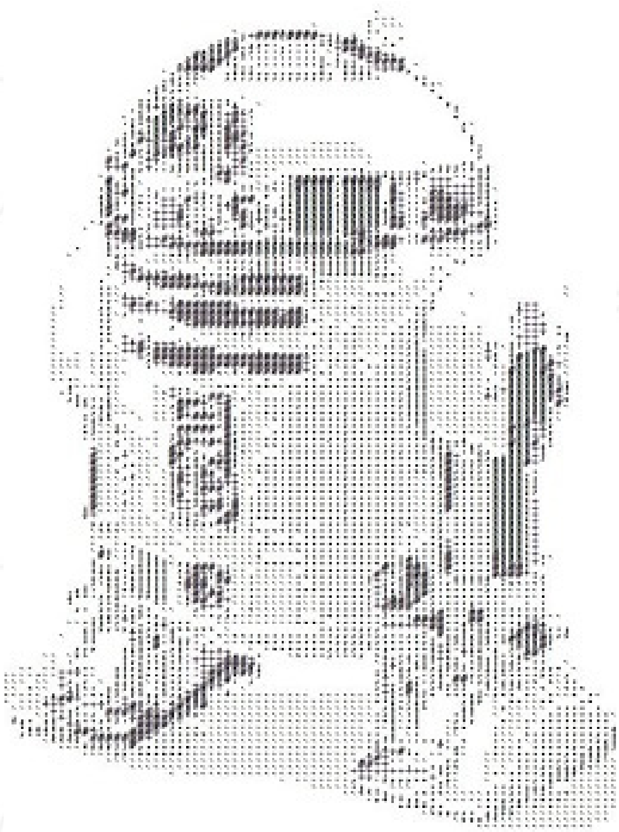
Coding in action labs
00

Contents

Chapter I

- Only this page will serve as reference; do not trust rumors.
- Watch out! This document could potentially change up to an hour before submission.
- These exercises are carefully laid out by order of difficulty - from easiest to hardest. We will not take into account a successfully completed harder exercise if an easier one is not perfectly functional.
- Make sure you have the appropriate permissions on your files and directories.
- You have to follow the submission procedures for every exercise.
- Your exercises will be checked and graded by your fellow classmates.
- On top of that, your exercises will be checked and graded by a program called Moulinette.
- Moulinette is very meticulous and strict in its evaluation of your work. It is entirely automated and there is no way to negotiate with it. So if you want to avoid bad surprises, be as thorough as possible.
- Exercises in Shell must be executable with /bin/sh.
- You cannot leave any additional file in your directory than those specified in the subject.
- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.
- Your reference guide is called Google / man / the Internet / ....
- Examine the examples thoroughly. They could very well call for details that are not explicitly mentioned in the subject...

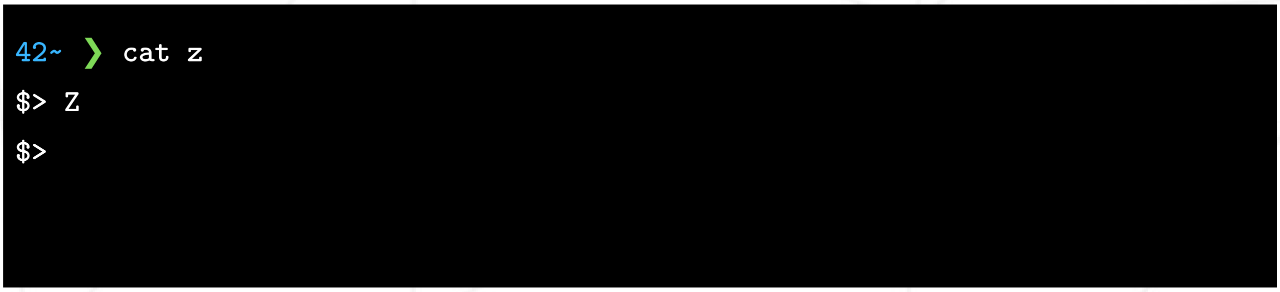


Chapter II

Exercice 00: Z

	Exercise 00
Only the best know how to display Z	
Turn-in directory: *ex*00/	
File to turn in: z	
Allowed external functions: none	

- Create a file called z that returns "Z", followed by a new line, whenever the command cat is used on it.



