ANNEX A · PRACTICAL 1.1 · BITS IN THE WILD

NAME _____ DATE _____

Today you open real files and look at their bytes. **You hand in this sheet at the end of the class.** It is marked out of 10: part A 3, part B 3, part C 3, exit ticket 1.

TASK 0 **Yesterday in three questions (no notes)**

1 What is the difference between data and information? _____

2 Which carries more information: "the sun rose today" or "tomorrow's exam is cancelled"? Why? _____

3 The file of the program `ls` starts with `7f 45 4c 46`. What do `45 4c 46` spell in ASCII? _____

Before you start. Download `bits-pack.zip` from Classroom. Open **Files** → **Downloads**, right-click the file → **Extract**. Open the new folder `bits`, right-click on the empty space → **Open in Terminal**. Type `ls`: you must see six files called `mystery-...`.

TASK 1 **Part A · Text is bytes**

The command `od -A x -t x1z` prints the bytes of a file in hex, and on the right the same bytes as text.

A1. Make a file with two letters and look inside.

```
printf 'Hi' > hi.txt
od -A x -t x1z hi.txt
wc -c hi.txt
```

The two bytes are: _____ _____ `wc -c` says the file has _____ bytes.

A2. Now with `echo` instead of `printf`.

```
echo 'Hi' > hi2.txt
od -A x -t x1z hi2.txt
```

There is one more byte. Which one? _____ What do you think it is for? _____

A3. Letters that are not in ASCII. The `|` sends the text straight to `od`.

```
printf 'año' | od -A x -t x1z
printf '€' | od -A x -t x1z
```

`año` has 3 letters and _____ bytes. The `ñ` is the bytes _____ _____. The `€` needs _____ bytes.

A4. Do the same with your own first name. Letters: _____ Bytes: _____

☐ **Done when:** you can say why `año` is 3 letters but not 3 bytes.

TASK 2 **Part B · Magic numbers**

Many file types start with fixed bytes, the **magic number**. Look at the first 8 bytes of each mystery file:

```
head -c 8 mystery-A | od -A x -t x1z
```

File	First bytes	Text	Type
`mystery-A`			
`mystery-B.txt`			
`mystery-C`			
`mystery-D.jpg`			
`mystery-E`			
`mystery-F`			

Starts with	It is a
`ff d8 ff`	JPEG image
`89 50 4e 47`	PNG image
`25 50 44 46`	PDF document
`50 4b 03 04`	ZIP archive
`1f 8b`	gzip file
`7f 45 4c 46`	Linux program
readable letters	plain text

Two files lie about their type. Which ones? _____ and _____ Now let Ubuntu check your table: `file mystery-*`

☐ **Done when:** `file` agrees with the six types you wrote.

TASK 3 **Part C · Shrink it**

C1. Make two files of exactly the same size: one very repetitive, one random.

```
yes hola | head -c 1M > rep.txt
head -c 1M /dev/urandom > random.bin
ls -l rep.txt random.bin
```

Both are _____ bytes. **Bet before you continue:** after compressing, `rep.txt` will be about _____ bytes and `random.bin` about _____ bytes.

C2. Compress both (`-k` keeps the original) and look at the sizes.

```
gzip -k rep.txt random.bin
ls -l
```

`rep.txt.gz`: _____ bytes `random.bin.gz`: _____ bytes

C3. Look at the first lines of each original.

```
od -A x -t x1z rep.txt | head -3
od -A x -t x1z random.bin | head -3
```

What can your eye see in one that it cannot see in the other? _____

C4. Compress the photo.

```
gzip -k mystery-A
ls -l mystery-A mystery-A.gz
```

Before: _____ bytes. After: _____ bytes.

C5. Explain the three results in two sentences. Use at least one of these words: repetition, surprise, predict.

_____

_____

_____

☐ **Done when:** your three "after" sizes are written and your explanation says why they are so different.

EXIT TICKET **Alone. One answer each.**

1. A file is called `holiday.txt`. Its first bytes show `.PNG` in the text column. What is it?

○ a) A text file about a holiday ○ b) A broken file
○ c) A PNG image with a misleading name ○ d) A program

2. In UTF-8, how many bytes does the word `niño` take?

○ a) 3 ○ b) 4
○ c) 5 ○ d) 8

3. Which file will `gzip` shrink the most?

○ a) A chat log with the same words again and again ○ b) A file of random bytes
○ c) A JPEG photo ○ d) A file that is already a `.gz`

4. `echo 'Hi' > f.txt` makes a file of 3 bytes, not 2. Why?

○ a) `echo` writes each letter twice ○ b) The file name is stored inside
○ c) `H` is a capital and needs 2 bytes ○ d) `echo` adds a line feed, `0a`, at the end