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. *[10/79]* In an empty directory, what is the output of this unquoted command
   sequence:
       mkdir a ; touch b? a/b1 a/b22 ; find a –name b*

   a. **a/b1**   *b.* **b1 a/b1 a/b22**
   c. no output   *d.* **b1 b22**
   e. **a/b1 a/b22**

2. *[29/79]* In an empty directory, how many words are in file **c** after this
   command line: **touch a >b ; ls >c**

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

3. *[34/77]* How many arguments and options are there in the command line:
       ls –ls ls

   a. Three arguments, one of which contains two options.
   b. Two arguments: one option argument and one command name argument.
   c. Two arguments, one of which contains two options.
   d. Two arguments: one option argument and two command name arguments.
   e. Three arguments, two of which are options.

4. *[36/79]* Which command line below shows only lines **10–15** of file **foo**?

   a. **tail –n 15 foo | head –n 6**
   b. **head –n 15 foo | tail –n 6**
   c. **head –n 15 foo | tail –n 5**
   d. **head –n 6 foo | tail –n 15**
   e. **tail –n 15 foo | head –n 5**

5. *[40/79]* In an empty directory, what is the output on your screen after this
   command line: **touch .onk >.brk ; echo .?***

   a. **.?***
   b. an error message from **echo** saying **.?*** does not exist
   c. **.   ..   .brk   .onk**
   d. **.brk .onk**
   e. **..   .brk   .onk**

6. *[43/79]* In an empty directory, what is in file **lines** after this command
   line: **ls nosuchfile | wc –l >lines**

   a. **nosuchfile**   *b.* **1**
   c. nothing (empty file)   *d.* **0**
   e. **1 nosuchfile**

7. *[45/79]* Create a symbolic link under **/bar** named **lib** that has target **xxx**

   a. **ln –s /bar/xxx /bar/lib**
   b. **ln –s /bar/lib xxx**
   c. **ln –s lib /bar/xxx**
   d. **ln –s /bar/xxx lib**
   e. **ln –s xxx /bar/lib**

8. *[45/79]* File **a** occupies one file disk block. How many file disk blocks are
   in use after this sequence of commands:
       mkdir b c ; mv a b/d ; ln b/* c/ ; rm –r b
   a. **2**   *b.* **3**   *c.* **0**   *d.* **1**   *e.* **4**

9. *[46/78]* If I am in directory **/home/onk** and **mt** is an empty sub-directory,
   what is true after this command line:
       touch foo moo ; mv ./mt/../moo ./onk/brk

   a. the directory **mt/..** now contains a file named **brk**
   b. there is a second copy of the file **moo** in the file named **brk**
   c. the command fails because path **./onk/brk** does not exist
   d. the command fails because path **./mt/../moo** does not exist
   e. the directory **onk** now contains a file named **brk**

10. *[47/78]* Dereference the following symlink **xxx** into its equivalent absolute
    path: **ln –s ../../a/./b/./yyy /lib/a/b/xxx**

    a. **/lib/a/b/xxx**   *b.* **/lib/a/yyy**
    c. **/lib/a/xxx**   *d.* **/lib/xxx**
    e. **/lib/a/b/yyy**

11. *[47/79]* File **a** contains 2 lines, and file **b** contains 3 lines, then how many lines are in file **c** after this command line: **ln b c ; cat b b a >c**

    a. **8**   *b.* **4**   *c.* **2**   *d.* **3**   *e.* **0**

12. *[47/79]* If a shell GLOB pattern fails to match anything, the shell:

    a. returns the closest match to the pattern
    b. gives an error message and does not execute
    c. gives a warning message but continues
    d. passes the pattern unchanged to the command
    e. removes the pattern and passes nothing

13. *[47/79]* If **moo** is a sub-directory that contains only the file **brk**, what happens after these commands:
    touch onk ; mv ./moo/onk ./moo/brk

    a. there is only the file named **onk** in the **moo** directory now
    b. a new file named **onk** is created in **moo**
    c. there is a second copy of the file **onk** in the file named **brk**
    d. the command fails because the name **onk** does not exist
    e. the command fails because **brk** is not a directory

14. *[48/79]* File **a** occupies one file disk block. How many file disk blocks are in use after this sequence of commands:
    ln a b ; ln b c ; cp c d ; ln c e ; rm a b c

    a. **4**   *b.* **1**   *c.* **2**   *d.* **3**   *e.* **5**

15. *[48/79]* If **/bin/prog** is a program that outputs **one** and **/usr/bin/prog** is a program that outputs **two** what would be the output on your screen of this two command sequence:
    PATH=/usr/bin/prog:/bin/prog ; prog

    a. **one** followed by **two**
    b. **one**
    c. **bash: prog: command not found**
    d. **two** followed by **one**
    e. **two**

