

Computational Astrophysics I: Introduction and basic concepts

Helge Todt

Astrophysics
Institute of Physics and Astronomy
University of Potsdam

SoSe 2025, 17.3.2025



Hands on Linux

Task 1.1 Arguments of commands

- 1 Call the program `cal`. What do you see?
- 2 What is shown, when you type in `cal 9 1752`?
- 3 Find out the weekday of your day of birth.

Task 1.2 Changing directories

- 1 Figure out in which directory you are.
- 2 Change to **your** home directory (not just `/home/`).
- 3 Create there a new directory called as you last name.
- 4 Change into this new directory and figure out again in which directory you are.
- 5 Change back to your home directory and delete the directory you have created before.

emacs *file* &

Task 1.3 emacs

- 1 Start the editor emacs with a **filename as argument**, e.g., your first name, start emacs in the background (&).
- 2 Enter the following text:
"Hello world!"
Double this line (kill & yank).
- 3 Save the file and quit emacs.

```
kate &
```

Task 1.4 kate

- ❶ If not already done so, open a terminal and change to your home directory.
- ❷ Use `cal` to create the calendar for the current month, copy this with your mouse to the clipboard (*copy by selection!*).
- ❸ Start `kate` in the background (&) and paste the calendar view from the clipboard.
- ❹ Delete the *column* with the Mondays, by using the *block selection mode* (menu *edit*) of kate and selecting the the correspondig column.
- ❺ Save the result and quit kate.

Task 1.5 Copying and deleting files

Use `cp` to copy the file `/etc/os-release` into your home directory and check that this has worked (How?). What is the content of this file? Finally, delete this file from your home directory.

Task 1.6 The file attribute `l`

In the, e.g., `/bin` or `/usr/bin/` directory there are some entries with the file attribute `l` (first column).

Inspect them with `ls -l`.

For example have a look on `vi`.

What is shown after the file name?

The following exercises require a personal NIS account in the computer lab, the local user account is not sufficient.

Task 1.7 Hostnames

- 1 On which host are you currently logged in (name)?
- 2 In which DNS domain(!) is your host located?

Hint: Look up the manpage for `hostname`, i.e., by `man hostname`.

Task 1.8 SSH login I

- 1 Login to the host of your neighbor (ask them for the name of their host).
You might be asked by `ssh` whether you want to proceed and then for your password.
- 2 Convince yourself that the login to the other host has worked (Which command?).
- 3 Close the connection with the command `exit`.

Task 1.9

- ① Open a new Xterm by using the command `xterm &`.
- ② Start `top` in the new Xterm.
- ③ How can you change the view in `top` so that the idle processes are not shown?
- ④ Find out in another Xterm the PID (process ID) of the `top` process and terminate it with the `kill` command.

Task 1.10 Setting variables in the shell

- 1 Create a new directory and change to it. Find out the value of PWD via `echo $PWD`, compare the result with that of the command `pwd`.
- 2 Set a variable `color` with the value of your favorite color, e.g., `color=blue`. Check the correct setting via `echo $color`.
- 3 Start a new shell in the current terminal by calling the name of the shell `bash`.
- 4 Print out the values of the variables `PWD` and `color` with `echo`. What do you note?
- 5 Quit the last started shell and repeat the last step 4.
- 6 Repeat the whole task but this time setting the variable `color` with help of the `export` command: `export color=red`.

Task 1.11 Automatically set shell variables

Print out all local and *environment* variables with the commands `set` and `export`. Usually the names of local variables contain only lower case letters and environment variables only capitals.

Task 1.12 The PATH variable

Print out the value of the PATH variable and compare the list with the output of the command `type bash`.

→ The PATH variable contains a list of directories in which to search when you try to execute a command or program in the shell.

Usage of grep: `grep pattern file`

Task 1.13 Searching in files with grep

With the help of `grep` get the line from file `/etc/passwd` printed out that contains the username `SS_08`.

Commands in the shell can be connected with the *pipe* `|` (vertical bar): `command1 | command2` where the output of `command1` will be used as input for `command2`

Task 1.14 grep with pipe

- 1 What is printed out by the combination `ls -l | grep drwx` ?
- 2 Get a list of files from `/bin/` printed, whose name contain the pattern `sh`. What kind of programs are these usually?

Usage of find: `find directory criterion`

Example: `find . -name cache`

→ find file with given name

Task 1.15 find

- 1 What is the meaning of the dot `.` in the example?
- 2 Search with `find` for the file `init.el` in your home directory. Where is it located?
- 3 The command `find` is quite powerfull and has a lot of options, e.g., the criterion `-type` (instead of name) allows to find specific files. How can you get a list of the *directories* in your home directory?

A shell script is a simple ASCII text file (e.g., created with an editor) containing one command per line.

Task 1.16 Simple shell script

- 1 Create a directory `myscripts` and change into it.
- 2 Write with an editor the following text into a file `hello.bash`:

```
#!/bin/bash
echo Hello world!
# this is a comment ...
exit
```
- 3 Make the file executable with `chmod u+x hello.bash`.
- 4 Execute the script: `./hello.bash`

Hint: The editor `emacs` might complain that your script file does not end with a newline. Let the editor do this (yes).

for loops

```
for (( var=begin ; condition ; incr ))  
do  
    ...  
done
```

Example:

```
for (( i=1 ; i<=10 ; i+=1 ))  
do  
    echo $i  
done
```

Task 1.17 Calculate the factorial with a for loop

- 1 Create with an editor a file `forloop.bash` with the shebang `#!/bin/bash` in the first line.
- 2 Then, write a for loop, that calculates the factorial 10!. Before the loop, initialize a variable `let f=1` and in the body of the loop `let f=f*i` where `i` is the loop variable / loop counter. Use `echo $f` to print out the result *after* the loop.
- 3 Make the file executable (`chmod u+x`) and execute it. The correct result must end with two zeros (why?).