subpastas, incluindo o `.` da pasta inicial.
- Exemplo de saída:

```
$>./count_files.sh | cat -e  
42$  
$>
```




O fracasso faz parte da tua jornada de aprendizagem.



# Capítulo VII

## Exercício 04: MAC

	Exercício : 04
	MAC.sh
	Pasta de entrega : <i>ex04/</i>
	Ficheiros para entregar : <i>MAC.sh</i>
	Funções autorizadas : <i>Nenhuma</i>

- Escreva uma linha de comando que mostre os endereços MAC da sua máquina. Cada endereço deve ser seguido de uma quebra de linha.




`man ifconfig`



A colaboração é a chave para o sucesso.

# Capítulo VIII

## Exercício 05: Consegue criar?

	Exercício : 05
Consegue criar?	
Pasta de entrega : <i>ex05/</i>	
Ficheiros para entregar : "\?*\$*'MaRViN'*\$?\"	
Funções autorizadas : Nenhuma	

- Crie um arquivo contendo apenas 42 e **nada** mais.
- Deve chamar-se:

```
"\?*$*'MaRViN'*$?\"
```

- Exemplo:

```
$>ls -lRa *MaRV* | cat -e
-rw---xr-- 1 75355 32015 2 Oct 2 12:21 "\?*$*'MaRViN'*$?\"$
$>
```

## Objetivo alcançado, continua!

Chegaste ao fim dos exercícios obrigatórios para validar este projeto.


Cabe-te a ti decidir se queres continuar com os exercícios opcionais seguintes ou passar para o teu próximo projeto. Ambos os caminhos te proporcionarão conhecimentos úteis mais cedo ou mais tarde.

Para fazer a tua escolha, por favor, considera os seguintes elementos:

- O primeiro exame é sobre programação em C. Portanto, é possível que já tenhas passado pela primeira experiência de um projeto em C antes. O mesmo acontece com o RUSH no final da semana (em breve aprenderás sobre o RUSH).
- A tua excelência nesta Piscine será avaliada com base em vários fatores. A conclusão de cada projeto é um deles, mas o progresso geral ao longo de toda a lista de projetos da Piscine é outro. Escolhe sabiamente para otimizar os teus resultados.
- Será sempre possível tentar o mesmo projeto novamente em alguns dias/semanas, até ao final da Piscine.
- Manteres-te sincronizado com os teus colegas garante uma melhor colaboração.

# Capítulo IX

## Exercício 06: Skip

	Exercício : 06
skip.sh	
Pasta de entrega : <i>ex06/</i>	
Ficheiros para entregar : <b>skip.sh</b>	
Funções autorizadas : Nenhuma	

- Escreva uma linha de comando que exiba um `ls -l` uma linha a cada duas (linha sim, linha não), a partir da primeira.
- Exemplo:

```
$>ls -l | cat -e
total 4$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:46 skip.sh$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:41 tata$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:41 titi$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:41 toto$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:41 tutu$
$>
```


```
$>./skip.sh | cat -e
total 4$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:41 tata$
-rw-rw-r-- 1 eagle eagle ** ** 15 15:41 toto$
$>
```



Git push regularmente.

# Capítulo X

## Exercício 07: r\_dwssap

	Exercício : 07
	r_dwssap.sh
	Pasta de entrega : <i>ex07/</i>
	Ficheiros para entregar : <b>r_dwssap.sh</b>
	Funções autorizadas : Nenhuma

- Escreva uma linha de comando que exiba o *output* de um `cat /etc/passwd`, retirando os comentários, uma linha a cada duas começando pela segunda, invertendo cada login e ordenando em ordem alfabética inversa, mantendo apenas os logins compreendidos entre, e incluindo, FT\_LINE1 e FT\_LINE2, separados por , , e terminando com ..
- Exemplo: Entre as linhas 7 e 15, o resultado será algo como:

```
$> ./r_dwssap.sh
sstq_, sorebrek_brk_, soibten_, sergtsop_, scodved_, rlaxcm_, rgmecived_, revreswodniw_, revressta_
.$>
```




Respeite estritamente a ordem do enunciado.



Verificaste com o teu vizinho da esquerda?

## Exercício 08: add\_chelou

	Exercício : 08
add_chelou.sh	
Pasta de entrega : <i>ex08/</i>	
Ficheiros para entregar : <b>add_chelou.sh</b>	
Funções autorizadas : <b>Nenhuma</b>	

- Escreva uma linha de comando que recebe os números contidos nas variáveis `FT_NBR1` de base `'\"?!` e `FT_NBR2` de base `mrdoc`, e que exiba a soma dos dois em base `gtaio`  
`luSnemf`

- Exemplo 1:

```
FT_NBR1=\ ' ? " \ " ' \
FT_NBR2=r cr d m d d d
```

- A soma é:

Salut

- Exemplo 2:

```
FT_NBR1=\"\"!\"\"!\"\"!\"\"!\"\"!\"\"!  
FT_NBR2=dcrcmcmooododmrrrmorcmcrmomomo
```

- A soma é:

## Segmentation Fault



Não hesites em escolher alguém aleatoriamente no Cluster para fazer uma pergunta.

# Capítulo XII

## Submissão e avaliação

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



É apenas necessário entregar os ficheiros pedidos para este projeto