16. *[48/79]* What is true about this output from **ls -il foo bar**?
    99 -rwxrwxrwx 2 bin bin 3 Jul 31 12:33 foo
    99 -r--r--r-- 2 bin bin 3 Jul 31 12:33 bar

    a. this output is not possible
    b. **foo** and **bar** each have three names (six names total)
    c. **foo** and **bar** are names for the same file
    d. **foo** and **bar** are two of three names for the same file
    e. **foo** and **bar** are names for different files

17. *[49/79]* File **a** contains 2 lines, and file **b** contains 3 lines, then how many lines are in file **c** after this command line:
    cat a b >c ; cat a >>b ; cat c b >c a

    a. **8**   *b.* **5**   *c.* **7**   *d.* **10**   *e.* **12**

18. *[50/78]* File **a** occupies one file disk block. How many file disk blocks are in use after this sequence of commands:
    cp a b ; ln b c ; cp c d ; ln a e ; cp a d ; rm c d

    a. **4**   *b.* **2**   *c.* **3**   *d.* **1**   *e.* **5**

19. *[50/79]* If my current directory is **/onk**, which of these is the shortest pathname equivalent to **/onk/a/b**?

    a. **./onk/a/b**   *b.* **../a/b**   *c.* **onk/a/b**
    d. **/a/b**   *e.* **../onk/a/b**

20. *[50/79]* Which line allows the shell to find the **assignment07check** command?

    a. **PATH=assignment07check:$PATH**
    b. **PATH=which assignment07check**
    c. **PATH=$PATH:~idallen/cst8207/19w/assignment07**
    d. **PATH=whereis assignment07check**
    e. **PATH=$PATH:assignment07check**

21. *[51/78]* File **a** occupies one file disk block. How many file disk blocks are in use after this sequence of commands:
    cp a b ; ln b c ; cp c d ; cp a c ; rm c d

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

22. *[51/77]* If I am in directory **/home/onk** and **mt** is an empty sub-directory, what is true after this command line:
    touch mt/./onk onk ; rm ./mt/onk ../onk/onk

    a. the directory **mt** is still empty
    b. the command fails because the path **./mt/onk** does not exist
    c. the directory **onk** now contains a file named **onk**
    d. the command fails because the path **../onk/onk** does not exist
    e. the directory **mt** now contains a file named **onk**

23. *[51/79]* If **/bin/prog** is a program that outputs **one** and
/usr/bin/prog is a program that outputs **two** what would be the output
on your screen of this two command sequence:
    PATH=/home:/usr:/bin:/etc:/usr/bin ; prog

    a. **two** followed by **one**
    b. **one**
    c. **two**
    d. **bash: prog: command not found**
    e. **one** followed by **two**

24. *[52/79]* What is the link count of directory **x** after this set of successful
commands? **mkdir x ; mkdir –p x/y/a x/y/b x/y/c x/y/d**

    a. **4**   *b.* **2**   *c.* **6**   *d.* **3**   *e.* **5**

25. *[52/79]* Which command line usually outputs the pathname **/bin/sh**?

    a. **cd /bin ; file sh**   *b.* **cd /bin ; ls sh**
    c. **which sh**   *d.* **cat /bin/sh**
    e. **cd /bin ; echo sh**

26. *[53/78]* In **/usr/sbin** using **ls –l** shows a symbolic link **sh ->**
/bin/bash then dereference the shortest absolute path of **sh** with no
symbolic links:

    a. **/sh/bin/bash**   *b.* **/usr/sbin/sh/bin/bash**
    c. **/bin/bash**   *d.* **/usr/sbin/bin/bash/sh**
    e. **/usr/sbin/bin/bash**

27. *[54/77]* In **/usr/bin** using **ls –l** shows a symbolic link **bar ->**
../sbin/foo then dereference the shortest absolute path of **bar** with no
symbolic links:

    a. **/usr/sbin/foo**   *b.* **/usr/bin/sbin/foo**
    c. **/usr/bin/sbin/bar/foo**   *d.* **/usr/sbin/bar/foo**
    e. **/usr/bin/bar/sbin/foo**

28. *[55/79]* Given this long listing:
    drwxr-xr-x 456 me me 123 Dec 4 9:12 dir
How many subdirectories lie immediately under **dir**?

