

PRINT Name: _____

Test Version: 774 One-Answer Multiple Choice 134 Questions – 40 of 40%

- ☞ Read **all** the words of these instructions and **both** sides (back and front) of all pages.
- ☞ Put your Name on this Question Sheet. You may write or draw on this Question Sheet.
- ☞ Use your full, unabbreviated name on the mark-sense form. Do not abbreviate your name.
- ☞ Put the three-digit **Test Version** above into both **NO. OF QUESTIONS** and **NO. OF STUDENTS**
- ☞ Fill in the bubbles with pencil only, no pen. Enter your NAME, Test Version, and answers.
- ☞ Manage your time. Answer questions you know, first. One Answer per question.

1. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-wx---rw- 2 bob ted 60 Jan 1 1:00 foo`
`---rwxrwx 1 bob bg2 0 Jan 1 1:00 foo/bar`
 - a. **pat** can rename the file
 - b. **pat** can access and write on the file
 - c. **bob** can access and write on the file
 - d. **bob** can list names in the directory
 - e. **bob** can create a new file in the directory
2. What permissions are given to **newdir** after this command line:
`umask 156 ; mkdir newdir`
 - a. `r-x--x---`
 - b. `rw--w----`
 - c. `rw--w---x`
 - d. `r-x-w-rw-`
 - e. `--xr-xrw-`
3. Given this successful command line (note the dot argument):
`cd /tmp ; mkdir dir ; cd dir ; chmod u-x .`
 Which next command will execute without any "permission denied" errors?
 - a. `ls .`
 - b. `ls /tmp/dir/..`
 - c. `ls /tmp/dir`
 - d. `ls /tmp/dir/.`
 - e. `ls ..`
4. Given the following, can user **bird** in group **sesame** rename `./foo` to `bar`?
`d----wx--- 2 root sesame 4096 Oct 7 14:00 .`
`----- 1 bird sesame 123 Oct 4 14:05 foo`
 - a. Yes, because **bird**'s group matches the group writable directory
 - b. Yes; permissions don't apply because **bird** owns **foo**
 - c. No, because the directory has no permissions for other users
 - d. No, because **bird** has no permissions on **foo**
 - e. No, because **bird** cannot read the directory

5. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr-x-wx--- 2 pat bg1 60 Jan 1 1:00 foo`
`-rwxrwxr-x 1 pat ted 0 Jan 1 1:00 foo/bar`
 - a. **bob** can create a new file in the directory
 - b. **pat** can create a new file in the directory
 - c. **bob** can list names in the directory
 - d. **bob** can access and write on the file
 - e. **pat** can rename the file
6. Given the following, can user **bird** in group **sesame** copy `./foo` to `bar`?
`drwxrw-r-x 2 root sesame 4096 Oct 7 14:00 .`
`-rwx-wx-wx 1 bird sesame 123 Oct 4 14:05 foo`
 - a. No, because the directory is not accessible to **bird**
 - b. No, because the directory has no write permissions for others
 - c. No, because **foo** has no read permissions for **bird**
 - d. Yes, because **bird** has write permissions on **foo**
 - e. Yes; permissions don't apply because **bird** owns **foo**
7. In an empty directory, what is output on your screen by:
`mkdir -p a/b/c 1/2/3 ; mv a/b 1/2/3 ; find . -name c`
 - a. `./1/2/3/c`
 - b. `./1/2/3/b/c`
 - c. `./1/2/a/b`
 - d. `./a/b/c`
 - e. `./1/2/3/a/b`
8. What value **umask** gives a new file permissions `r--r-----`?
 - a. 237
 - b. 110
 - c. 440
 - d. 220
 - e. 446
9. Given the following, can user **bird** in group **sesame** copy `./foo` to `bar`?
`drwx-wx--x 2 root sesame 4096 Oct 7 14:00 .`
`--wxrwxrwx 1 bird sesame 123 Oct 4 14:05 foo`
 - a. No, because the directory is not readable by **bird**
 - b. No, because **foo** has no read permissions for **bird**
 - c. Yes, because **bird** has write permissions on **foo**
 - d. Yes; permissions don't apply because **bird** owns **foo**
 - e. No, because the directory has no write permissions for **bird**
10. Given the following, can user **bird** in group **sesame** append to `./foo`?
`dr-xr-xr-x 2 root sesame 4096 Oct 7 14:00 .`
`-r-xrwxrwx 1 bird sesame 123 Oct 4 14:05 foo`
 - a. No, because **bird** has no write permissions on **foo**
 - b. No, because **bird** has no write permission on the directory
 - c. No, because execute permissions are not set for **bird** on **foo**
 - d. Yes; permissions don't apply because **bird** owns **foo**
 - e. No, because the directory is not accessible to **bird**

11. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-w-rw---x 2 bob ted 60 Jan 1 1:00 foo`
`--w-rwxrwx 1 pat bg1 0 Jan 1 1:00 foo/bar`
- bob** can list names in the directory
 - pat** can rename the file
 - bob** can access and write on the file
 - bob** can create a new file in the directory
 - pat** can access and write on the file
12. Dereference the following symlink **bar** into its equivalent absolute path:
`ln -s ../b/../../a/./foo /tmp/a/b/bar`
- `/tmp/a/b/bar`
 - `/tmp/foo`
 - `/tmp/a/foo`
 - `/tmp/b/foo`
 - `/tmp/b/bar`
13. Can three different files have the same inode number on three different file systems?
- no: inode numbers are unique across all file systems
 - yes: if the files are all names for the same inode
 - no: inode numbers only apply to directories, not files
 - yes: inode numbers are only unique inside a file system
 - no: you can't have inode numbers on three file systems
14. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to access and change or create new content (data) in the file **dir/bar** but not delete the file?
- Permissions **600** on directory **dir** and **700** on file **dir/bar**.
 - Permissions **400** on directory **dir** and **400** on file **dir/bar**.
 - Permissions **500** on directory **dir** and **600** on file **dir/bar**.
 - Permissions **300** on directory **dir** and **200** on file **dir/bar**.
 - Permissions **100** on directory **dir** and **100** on file **dir/bar**.
15. Which command counts the number of Unix permission groups you are in?
- `umask | wc`
 - `echo groups | wc`
 - `id | wc`
 - `wc groups`
 - `groups | wc`
16. A Unix/Linux "tarball" is:
- a multi-file directory containing individual compressed files
 - a single-file that contains individual compressed files
 - a single compressed file containing one uncompressed file
 - a multi-file directory containing individual uncompressed files
 - a single-file that contains individual uncompressed files

17. Given the following, can user **bird** in group **sesame** append to `./foo`?
`dr-xr-xr-x 2 root sesame 4096 Oct 7 14:00 .`
`-rw-r-xr-x 1 bird sesame 123 Oct 4 14:05 foo`
- No, because the directory is not accessible to **bird**
 - Yes, because **bird** has write permissions on **foo**
 - No, because execute permissions are not set for **bird** on **foo**
 - No, because **bird** has no write permission on the directory
 - Yes; permissions don't apply because **bird** owns **foo**
18. Process signals in increasing order of strength:
- TERM HUP KILL**
 - KILL HUP TERM**
 - HUP KILL TERM**
 - HUP TERM KILL**
 - TERM KILL HUP**
19. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr-xrwx-wx 2 pat pgg 60 Jan 1 1:00 foo`
`-r-xrwxr-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
- bob** can access and write on the file
 - bob** can list names in the directory
 - pat** can access and write on the file
 - pat** can rename the file
 - bob** can create a new file in the directory
20. Other than **root**, who can change the permissions of the following directory?
`dr-xrwxrwx 17 foo bar 4096 Apr 15 16:40 .`
- only **root** can change the permissions
 - user **foo** and any user in group **bar**
 - anyone except user **foo**
 - only user **foo**
 - only users in group **bar**
21. Which command line below does not show any lines from inside the file **out**?
- `sort out`
 - `wc out`
 - `more out`
 - `head out`
 - `tail out`
22. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr---wx--x 2 bob ted 60 Jan 1 1:00 foo`
`-r-xrwxrwx 1 pat bg1 0 Jan 1 1:00 foo/bar`
- bob** can create a new file in the directory
 - pat** can access and write on the file
 - pat** can rename the file
 - bob** can access and write on the file
 - bob** can list names in the directory
23. What minimal permissions must you have on a directory to be able to execute successfully the command `ls .` from *inside* the directory?
- r-x**
 - x**
 - rw-**
 - r--**
 - wx**

24. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d--xr----x 2 bob ted 60 Jan 1 1:00 foo`
`--w--w-r-x 1 bob bg1 0 Jan 1 1:00 foo/bar`
- pat** can rename the file
 - bob** can create a new file in the directory
 - pat** can access and write on the file
 - bob** can list names in the directory
 - bob** can access and write on the file
25. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr-xrwx--x 2 pat pgg 60 Jan 1 1:00 foo`
`--w----r-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
- bob** can access and write on the file
 - pat** can access and write on the file
 - pat** can rename the file
 - bob** can create a new file in the directory
 - bob** can list names in the directory
26. Which command usually goes in your `.bash_profile` file?
- `source ./bash_profile`
 - `source ./bashrc`
 - `./bashrc source`
 - `./bash_profile source`
 - `cat ./bashrc`
27. In an empty directory, what permissions are on file `???` after these commands:
`touch ??? *** ; chmod 111 *`
`chmod 222 ??? ; chmod 444 '***'`
- `rw-rw-rw-`
 - `-w--w--w-`
 - `--x--x--x`
 - `r--r--r--`
 - `-wx-wx-wx`
28. When a user named **bob** runs a command in an executable file owned by **foo**, in a directory owned by **root**, the file executes with the permissions of:
- root**
 - root and foo**
 - foo**
 - bob**
 - root and bob**
29. When a user named **bob** runs a command in a **setuid** executable file owned by **foo**, in a directory owned by **root**, the file executes with the permissions of:
- root**
 - bob**
 - root and bob**
 - foo**
 - root and foo**
30. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-wxr-xrw- 2 bob pgg 60 Jan 1 1:00 foo`
`-r-xrwxr-x 1 bob bg1 0 Jan 1 1:00 foo/bar`
- bob** can create a new file in the directory
 - bob** can list names in the directory
 - pat** can access and write on the file
 - pat** can rename the file
 - bob** can access and write on the file

31. Under what directory are system configuration files usually stored?
- `/usr/bin`
 - `/log/var/`
 - `/etc`
 - `/bin/`
 - `/var/log/`
32. The `-v` option to the `grep` command does what?
- selects lines that do not contain a match for the supplied pattern
 - turns off the translation of unprintable characters
 - prints the version number of the `grep` command
 - selects lines that do not contain unprintable characters
 - turns on the translation of unprintable characters
33. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d--xrw-wx 2 bob ted 60 Jan 1 1:00 foo`
`-r-x-w-r-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
- pat** can access and write on the file
 - pat** can rename the file
 - bob** can create a new file in the directory
 - bob** can list names in the directory
 - bob** can access and write on the file
34. In an empty directory, what is output on your screen by:
`mkdir -p a/b/c 1/2/3 ; mv a 1/2 ; find . -name c`
- `./1/a`
 - `./1/2/3/a/b/c`
 - `./1/2/a/b/c`
 - `./1/2/a`
 - `./1/2/3/a/b`
35. What command terminates processes based on their name (not safe!):
- `killall`
 - `crontab`
 - `kill`
 - `ps lxww`
 - `dmesg`
36. In an empty directory, what permissions are on file `???` after these commands:
`touch ??? *** ; chmod 111 *`
`chmod 222 ? ; chmod 444 '*'`
- `r--r--r--`
 - `-w--w--w-`
 - `-wx-wx-wx`
 - `rw-rw-rw-`
 - `--x--x--x`
37. To bring a background shell job into the foreground, type:
- `fg`
 - `bg`
 - `kill %1`
 - `[Ctrl-Z]`
 - `[Ctrl-D]`
38. Given my directory `dir` and my file `dir/c` owned by me, which permissions allow me to access and change or create new content (data) in the file `dir/c` but not delete the file?
- Permissions `100` on directory `dir` and `100` on file `dir/c`.
 - Permissions `100` on directory `dir` and `200` on file `dir/c`.
 - Permissions `400` on directory `dir` and `400` on file `dir/c`.
 - Permissions `600` on directory `dir` and `700` on file `dir/c`.
 - Permissions `200` on directory `dir` and `200` on file `dir/c`.

