

PRINT Name: \_\_\_\_\_

LAB Section:

--	--	--

One-Answer Multiple Choice 44 Questions

Weight 15%

- ☞ Read **all** the words of these instructions and **both** sides (back and front) of all pages.
- ☞ Manage your time. Answer questions you know, first. One Answer per question.
- ☞ **PRINT** your Name and Lab on this Question Sheet. You may write or draw on this sheet.
- ☞ Use your full, unabbreviated name on the mark-sense form. Do not abbreviate your name.
- ☞ Enter your NAME, Student Number, and Answers. Fill in the bubbles with pencil, no pen.
- ☞ Leave the last question about reading all these test instructions blank. No answer. **Neniu**

1. [53/153] What does this command print: `awk '{print $NF}'`
  - a. the shell variable `$NF`
  - b. field number `N` followed by field number `F`
  - c. the number of fields
  - d. the first field
  - e. the last field
2. [54/153] Create a symbolic link under `/bin` named `sh` that has target `dash`
  - a. `ln -s /bin/sh 'dash'`
  - b. `ln -s /bin/sh '/bin/dash'`
  - c. `ln -s '/bin/dash' /bin/sh`
  - d. `ln -s 'dash' '/bin/sh'`
  - e. `ln -s /bin/dash sh`
3. [56/153] If my current directory is `/usr/bin`, which of these pathnames is equivalent to the file name `/bin/bash`?
  - a. `./bash`
  - b. `../bin/bash`
  - c. `../../bin/./bash`
  - d. `./bin/bash`
  - e. `/usr/bin/./bash`
4. [57/153] In an empty directory, how many words are in file `a` after this:
 

```
mkdir .1 .2 .3 ; ls >a
```

  - a. 1
  - b. 3
  - c. 4
  - d. 0
  - e. 2
5. [60/153] What command **always** recursively finds all things with names that begin with `foo` no matter what is in the current directory?
  - a. `fgrep -name 'foo*'`
  - b. `find foo* -name`
  - c. `find -name 'foo*'`
  - d. `find -name foo*`
  - e. `fgrep foo* -print`

6. [63/153] File `a` contains 2 lines. File `b` contains 3 lines. How many lines are output on your screen by this: `cat b | echo a ; echo a`
  - a. 1
  - b. 5
  - c. 2
  - d. 7
  - e. 4
7. [64/153] File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this command line:
 

```
ln a c ; ln c d ; cat a b >d d
```

  - a. 4
  - b. 0
  - c. 5
  - d. 3
  - e. 2
8. [64/152] What is the output on your screen of this unquoted command line:
 

```
mkdir a ; touch b a/b1 a/b b2 ; find a -name b?
```

  - a. `a/b1 a/b`
  - b. `a/b1`
  - c. `b1 b`
  - d. `a/b2`
  - e. no output
9. [66/153] In an empty directory, how many words are in file `c` after this:
 

```
echo a >z ; echo y >x ; mv x y >x ; ls >c
```

  - a. 0
  - b. 1
  - c. 2
  - d. 3
  - e. 4
10. [67/153] In an empty directory, how many words output on your screen after this: `echo It's "redirected" >a isn't it? >b ; ls >c`
  - a. 0
  - b. 3
  - c. 1
  - d. 2
  - e. 4
11. [68/153] What is the output of this in an empty directory:
 

```
mkdir abc cba ab a ; echo [abc]
```

  - a. `abc cba`
  - b. `a ab abc`
  - c. `a`
  - d. an error message from `echo` saying `[abc]` does not exist
  - e. `abc`
12. [70/152] Dereference the following symlink `bar` into its equivalent absolute path: `ln -s ../../1/../../xyz /etc/1/2/bar`
  - a. `/etc/xyz`
  - b. `/etc/1/xyz`
  - c. `/etc/2/bar`
  - d. `/etc/2/xyz`
  - e. `/etc/1/2/bar`
13. [71/153] In an empty directory, how many words are in file `a` after this:
 

```
touch 1 2 1 2 ; ls >a
```