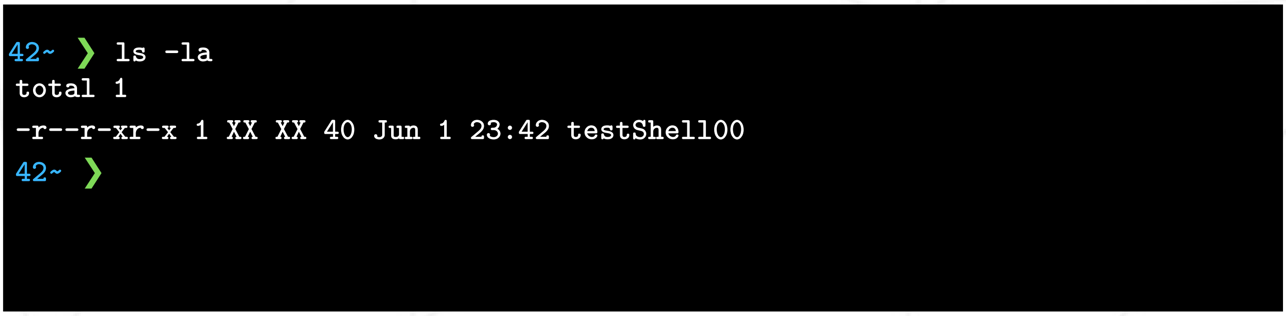
```
42~ ❯ cat z
$> Z
$>
```

Chapter III

Exercice 01: testShell00

	Exercise 01
What are attributes anyway ?	
Turn-in directory: *ex*01/	
File to turn in: testShell00.tar	
Allowed external functions: none	

- Create a file called testShell00 in your submission directory.
- Figure out a way for the output to look like this.
  (except for the "total 1" line):



```
42~ ❯ ls -la
total 1
-r--r-xr-x 1 XX XX 40 Jun 1 23:42 testShell00
42~ ❯
```

Don't worry about what you've got instead of "XX".

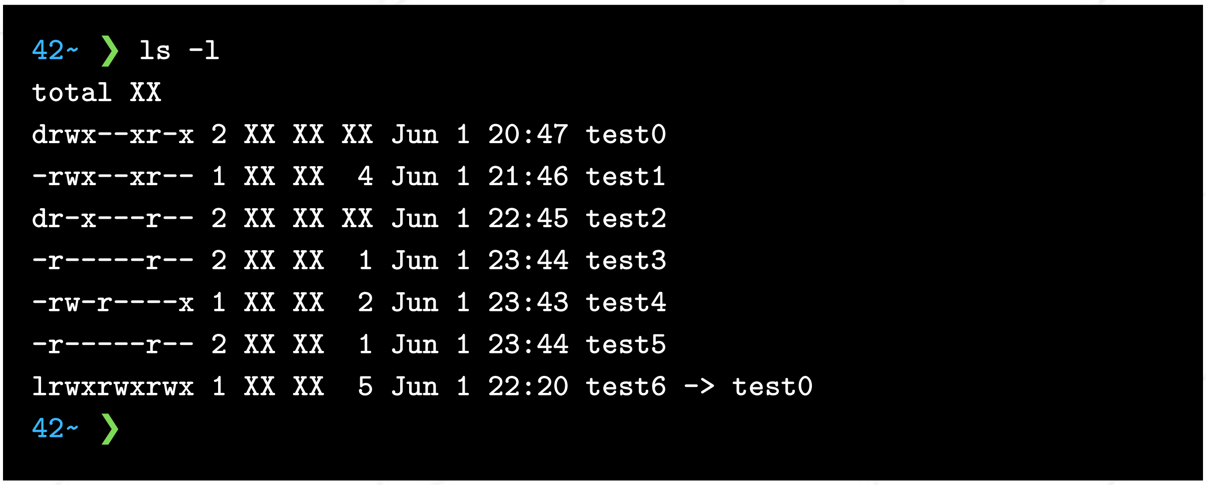
A year will be accepted instead of the time, on the timestamp of the file.

Chapter IV

Exercice 02: Oh yeah, mooore...

