Lab 2
Introduction to Cloud Computing
-- Basic Interface and Programming

Prof. Zichen Xu

Lab 2: Bash Programming

Dear Script Programming Newbees

- I think the Docker experiment starts to exhaust your strength and intellect on finding a good solution
- Now you need to do more for your work
- When you start coding, basically you are bashing
- Perquisite: you installed VM with a Linux-kernel system
- Help docs for bashing:
  - https://devhints.io/bash
  - https://linuxconfig.org/bash-scripting-tutorial-for-beginners
  - Our course has established a well-defined relationship with Baidu Inc. and Google Inc. For any of your questions, you can use their service to find out

Goal for Lab 2

- Objectives:
  - To write shell scripts to solve problems
  - To implement some standard Linux-kernel utilities such as ls,cp,etc using system calls.
  - Understanding the idea of multi-programming (or multiplexing) and threading
  - Learn to read an English poem
  - Learn to read instructions, carefully
  - Write a decent report

Use Bash for Shell scripts

1. Write a Shell script that accepts a filename, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

2. Write a Shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.

3. Write a Shell script that displays list of all the files in the current directory to which the user has read, Write and execute permissions.

4. Write a Shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.

5. Write a Shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.

6. Write a Shell script to list all of the directory files in a directory

7. Write a Shell script to find factorial of a given integer.

File to use in your Test
Linux Poem: The Reentrant Kernel

By Morgan Phillips

Save the poem as your text file

A reentrant function,
if interrupted,
will return a result,
which is not perturbed.

```
int global_int;int is_not_reentrant(int x) { int x = x; return global_int + x; },
```

depends on a global variable,
which may change during execution.

```
int global_int;int is_reentrant(int x) { int saved = global_int; return saved + x; },
```

mitigates external dependency,
it is reentrant, though not thread safe.

Now to Show it Off

- Instead of running basic bash coding solely, you need to *exec* it
- You need to write a control script (i.e., script for script) that execute all your previous bash code one by one in a certain order
- You need to append this piece of code to your report, as well as screen shot of running this code

- HINTS: bash allows multiplexing in executing certain codes, the specific cmd to run it is hidden in this page.

Lab1: Submission Requirement

- Use the provided basic Markdown template for your lab report online
- Fill the report with the Markdown template, including all your test procedure
- Compile it and submit it to ncuhomework@ncu.edu.cn
- Subject shall be Course Z6110X0035 CC Lab 2 #name #ID
- The tentative deadline for this lab is postponed to Apr. 14th, 11:59PM (Hard Deadline)