ASTR4004/ASTR8004
Astronomical Computing
Assignment 1

Christoph Federrath

due Monday, August 12, 2019

1 The Bourne Again Shell (Bash)

1.1 Bash environment and basics

1. What is Bash? *Write one or two sentences.*

2. Explain briefly what the difference is between `.bashrc` and `.bash_profile`?

3. How do you show all the directories that are listed in the `PATH` environment variable?

4. What is the `PATH` variable good for? *Write max three sentences.*

5. What is the file descriptor FD0 normally used for (by default)?

6. Redirect the standard output and the standard error of
   `Bash> ls * wrong_path_or_file`
   to the same file called `shell.out`.

1.2 Bash scripts basics

1. Write a script using your favourite editor. The script should display the path to your home directory and on the next line show your user name.

2. Modify the script, so it defines a variable called `nonsense` and assign the string 'This is such a nonsense!' to it. Then print the content of the new variable on the screen.

3. Modify the script further, such that the variable `nonsense` is redirected into a file called `nonsense.txt` and print on the screen that this file was just created/overwritten.

4. Go on by writing to the 2nd line of the file (append) a version of `nonsense` that has all white space characters replaced by underscores.

5. Finally replace 'such a' in `nonsense` with 'not' and add ' It's Bash' to the end of the variable and append it to `nonsense.txt`.

6. Write a brief comment for each major line/command in the script, which briefly (in a few words) explains what that line/command does.

7. Write down the Bash command that shows your home directory with all the slashes removed.

1.3 More advanced Bash scripting

1. Write a script using your favourite editor. The script should display the integers from 1 to 20, with one number per line. Use a loop.

2. Now let the script display the same numbers, but now on a single line separated by spaces (result: 1 2 3 ... 20).

3. Modify the script to show the same numbers in one line, but pad with zeros and separate by commas including the last integer (result: 01, 02, ..., 20,).

4. Do the same as in the previous step, but now get rid of the last ', ' by using an if-else-construction (result: 01, 02, ..., 20).

5. Display all the files and directories in the root directory (`/`) and add 'Listing item in root: ' in front of each file or directory for this listing.

2 Useful shell commands

1. Download the data file `EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data` from `http://www.mso.anu.edu.au/~chfeder/teaching/astr_4004_8004/material/mM4_10048_pdfs/EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data`

2. You can use the program `wget` to download the file from the internet.

3. Use `grep` to search and show all lines with 'sigma' in them.

4. Now extract also the next line after 'sigma'.

5. In addition, use a *pipe* to extract the line with the data (the line after 'sigma').

6. Show the help message of Bash. Use `bash --help`.

7. Now only show the lines that explain the usage details of Bash.

8. Show the number of words in the last output. Use another *pipe* to `wc` to achieve this.

9. Add 'Number of words:' in front of the previous output and print to the screen.

10. Use `awk` to extract the entire header (first 11 lines) of the file downloaded above (see point 1). Redirect the output of this to a file called `header.txt`.

11. Use awk to extract the first column of the downloaded file (see point 1 above) without the header (the first 11 lines). Redirect the output of this to a file called column1.txt.

Please send exactly 3 scripts (with the exact same numbering of sections and points as above), named 1.2.sh, 1.3.sh, and 2.sh. Check that your scripts do not produce any errors or warnings when run on *malice* or a similar computer in the MSO network, and add comments to each code line/block of each script. Also produce a writeup for Section 1.1 (preferably in pdf or txt format, again with the same numbering of sections as above). Send these 4 files (3 scripts, 1 write-up file; do *not* include input or output files; these are to be produced according to the instructions above, automatically by your scripts) by the assignment deadline.

Return of assignments is via Turnitin (see respective assignment link at course Wattle page). To upload the files, make a tarball named <YOUR_NAME>.tar.gz that contains the submission files.