Exercise 02
Oh yeah, mooore...
Turn-in directory: ex02/
File to turn in: exo2.tar
Allowed external functions: none

- Create the following files and directories. Do what's necessary so that when you use the ls -l command in your directory, the output will looks like this :



```
42~ ❯ ls -l
total XX
drwx--xr-x 2 XX XX XX Jun 1 20:47 test0
-rwx--xr-- 1 XX XX  4 Jun 1 21:46 test1
dr-x---r-- 2 XX XX XX Jun 1 22:45 test2
-r-----r-- 2 XX XX  1 Jun 1 23:44 test3
-rw-r----x 1 XX XX  2 Jun 1 23:43 test4
-r-----r-- 2 XX XX  1 Jun 1 23:44 test5
lrwxrwxrwx 1 XX XX  5 Jun 1 22:20 test6 -> test0
42~ ❯
```

- Once you've done that, run tar -cf exo2.tar * to create the file to be submitted.

Don't worry about what you've got instead of "XX".

A year will be accepted instead of the time, on the timestamp of the files.

Chapter V

Exercice 03: SSH me!

	Exercise 03
SSH me!	
Turn-in directory: *ex*03/	
File to turn in: id_rsa_pub	
Allowed external functions: none	

- Create your own SSH key. Once it is done:
  - Add your public key to your repository, in a file name id_rsa_pub
  - Update your ssh key on the intranet. This will allow you to push the repository to our git server.

> The file's name was not chosen randomly.

> Make sure you understand the difference between the public key and the private key.

Chapter VI

Exercice 04: midLS

<table>
<tr><td></td><td>Exercise 04</td></tr>
<tr><td colspan="2">midLS</td></tr>
<tr><td colspan="2">Turn-in directory: ex04/</td></tr>
<tr><td colspan="2">File to turn in: midLS</td></tr>
<tr><td colspan="2">Allowed external functions: None</td></tr>
</table>

- In a midLS file, place the command line that will list all files and directories in your current directory (except for hidden files or any file that starts by a dot - yes, that includes double-dots), separated by a comma, by order of modification date. Make sure the directory's names are followed by a slash character.

What has not been asked for should not be done!

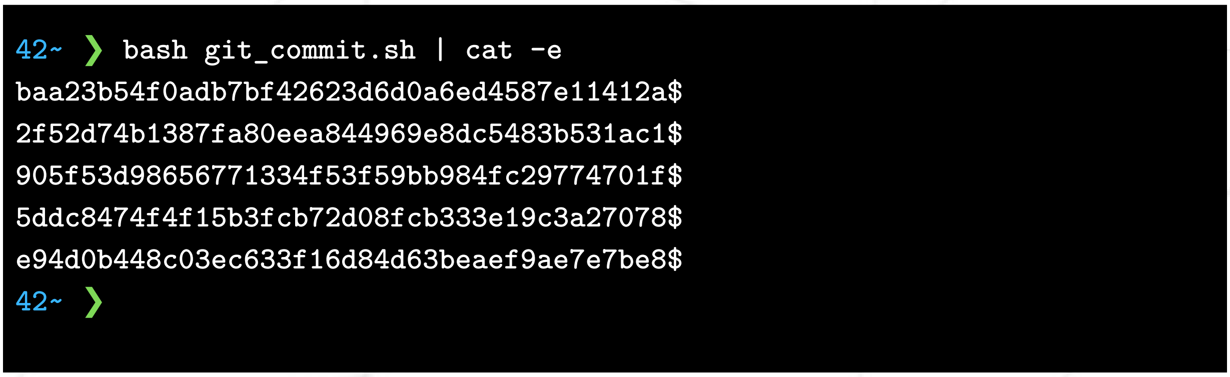
RTFM!

Chapter VIII

Exercice 05: GiT commit

	Exercise 05
GiT commit?	
Turn-in directory: ex05/	
File to turn in: git_commit.sh	
Allowed external functions: None	

- Create a shell script that displays the ids of the last 5 commits of your git repository.



```
42~ ❯ bash git_commit.sh | cat -e
baa23b54f0adb7bf42623d6d0a6ed4587e11412a$
2f52d74b1387fa80eea844969e8dc5483b531ac1$
905f53d98656771334f53f59bb984fc29774701f$
5ddc8474f4f15b3fcb72d08fcb333e19c3a27078$
e94d0b448c03ec633f16d84d63beaef9ae7e7be8$
42~ ❯
```

To test your script, we will use our own environment.

