**CIS 241 Winter 2018 - Homework 2 (4% of the final grade)**

**Due date: 4:00PM on April 17, 2018**

Maximum Points: 100

(Please finish your homework individually, and submit it **in hard copy** in April 17’s class)

1. (12 points) Please explain the file access permissions after the following commands are executed respectively.
   1. chmod u+x example\_a

After the execution of the command, the user (u) will be able to execute (x) the file example\_a.

* 1. chmod o-rw example\_b

After the execution of the command, users not in the group (o) won’t be able to read (r) or write (w) on the file example\_b

* 1. chmod 777 example\_c

7 stands for a combination of read (4), write (2) and execute (1). The first number is applied to the user, the second to the group, the third to others. So, after the execution of the command, the user, the group and others will be able to read, write and execute the file example\_c.

* 1. chmod 640 example\_d

0 means no permission, 4 read permission. 6 stands for a combination of read(4) and write(2). After the execution of the command, the user will be able to read and write, the group to read, and others can do nothing, to the file example\_d.

1. (21 points) Please briefly explain the meanings of the following Linux commands.
   1. pwd

“print working directory”, the name says it all; it displays on the command line the absolute path from root to the current directory you are in.

* 1. grep

“global regular expression print”, the command search for the given file and print all the matching words or lines of text where all the regular expressions given match.

* 1. ps

The “process status” command displays a list with information about the currently running processes (or “tasks”). One such information is the PID, the process identification number.

* 1. Find

The find command search for a specific file inside the file hierarchy provided or, if one is not provided, starts from the root.

* 1. p1 & ls #assuming p1 is an executable Linux command

& makes the command that precedes it to run in background. This full command says to run “p1” in background and to execute “ls” in the foreground.

* 1. p2 & ls & #assuming p2 is an executable Linux command

This command makes “p2” and “ls” run in the background. By some testing I have done, ls will finish its process instantly, so it won’t stay in the “ps” for much.

* 1. fg

The command moves a process to the foreground. This could be used to move a background process to the foreground, or to start a frozen process back again

1. (15 points) Please explain which kinds of files are shown for the following “ls” commands respectively.
   1. ls \*.txt

Display the information about all files ending with .txt

* 1. ls a\*.c

Display the information about all files starting with ‘a’ and ending with “.c”

* 1. ls ab?.\*

Display the information about all files starting with ‘ab’, followed by any one letter (could be none), followed by a dot ‘.’, and ending with any string. An example could be “abc.wow” or “ab.f”

* 1. ls [^a-g]\*

Display information about all the files except the ones starting with a letter between ‘a’ and ‘g’.

* 1. ls [abc]\*

Display information about all the files starting with ‘a’, ‘b’ or ‘c’.

1. (15 points) Please describe what the matched strings are for each regular expression below. You can provide some examples to help you explain your idea.
2. a\*b

Any expression that contains a string defined by starting with ‘a’ and ending with ‘b’. the rest of the string between ‘a’ and ‘b’ could be empty.

1. a.\*b

Any expression that contains a string defined by starting with ‘a’ and ending with ‘b’, and must have at least one character between ‘a’ and ‘b’.

1. [^a]bc

Any expression that contains a string defined by starting with any letter except ‘a’, and continues with ‘bc’.

1. ^abc

Any expression that starts with ‘a’ and continues with ‘bc’ and ends with any other string (could be empty).

1. a+b

Any expression that contains a string starting with one or more ‘a’s and ends with b.s

1. (10 points) Based on the given text below

*Is this a car? No, this is a bike.*

Please describe what the matched strings are for the two regular expressions below respectively, and briefly explain why.

1. th.\*s

This matches the longest string possible starting with ‘th’ and ending with ‘s’, in this case:

“this a car? No, this is”

1. th\*is

This matches the shortest string possible starting with ‘th’ and ending with ‘is’, in this case:

“this” from “Is this a car?”

1. (15 points) Links. Please answer the following questions.
2. What is the command to create a hard link?

To create a hard link the command is “ln filename linkname”

1. What is the command to create a symbolic/soft link?

To crease a soft link the command is “ln -s filename linkname”

1. What is the difference between file creation by duplication (using “cp” command) and file creation by hard link? You can explain your answer with an example.

“cp” copies the contents of a file to a new file, effectively creating two different files with the same content. A hard link, instead, references an already existing file, without making a copy of its content.

1. (12 points) What are the commands for the following tasks respectively?
2. Replace all occurrences of the string “cis” with “cs” in the file “syllabus”

sed -i ‘s/cis/cs/**g**’ syllabus

1. Add “GVSU” at the beginning of each line in the file “courses”

sed -i ‘s/^/GVSU /g’ courses

note: I put a space after each GVSU.

1. Remove all strings between < and > (including < and >) on each line in the file “index.html”

sed 's/<[^>]\*>//g' index.html

this will only make a temporary output with all strings between < and > removed.

Sed -i 's/<[^>]\*>//g' index.html

this will actually modify the file.