39. The password **:x:** in **/etc/passwd** means:
- the unencrypted password is stored in the group file
 - the account is locked
 - the encrypted password is **"x"**
 - the password is locked
 - the encrypted password is stored in the shadow file
40. What value **umask** gives a new file permissions **r--r-----?**
- 440
 - 337
 - 446
 - 220
 - 110
41. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
d---rwx--x 2 pat pgg 60 Jan 1 1:00 foo
--w----rwx 1 bob bg1 0 Jan 1 1:00 foo/bar
- bob** can list names in the directory
 - pat** can rename the file
 - bob** can create a new file in the directory
 - bob** can access and write on the file
 - pat** can access and write on the file
42. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
dr-xr-xrwx 2 pat bg1 60 Jan 1 1:00 foo
-rwxrwxr-x 1 pat ted 0 Jan 1 1:00 foo/bar
- pat** can create a new file in the directory
 - bob** can access and write on the file
 - bob** can list names in the directory
 - pat** can rename the file
 - bob** can rename the file
43. Under what directory are system log files usually stored?
- /usr/bin**
 - /log/var**
 - /etc/log**
 - /var/log**
 - /bin/**
44. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
d-w---xr-x 2 pat ted 60 Jan 1 1:00 foo
-rwxr-xrwx 1 pat bg2 0 Jan 1 1:00 foo/bar
- pat** can access and write on the file
 - bob** can access and write on the file
 - bob** can rename the file
 - bob** can create a new file in the directory
 - bob** can list names in the directory
45. The **minimum** permissions you need to delete a file **foo** from directory **a** are:
- wx** on **a**, none on **foo**
 - rw** on **a**, none on **foo**
 - rw** on **a**, **rw** on **foo**
 - wx** on **a**, **r** on **foo**
 - wx** on **a**, **w** on **foo**

46. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
dr-xrw-rwx 2 pat bg1 60 Jan 1 1:00 foo
-rwxrwxrwx 1 pat ted 0 Jan 1 1:00 foo/bar
- pat** can create a new file in the directory
 - bob** can list names in the directory
 - bob** can rename the file
 - pat** can rename the file
 - bob** can access and write on the file
47. Which command line creates a directory **dir** into which anyone can put a file, but in which nobody can see the names of the files that are there?
- mkdir dir ; chmod 333 .**
 - mkdir dir ; cd dir ; chmod ugo-rw .**
 - mkdir dir ; chmod 333 dir**
 - mkdir dir ; chmod 222 dir**
 - mkdir dir ; cd dir ; chmod ugo=w .**
48. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
d-wx--x--x 2 bob ted 60 Jan 1 1:00 foo
-r-xr-xrwx 1 pat bg2 0 Jan 1 1:00 foo/bar
- pat** can rename the file
 - bob** can create a new file in the directory
 - bob** can access and write on the file
 - bob** can list names in the directory
 - pat** can access and write on the file
49. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to access and change or create new content (data) in the file **dir/bar** but not delete the file?
- Permissions **500** on directory **dir** and **100** on file **dir/bar**.
 - Permissions **400** on directory **dir** and **400** on file **dir/bar**.
 - Permissions **600** on directory **dir** and **700** on file **dir/bar**.
 - Permissions **200** on directory **dir** and **200** on file **dir/bar**.
 - Permissions **500** on directory **dir** and **200** on file **dir/bar**.
50. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
dr-xrwxrw- 2 pat pgg 60 Jan 1 1:00 foo
--w----r-x 1 bob bg1 0 Jan 1 1:00 foo/bar
- pat** can access and write on the file
 - bob** can access and write on the file
 - bob** can list names in the directory
 - bob** can create a new file in the directory
 - pat** can rename the file
51. What value **umask** gives a new file permissions **r--r-----?**
- 110
 - 326
 - 447
 - 440
 - 220

52. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-wx----w- 2 pat pgg 60 Jan 1 1:00 foo`
`-rwxrwxr-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
 a. **bob** can create a new file in the directory
 b. **bob** can access and write on the file
 c. **pat** can access and write on the file
 d. **pat** can rename the file
 e. **bob** can list names in the directory
53. Given the following, can user **bird** in group **sesame** append to **foobar**?
`drwx--xrw 2 root sesame 4096 Oct 7 14:00 .`
`-rw----- 1 bird sesame 1024 Oct 4 14:05 foobar`
 a. No, because the directory is not accessible to **bird**
 b. No, because **sesame** has no write permissions on **foobar**
 c. Yes, because **bird** has write permissions on **foobar**
 d. No, because execute permissions are not set for **bird** on **foobar**
 e. Yes, because **bird** owns **foobar**
54. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d--x----w- 2 pat ted 60 Jan 1 1:00 foo`
`--w-r-xrwx 1 pat bg2 0 Jan 1 1:00 foo/bar`
 a. **pat** can access and write on the file
 b. **bob** can access and write on the file
 c. **bob** can create a new file in the directory
 d. **bob** can list names in the directory
 e. **pat** can rename the file
55. Which **crontab** line executes at **13:54** every day?
 a. `13 54 * * * command` b. `* * * 13 54 command`
 c. `* * * 54 13 command` d. `13 * * * 54 command`
 e. `54 13 * * * command`
56. What value **umask** gives a new directory permissions **rw--w---x**?
 a. 211 b. 156 c. 621 d. 421 e. 432
57. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`drw-r-xrwx 2 pat bg1 60 Jan 1 1:00 foo`
`-rwxrwxr-x 1 pat ted 0 Jan 1 1:00 foo/bar`
 a. **pat** can rename the file
 b. **bob** can rename the file
 c. **bob** can access and write on the file
 d. **pat** can create a new file in the directory
 e. **bob** can list names in the directory

58. If I mount **sda1** on **/one** and **sda2** on **/two**, how can I link the existing file **/one/foo** to the new pathname **/two/bar**?
 a. `ln /one/bar /two/foo`
 b. `ln -s /one/foo /two/bar`
 c. `ln /one/foo /two/bar`
 d. `ln /two/bar /one/foo`
 e. `ln -s /two/bar /one/foo`
59. What command changes a user's password?
 a. `chpasswd` b. `chsh` c. `mkpasswd`
 d. `password` e. `passwd`
60. In an empty directory, what is output on your screen by:
`mkdir -p a/b/c 1/2/3 ; mv a/b 1/2 ; find . -name c`
 a. `./1/2/a/b` b. `./1/2/c` c. `./1/a/b`
 d. `./a/b/c` e. `./1/2/b/c`
61. Dereference the following symlink **bar** into its equivalent absolute path:
`ln -s ../b/../../a/./foo /tmp/a/b/bar`
 a. `/tmp/foo` b. `/tmp/b/bar` c. `/tmp/a/b/bar`
 d. `/tmp/b/foo` e. `/tmp/a/foo`
62. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`drw-----x 2 pat ted 60 Jan 1 1:00 foo`
`--w--w-r-x 1 pat bg1 0 Jan 1 1:00 foo/bar`
 a. **bob** can rename the file
 b. **bob** can list names in the directory
 c. **pat** can access and write on the file
 d. **bob** can access and write on the file
 e. **bob** can create a new file in the directory
63. What is the output of this command line in an empty directory:
`touch .a .b .c ; echo [.]*`
 a. `.a .b .c`
 b. `[.]*`
 c. `.. .a .b .c`
 d. no output
 e. an error message from **echo** saying `[.]*` does not exist
64. Which command line creates a directory **dir** into which anyone can put a file, but in which nobody can see the names of the files that are there?
 a. `mkdir dir ; chmod 777 dir`
 b. `mkdir dir ; cd dir ; chmod go-x .`
 c. `mkdir dir ; cd dir ; chmod go+wx .`
 d. `mkdir dir ; chmod 777 .`
 e. `mkdir dir ; chmod 333 dir`