  - a. 2
  - b. 0
  - c. 1
  - d. 3
  - e. 4

14. [73/153] File **a** occupies one disk block. How many disk blocks are in use after this sequence of commands:  
`ln a b ; ln b c ; cp c d ; mv c e ; rm a d`  
 a. 0            b. 4            c. 1            d. 3            e. 2
15. [75/152] File **a** occupies one disk block. How many disk blocks are in use after this sequence of commands:  
`cp a c ; ln c b ; cp b d ; ln b e ; cp d a ; rm a b c`  
 a. 4            b. 1            c. 2            d. 0            e. 3
16. [75/153] In an empty directory, what is in file **c** after this:  
`echo a >c ; ls b >c`  
 a. nothing (empty file)            b. c  
 c. `ls: cannot access b`            d. a  
 e. b
17. [76/153] If `/bin/cmd` is a program that outputs **one** and `/usr/bin/cmd` is a program that outputs **two** what would be the output on your screen of this three command sequence:  
`PATH=/bin:/usr/bin:/etc ; cd /usr/bin ; ./cmd`  
 a. **one**  
 b. `bash: ./cmd: no such file or directory`  
 c. **two**  
 d. **two** followed by **one**  
 e. **one** followed by **two**
18. [77/153] Given this long listing:  
`drwxr-xr-x 248 bin bin 496 Jan 1 00:00 dir`  
 How many subdirectories lie immediately under **dir**?  
 a. 494            b. not enough information  
 c. 246            d. 248  
 e. 496
19. [77/153] Which command line outputs `/bin/date`?  
 a. `touch /bin/date`            b. `cd /bin ; ls date`  
 c. `cd /bin ; echo date`            d. `which date`  
 e. `cat /bin/date`
20. [79/153] File **a** occupies one disk block. How many disk blocks are in use after this sequence of commands:  
`cp a c ; ln c b ; cp b d ; cp a b`  
 a. 0            b. 3            c. 2            d. 1            e. 4

21. [80/152] If `/bin/cmd` is a program that outputs **one** and `/usr/bin/cmd` is a program that outputs **two** what would be the output on your screen of this three command sequence:  
`PATH=/bin/cmd:/usr/bin/cmd:/usr ; cd /bin ; cmd`  
 a. **one** followed by **two**  
 b. **two**  
 c. **one**  
 d. **two** followed by **one**  
 e. `bash: cmd: command not found`
22. [82/152] In `/etc` using `ls -l` shows a symbolic link `lnk -> dir/foo` then dereference the absolute path of `lnk` with no symbolic links:  
 a. `/lnk/dir/foo`            b. `/etc/dir/foo`  
 c. `/etc/dir/foo/lnk`            d. `/dir/foo`  
 e. `/etc/lnk/dir/foo`
23. [82/153] Which line allows the shell to find the `assignment07check` command?  
 a. `$PATH=$PATH:~idallen/cst8207/16f/assignment07`  
 b. `PATH=$PATH:~idallen/cst8207/16f/assignment07`  
 c. `PATH=PATH:~idallen/cst8207/16f/assignment07`  
 d. `$PATH=PATH:~idallen/cst8207/16f/assignment07`  
 e. `PATH=which assignment07check`
24. [82/153] Which pathname always leads to the same file named: `/bin/rm`  
 a. `././bin/rm`            b. `.././bin/./rm`  
 c. `/bin/../../../../rm`            d. `/bin/rm/../../../../`  
 e. `/bin/rm/./.`
25. [83/153] Dereference the following symlink `bar` into its equivalent absolute path: `ln -s /bin/sh /tmp/1/2/bar`  
 a. `/bin/1/2/bar`            b. `/bin/1/sh`  
 c. `/tmp/sh`            d. `/bin/sh`  
 e. `/bin/2/bar`
26. [83/153] File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this command line:  
`sort b >c a ; cat a >>b ; sort c a >c b`  
 a. 5            b. 12            c. 7            d. 0            e. 8

27. [86/152] File **a** occupies one disk block. How many disk blocks are in use after this sequence of commands:
- ```
cp a b ; ln b c ; ln a e ; cp e c ; rm a b
```
- a. 1            b. 3            c. 2            d. 0            e. 4
28. [86/153] What is true about this output from `ls -ild foo bar`?
- ```
111 -r--r--r-- 3 root root 2 Jan 1 00:00 foo
111 -r--r--r-- 3 root root 2 Jan 1 00:00 bar
```
- a. **foo** and **bar** are two of two names for this file  
 b. this output is not possible  
 c. **foo** and **bar** each have three names (six names total)  
 d. **foo** and **bar** are two of three names for this file  
 e. **foo** and **bar** are names for different files
29. [88/153] How many arguments are passed to the command by the shell:
- ```
touch It's not hard, it's logical.
```
- a. 3            b. 2            c. 4            d. 1            e. 5
30. [89/151] If `/bin/cmd` is a program that outputs **one** and `/usr/bin/cmd` is a program that outputs **two** what would be the output on your screen of this three command sequence:
- ```
PATH=/lib:/etc:/usr/bin:/bin ; cd /bin ; cmd
```
- a. **bash: cmd: command not found**  
 b. **one**  
 c. **one** followed by **two**  
 d. **two**  
 e. **two** followed by **one**
31. [92/153] File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this command line: `cat b | cp a b`
- a. no output            b. 2            c. 6  
 d. 3            e. 5
32. [94/153] If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
- ```
111 -rw-r--r-- 2 me me 1 Jan 1 1:00 a
111 -rw-r--r-- 2 me me 1 Jan 1 1:00 b
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 c
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 d
```
- a. 2            b. 0            c. 1            d. 3            e. 4