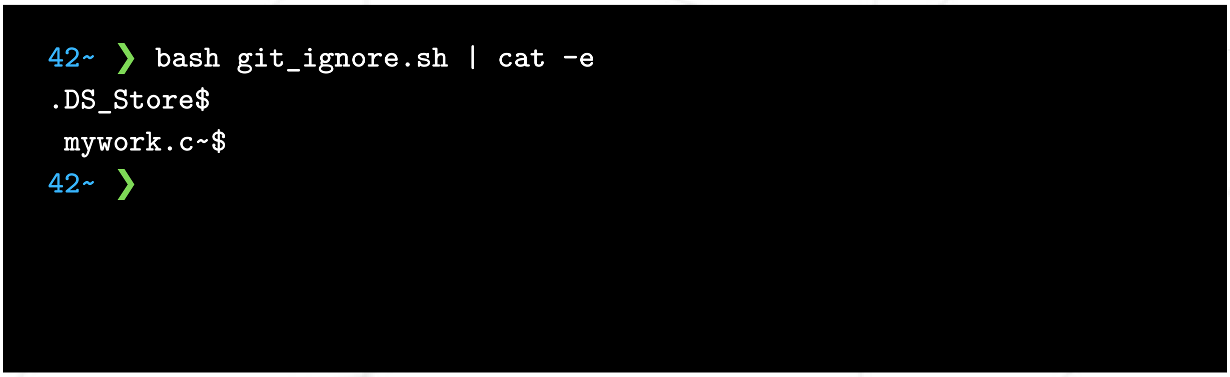
RTFM!

Chapter IX

Exercice 06: gitignore

	Exercise 06
GiT	
Turn-in directory: ex06/	
File to turn in: git_ignore.sh	
Allowed functions: None	

- In this exercice, you will write a short shell script that lists all the existing files ignored by your GiT repository. Example:



```
42~ ❯ bash git_ignore.sh | cat -e
.DS_Store$
 mywork.c~$
42~ ❯
```

To test your script, we will use our own environment.

RTFM!

Chapter IX

Exercice 07: diff

Exercise 07
GiT
Turn-in directory: ex07/
File to turn in: b
Allowed functions: None

- Create a file b, so that :

```
42~ ❯ cat -e a
STARWARS$
Episode IV, A NEW HOPE It is a period of civil war.$
$
Rebel spaceships, striking from a hidden base, have won their first victory against the evil
     Galactic Empire.$
During the battle, Rebel spies managed to steal secret plans to the Empire's ultimate weapon, the
     DEATH STAR,$
an armored space station with enough power to destroy an entire planet.$
$
Pursued by the Empire's sinister agents, Princess Leia races home aboard her starship,custodian of
     the stolen plans that can save her people and restore freedom to the galaxy...$
$
42~ ❯
```

```
42~ ❯ diff a b > sw.diff
```

man patch

Chapter IX

Exercice 08: clean

	Exercise 08
clean	
Turn-in directory: *ex*08/	
File to turn in: clean	
Allowed functions: None	

- In a file called the clean place the command line that will search for all files - in the current directory as well as in its sub-directories - with a name ending by ~, or a name that starts and end by #
- The command line will show and erase all files found.
- Only one command is allowed: no ';' or '&&' or other shenanigans.

man patch

Chapter XII

Exercice 09: Illusions, not tricks, Michael...

	Exercise 09
Illusions, not tricks, Michael...	
Turn-in directory: *ex09/*	
File to turn in: ft_magic	
Allowed functions: None	

- Create a magic file called ft_magic that will be formatted appropriately to detect files of 42 file type, built with a "42" string at the 42nd byte.

man file

Chapter XIII
Submission and peer-evaluation

Turn in your assignment in your Git repository as usual. Only the work inside your repository will be evaluated during the defense. Don't hesitate to double check the names of your files to ensure they are correct.

> ⚠ You need to return only the files requested by the subject of this project.