65. The shadow password file is used:
- to hide encrypted passwords from viewing by ordinary users
 - to allow passwords to exist on partitions other than the ROOT
 - to store secondary passwords for times when you forget your main one
 - to keep a back-up of the main password file in case of corruption
 - to reduce the size of the main password file for faster access
66. If the current directory contains 10 visible files and 5 visible sub-directories, what is the output on your screen of this command: `ls -d */.`
- no output
 - an error message because `*/.` does not exist
 - 15 pathnames
 - 5 directory names
 - `*/.`
67. To change your own account password, use this exact command line:
- `$ passwd cst8207.idallen.ca`
 - `$ passwd`
 - `$ passwd idallen-ubuntu`
 - `$ passwd cst8207`
 - `$ passwd 10.50.254.150`
68. User `bob` is in groups `bg1` and `bg2`. User `pat` is in group `pgg`.
`dr-x----wx 2 pat ted 60 Jan 1 1:00 foo`
`-r-xr-xrwx 1 pat bg1 0 Jan 1 1:00 foo/bar`
- `bob` can list names in the directory
 - `pat` can access and write on the file
 - `pat` can rename the file
 - `bob` can access and write on the file
 - `bob` can create a new file in the directory
69. Given my directory `dir` and my file `dir/bar` owned by me, which permissions allow me to delete the file `dir/bar` from the directory, but not change the content (data) in the file?
- Permissions `100` on directory `dir` and `200` on file `dir/bar`.
 - Permissions `100` on directory `dir` and `100` on file `dir/bar`.
 - Permissions `500` on directory `dir` and `400` on file `dir/bar`.
 - Permissions `300` on directory `dir` and `300` on file `dir/bar`.
 - Permissions `300` on directory `dir` and `500` on file `dir/bar`.

70. User `bob` is in groups `bg1` and `bg2`. User `pat` is in group `pgg`.
`d-w-rwx-wx 2 bob ted 60 Jan 1 1:00 foo`
`-r-xrwxrwx 1 pat bg2 0 Jan 1 1:00 foo/bar`
- `pat` can access and write on the file
 - `pat` can rename the file
 - `bob` can list names in the directory
 - `bob` can create a new file in the directory
 - `bob` can access and write on the file
71. The output of the `whoami` command is:
- a list of users logged in to the system
 - a list of accounts in the password file
 - your userid
 - your HOME directory
 - the current directory
72. User `bob` is in groups `bg1` and `bg2`. User `pat` is in group `pgg`.
`dr-xrwx-wx 2 pat ted 60 Jan 1 1:00 foo`
`-r-xr-xrwx 1 pat bg2 0 Jan 1 1:00 foo/bar`
- `pat` can create a new file in the directory
 - `pat` can access and write on the file
 - `bob` can access and write on the file
 - `bob` can rename the file
 - `bob` can list names in the directory
73. User `bob` is in groups `bg1` and `bg2`. User `pat` is in group `pgg`.
`d-wx-w-rwx 2 pat bg1 60 Jan 1 1:00 foo`
`-rwxrwxrwx 1 pat ted 0 Jan 1 1:00 foo/bar`
- `bob` can list names in the directory
 - `bob` can rename the file
 - `bob` can create a new file in the directory
 - `pat` can create a new file in the directory
 - `bob` can access and write on the file
74. In a directory containing one file named `dog`, what is the output on your screen after this command line: `2>/dev/null ls nosuchfile`
- `nosuchfile`
 - no output
 - `dog`
 - `ls: nosuchfile: No such file or directory`
 - `bash: 2>/dev/null: command not found`

75. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr-x-wx--x 2 bob ted 60 Jan 1 1:00 foo`
`-r-xr-xrwx 1 pat bg1 0 Jan 1 1:00 foo/bar`
- bob** can access and write on the file
 - bob** can list names in the directory
 - bob** can create a new file in the directory
 - pat** can access and write on the file
 - pat** can rename the file
76. The **minimum** permissions you need to read a file **foo** in directory **a** are:
- wx** on **a**, none on **foo**
 - rw**x on **a**, none on **foo**
 - wx** on **a**, **w** on **foo**
 - x** on **a**, **r** on **foo**
 - rw**x on **a**, **rw** on **foo**
77. Which of these commands makes a file owned by me, also readable by me?
- `chmod r=u ./myfile`
 - `chmod u+r ./myfile`
 - `umask 400 myfile`
 - `chmod r+u myfile`
 - `umask 300 ./myfile`
78. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr--r-x-w- 2 bob pgg 60 Jan 1 1:00 foo`
`-rwxrwxr-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
- bob** can list names in the directory
 - pat** can access and write on the file
 - pat** can rename the file
 - bob** can access and write on the file
 - bob** can create a new file in the directory
79. Given my directory **dir** and my file **dir/foo** owned by me, which permissions allow me to access and change or create new content (data) in the file **dir/foo** but not delete the file?
- Permissions **100** on directory **dir** and **100** on file **dir/foo**.
 - Permissions **600** on directory **dir** and **700** on file **dir/foo**.
 - Permissions **300** on directory **dir** and **200** on file **dir/foo**.
 - Permissions **500** on directory **dir** and **600** on file **dir/foo**.
 - Permissions **400** on directory **dir** and **400** on file **dir/foo**.
80. What command displays the groups you are in?
- `grouprint`
 - `mkgroups`
 - `gpaswd`
 - `lstgroups`
 - `groups`

