

# Spring 2022 courses

## IN PROGRESS os quiz 6

### Settings

- A question has only ONE right answer unless otherwise noted
- This quiz covers content from the last week of the course
- After the first play, the quiz will be opened for unlimited play
- Let me know if you have any comments or corrections

### The order of shell command options is relevant

TRUE

Feedback: E.g. the command `rm -if file` will remove `file` because `-f` invokes the force option, and the command is evaluated in a pipeline from left to right - that is FORCE beats INTERACTIVE because it comes last.

### The `wc` program does not count duplicate words.

FALSE

Feedback: check the Unix man page. `wc` does count duplicate words. To remove duplicate words from a sentence, you need to use multiple commands in a pipe, e.g.:

```
echo "ant spider spider ant zebra ant" | xargs -n1 | sort -u
```

### What do you need to run bash code blocks in GNU Emacs?

*More than one answer is correct.*

TRUE:

- The self-extensible GNU Emacs text editor
- The Org-mode editing environment
- An `.emacs` initialization file
- The bash command line shell

Feedback: To be sure, the `.emacs` file needs to contain special Emacs Lisp commands from the `Org-babel` package.

### Which keyword gives an Org-mode code block a name?

*More than one answer is correct.*

TRUE:

- `#+name:`

- `#+NAME:`

FALSE:

- `#+name`
- `#+TITLE:`

Feedback: Org-mode does not check meta data, just like preprocessor directives are not checked by definition (they are used before the actual processing of a program begins). The keyword is `#+NAME:` but Org-mode is not case sensitive, so `#+name:` works, too. `+TITLE:` is meta data for the Org-mode file title when printed. `#+name` is wrong because the colon is missing.

## Understand containers.

A container is another layer of software between [1] the OS and [2] the applications. Which OS and which applications would you want to connect in this way? Match the correct pairs!

---

Linux	ksh, zsh, csh, sh, bash
Emacs	eshell, shell
Windows	PowerShell, CMD terminal

---

Feedback: containerized software will always run the same, regardless of the infrastructure. Though Emacs is listed as a "text editor", it shares many similarities with an operating system - it can even be used as a windows manager - and can (almost) completely obscure the underlying infrastructure. The Cygwin terminal is another example for a container - while you're in the app, you can access (emulated) Unix tools. Containers are an example of OS virtualization.

## Match the wildcard command and the output

On my Cygwin terminal, the following directories are listed with the command `ls /:`

```
Cygwin-Terminal.ico  Cygwin.ico  cygdrive  etc  lib  sbin  usr
Cygwin.bat          bin         dev       home  proc tmp  var
```

The `echo` command displays whatever comes after it as a line of text. If followed by a wildcard, the shell replaces the wildcard with the list of all files in the current directory (`/` or root) and then passes it on to `echo`, which prints whatever it gets.

Match the command and the output:

---

<code>echo *[s]*</code>	<code>sbin usr</code>
<code>echo *?t*</code>	<code>Cygwin.bat etc</code>
<code>echo *.ico</code>	<code>Cygwin-Terminal.ico Cygwin.ico</code>
<code>echo *[0-9]*</code>	<code>*[0-9]*</code>

---

## Be the system!

Which of the statements below about the file listing are true?

```
71776119061327028 lrwxrwxrwx 1  8 Mar  5 16:34 hello -> hello.sh
25614222880787780 -rw-r--r-- 2 34 Jan 25 08:29 hello.sh
25614222880787780 -rw-r--r-- 2 34 Jan 25 08:29 hello_h
```

*More than one answer is correct.*

TRUE:

- hello is a symbolic link
- The listing shows the file inodes
- hello\_h is a hard link of hello.sh

FALSE:

- hello is a broken link

## What's a broken symbolic link?

TRUE:

- A symbolic link whose origin has been deleted

FALSE:

- A weak hard link
- A hard soft link
- A hard link whose origin has been deleted

## How can you create an alternate command?

When using the command line, you find that you always type `ls -la` - you always want to see all files, including "hidden" dot files, in long format. How can you define a command `ll` that always does exactly that, execute `ls -la`?

TRUE:

- `alias ll = 'ls -la'`

FALSE

- `ll <- ls -la`
- Not possible, `ls` cannot be redefined
- You have to write a bash script to do that

Feedback: you can create an alias for a simple or a compound command (like an entire pipeline) with the `alias` command. An alias can be unmade with the command `unalias`. The `alias` command creates a permanent command alias. That's why the command `unalias` exists. You can find all aliases by typing `alias` at the prompt. You can even override standard commands. E.g. if you always want `ls` to show all files in long format, alias it to `ls -la`. However, for standard utilities, the OS might override your choices at boot time.

## What are the OS communication streams?

*There is more than one correct answer.*

TRUE:

- stderr
- stdout
- stdin

FALSE:

- stdio

Feedback: `stdio.h` is a library of C functions for I/O operations.

Author: Marcus Birkenkrahe

Created: 2022-03-07 Mon 22:29

[Validate](#)