33. [96/153] If a shell GLOB pattern fails to match anything, the shell:
- a. gives an error message and does not execute  
 b. passes the pattern unchanged to the command  
 c. gives a warning message but continues  
 d. returns the closest match to the pattern  
 e. removes the pattern and passes nothing
34. [100/153] Which file is a DOS/Windows file?
- a. **ASCII text**  
 b. **ASCII text, with CR line terminators**  
 c. **ASCII text, with CRLF line terminators**  
 d. **ASCII text, with no line terminators**  
 e. **ASCII text, with LF line terminators**
35. [104/153] What is the output of this in an empty directory:
- ```
touch .abc .dog ; echo .??*
```
- a. **.. .abc .dog**  
 b. **.abc**  
 c. **.??\***  
 d. an error message from **echo** saying **.??\*** does not exist  
 e. **.abc .dog**
36. [106/153] If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
- ```
111 -rw-r--r-- 3 me me 1 Jan 1 1:00 a
111 -rw-r--r-- 3 me me 1 Jan 1 1:00 b
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 c
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 d
```
- a. 0            b. 4            c. 2            d. 3            e. 1
37. [106/153] Which command counts lines containing the string **refused** in only the month **October 2015** in the **denyhosts** log file?
- a. **fgrep refused denyhosts ; fgrep -c 2015-10**  
 b. **fgrep -c 2015-10 denyhosts | fgrep refused**  
 c. **fgrep refused denyhosts | fgrep -c October 2015**  
 d. **fgrep '2015-10 refused' denyhosts**  
 e. **fgrep 'refused' denyhosts | fgrep -c '2015-10'**

38. [107/153] Which command shows names under directory **oldnotes** containing **RTFM** anywhere in the name?
- a. `ls oldnotes *RTFM*`                      b. `ls oldnotes*RTFM*`  
 c. `ls oldnotes RTFM *`                      d. `ls oldnotes/*RTFM*`  
 e. `ls oldnotes/RTFM*`
39. [108/153] If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
- ```
111 -rw-r--r-- 1 me me 1 Jan 1 1:00 a
222 -rw-r--r-- 1 me me 1 Jan 1 1:00 b
333 -rw-r--r-- 1 me me 1 Jan 1 1:00 c
444 -rw-r--r-- 2 me me 1 Jan 1 1:00 d
```
- a. 2                      b. 3                      c. 4                      d. 1                      e. 0
40. [108/152] What is the link count of directory **d** after this set of successful commands? `mkdir d ; cd d ; touch a ; ln a b ; mkdir c`
- a. 3                      b. 4                      c. 5                      d. 6                      e. 2
41. [114/153] If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
- ```
111 -rw-r--r-- 1 me me 1 Jan 1 1:00 a
222 -rw-r--r-- 1 me me 1 Jan 1 1:00 b
444 -rw-r--r-- 2 me me 1 Jan 1 1:00 c
444 -rw-r--r-- 2 me me 1 Jan 1 1:00 d
```
- a. 0                      b. 3                      c. 2                      d. 1                      e. 4
42. [116/153] If I am in directory **/tmp** and **d** is an empty sub-directory, what is true after this command line:
- ```
mkdir d/x ; touch d/foo ; cp d/foo d/../x
```
- a. the command fails because the name **d/foo** does not exist  
 b. the directory **d** now contains only a file named **x**  
 c. there is a second copy of the file **foo** in file **/tmp/x**  
 d. the directory **d** is now empty  
 e. there is a second copy of the file **foo** in directory **d**
43. [120/152] If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
- ```
111 -rw-r--r-- 1 me me 1 Jan 1 1:00 a
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 b
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 c
222 -rw-r--r-- 3 me me 1 Jan 1 1:00 d
```
- a. 4                      b. 3                      c. 0                      d. 2                      e. 1

44. [122/153] A "dangling symlink" is a symlink to:
- a. a directory                                      b. the current directory  
 c. a non-existent target                      d. a parent directory  
 e. a special device file