    a. **456**
    b. **454**
    c. **123**
    d. there is not enough information shown to answer the question
    e. **121**

29. *[55/78]* If I am in directory **/home/onk** and **mt** is an empty sub-directory,
what is true after this command line:
    touch onk moo ; mkdir brk ; mv moo brk/mt

    a. the directory **mt** is still empty
    b. the directory **brk** now contains a file named **moo**
    c. the command fails because **brk/mt** is not a directory
    d. the directory **mt** now contains a file named **moo**
    e. the directory **mt** now contains a directory named **brk**

30. *[56/79]* File **a** occupies one file disk block. How many file disk blocks are
in use after this sequence of commands:
    cp a b ; ln b c ; ln c d ; ln a e ; rm a b c

    a. **2**   *b.* **3**   *c.* **5**   *d.* **4**   *e.* **1**

31. *[56/78]* Give the minimum number of directories in this pathname:
    /a/b/c/d/e

    a. **3**   *b.* **2**   *c.* **5**   *d.* **1**   *e.* **4**

32. *[56/78]* What is the link count of directory **x** after this set of successful
commands?

mkdir x ; mkdir y ; cd x ; touch a ; mkdir b ; ln a b/c

    a. **4**   *b.* **3**   *c.* **5**   *d.* **2**   *e.* **6**

33. *[58/79]* How many arguments are passed to the command by the shell:
    echo " 1 '2 3' 4 "5 6 ' 7 "8 '9 10 >out

    a. **2**   *b.* **5**   *c.* **6**   *d.* **3**   *e.* **4**

34. *[58/79]* How many files are touched or created?
    touch " 1 '2 3 4' "56 ' "7" 8 '

    a. **1**   *b.* **5**   *c.* **2**   *d.* **3**   *e.* **4**

35. *[58/79]* If files occupy one disk block, how many file 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-- 2 me me 1 Jan 1 1:00 b
    333 -rw-r--r-- 2 me me 1 Jan 1 1:00 c
    333 -rw-r--r-- 2 me me 1 Jan 1 1:00 d

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

36. *[58/78]* If files occupy one disk block, how many file 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
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. **2**   *b.* **1**   *c.* **0**   *d.* **4**   *e.* **3**

37. *[61/79]* Which shell GLOB pattern matches only the case-insensitive WAREZ files from Assignment 5?
a. ***abcd0001*[w,W][a,A][r,R][e,E][z,Z]***
b. ***abcd0001*[wW][aA][rR][eE][zZ]***
c. ***abcd0001*[warez][WAREZ][Warez][WareZ]***
d. ***abcd0001*[w-W][a-A][r-R][e-E][z-Z]***
e. ***abcd0001*[wWaArReEzZ]***

38. *[62/79]* If files occupy one disk block, how many file 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.* **4**   *c.* **3**   *d.* **1**   *e.* **0**

39. *[62/79]* What is true about this output from **ls -il one two**
```
11 -rwxrwxrwx 2 bin bin 3 Jul 31 12:33 one
99 -rwxrwxrwx 2 bin bin 3 Jul 31 12:33 two
```
a. **one** and **two** are two of three names for the same file
b. **one** and **two** are names for the same file
c. **one** and **two** are names for different files
d. **one** and **two** each have three names (six names total)
e. this output is not possible

40. *[64/79]* How do you execute the program **prog** in the current directory?
a. **prog**   *b.* **./prog**   *c.* **~/prog**
d. **$HOME/prog**   *e.* **/prog**

41. *[65/78]* File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this command line: **cat a | tail b**
a. **3**   *b.* **4**   *c.* **2**
d. **3** followed by **2**   *e.* **2** followed by **3**

42. *[66/79]* If files occupy one disk block, how many file 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-- 2 me me 1 Jan 1 1:00 c
444 -rw-r--r-- 2 me me 1 Jan 1 1:00 d
```
a. **3**   *b.* **0**   *c.* **4**   *d.* **1**   *e.* **2**

43. *[68/79]* What is true about this output from **ls -il one two**
```
11 -rwxrwxrwx 2 bin bin 3 Jul 31 12:33 one
11 -rwxrwxrwx 2 bin bin 3 Jul 31 12:33 two
```
a. this output is not possible
b. **one** and **two** are names for the same file
c. **one** and **two** are names for different files
d. **one** and **two** are two of three names for the same file
e. **one** and **two** each have three names (six names total)

44. *[69/79]* If files occupy one disk block, how many file 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. **3**   *b.* **2**   *c.* **4**   *d.* **1**   *e.* **0**