81. Given the following, can user **bird** in group **sesame** remove **./foo**?
`drwxr-xrwx 2 root sesame 4096 Oct 7 14:00 .`
`-rwxrwxrwx 1 bird sesame 123 Oct 4 14:05 foo`
- No, because **bird** has no write permission on the directory
 - Yes; permissions don't apply because **bird** owns **foo**
 - Yes, because **bird** has full permissions on **foo**
 - No, because the directory is not accessible to **bird**
 - Yes, because **bird** matches the writable other permissions
82. What value to **chmod** would change the permissions on a file to **r-----rw-?**
- 122**
 - 654**
 - 406**
 - 102**
 - 322**
83. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d--x-wx--- 2 bob pgg 60 Jan 1 1:00 foo`
`-r-x-w-r-x 1 bob bg1 0 Jan 1 1:00 foo/bar`
- pat** can rename the file
 - bob** can access and write on the file
 - bob** can create a new file in the directory
 - bob** can list names in the directory
 - pat** can access and write on the file
84. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d--xr-x-w- 2 bob pgg 60 Jan 1 1:00 foo`
`--w----r-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
- bob** can access and write on the file
 - pat** can access and write on the file
 - bob** can create a new file in the directory
 - pat** can rename the file
 - bob** can list names in the directory
85. Given the following, can user **bird** in group **sesame** copy **./foo** to **bar**?
`drwxr-xrwx 2 root sesame 4096 Oct 7 14:00 .`
`-r-xr-xr-x 1 bird sesame 123 Oct 4 14:05 foo`
- Yes; permissions don't apply because **bird** owns **foo**
 - Yes, because **bird** has read permissions on **foo**
 - No, because the directory is not accessible to **bird**
 - No, because **foo** has no write permissions for **bird**
 - No, because the directory has no write permissions for **bird**
86. What value to **chmod** would change the permissions on a file to **rw-r--r--?**
- 344**
 - 211**
 - 644**
 - 244**
 - 311**

87. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d--xrwX--x 2 bob ted 60 Jan 1 1:00 foo`
`----rw--w- 1 bob bg1 0 Jan 1 1:00 foo/bar`
- bob** can create a new file in the directory
 - bob** can access and write on the file
 - pat** can rename the file
 - pat** can access and write on the file
 - bob** can list names in the directory
88. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to delete the file **dir/bar** from the directory, but not change the content (data) in the file?
- Permissions **600** on directory **dir** and **300** on file **dir/bar**.
 - Permissions **700** on directory **dir** and **500** on file **dir/bar**.
 - Permissions **500** on directory **dir** and **500** on file **dir/bar**.
 - Permissions **600** on directory **dir** and **500** on file **dir/bar**.
 - Permissions **700** on directory **dir** and **200** on file **dir/bar**.
89. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`dr---wx--x 2 bob ted 60 Jan 1 1:00 foo`
`--w--w-r-x 1 bob bg2 0 Jan 1 1:00 foo/bar`
- bob** can access and write on the file
 - bob** can list names in the directory
 - bob** can create a new file in the directory
 - pat** can access and write on the file
 - pat** can rename the file
90. If I mount one file system on directory **/a** and another file system on directory **/b**, how can I link the existing file **/a/foo** to the new pathname **/b/new**?
- `ln /b/new /a/foo` `b. ln /a/foo /b/new`
 - `ln -s /b/new /a/foo` `d. ln /a/new /b/foo`
 - `ln -s /a/foo /b/new`
91. Given my directory **dir** and my file **dir/c** owned by me, which permissions allow me to delete the file **dir/c** from the directory, but not change the content (data) in the file?
- Permissions **500** on directory **dir** and **400** on file **dir/c**.
 - Permissions **100** on directory **dir** and **100** on file **dir/c**.
 - Permissions **100** on directory **dir** and **200** on file **dir/c**.
 - Permissions **300** on directory **dir** and **300** on file **dir/c**.
 - Permissions **300** on directory **dir** and **500** on file **dir/c**.
92. Dereference the following symlink **bar** into its equivalent absolute path:
`ln -s ../b/../../b/../../foo /tmp/a/b/bar`
- `/tmp/foo` `b. /tmp/b/foo` `c. /tmp/b/bar`
 - `/tmp/a/b/bar` `e. /tmp/a/foo`

93. What command displays the kernel ring buffer of log messages:
- `ps lxww` `b. psmine` `c. crontab`
 - `dmesg` `e. showall`
94. To list your personal crontab, type:
- `/var/log/crontab` `b. crontab -l`
 - `atq` `d. cat crontab`
 - `/etc/crontab`
95. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-wxrwX-w- 2 pat ted 60 Jan 1 1:00 foo`
`-r-xr-xrwX 1 pat bg1 0 Jan 1 1:00 foo/bar`
- pat** can access and write on the file
 - pat** can rename the file
 - bob** can create a new file in the directory
 - bob** can list names in the directory
 - bob** can access and write on the file
96. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`drw-rw-rwX 2 pat bg1 60 Jan 1 1:00 foo`
`-rwxrwxrwx 1 pat ted 0 Jan 1 1:00 foo/bar`
- bob** can rename the file
 - pat** can create a new file in the directory
 - bob** can access and write on the file
 - pat** can rename the file
 - bob** can list names in the directory
97. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-wX-w-rwX 2 pat bg2 60 Jan 1 1:00 foo`
`-rwxrwxrwx 1 pat ted 0 Jan 1 1:00 foo/bar`
- bob** can access and write on the file
 - pat** can rename the file
 - bob** can list names in the directory
 - bob** can rename the file
 - bob** can create a new file in the directory
98. To show all your one-time scheduled commands, type:
- `cat crontab` `b. atq`
 - `crontab -l` `d. /etc/crontab`
 - `/var/log/crontab`

99. Given the following, can user **bird** in group **sesame** append to **foobar**?
`drwxrw-rwx 2 root sesame 4096 Oct 7 14:00 .`
`-rw-rw-r-- 1 bird sesame 1024 Oct 4 14:05 foobar`
- No, because execute permissions are not set for **bird** on **foobar**
 - No, because the directory is not accessible to **bird**
 - Yes, because **sesame** has write permissions on **foobar**
 - Yes, because **bird** owns **foobar**
 - Yes, because **bird** has write permissions on **foobar**
100. Given my directory **dir** and my file **dir/f** owned by me, which permissions allow me to delete the file **dir/f** from the directory, but not change the content (data) in the file?
- Permissions **600** on directory **dir** and **500** on file **dir/f**.
 - Permissions **600** on directory **dir** and **300** on file **dir/f**.
 - Permissions **500** on directory **dir** and **500** on file **dir/f**.
 - Permissions **700** on directory **dir** and **200** on file **dir/f**.
 - Permissions **300** on directory **dir** and **500** on file **dir/f**.
101. Given this successful command line (note the dot argument):
`cd /home/foo ; mkdir bar ; cd bar ; chmod a-x .`
 Which of the following subsequent commands will execute without any "permission denied" errors?
- `ls /home/foo/bar/.`
 - `ls /home/foo/bar/..`
 - `ls ..`
 - `ls .`
102. Which of these statements is true?
- You can make a hard link to a directory.
 - You only need "**r--**" permission on directory "**foo**" for "`ls -l foo`" to work.
 - To make a hard link to file "**foo**" named "**bar**", file "**foo**" must exist.
 - If you give me write permission on a file owned by you, I can then use **chmod** to change its permissions.
 - The "**ln**" command takes two arguments, so the maximum number of hard links a file can have is two.
103. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`d-w---xr-- 2 pat ted 60 Jan 1 1:00 foo`
`-rwxrwxrwx 1 pat bg2 0 Jan 1 1:00 foo/bar`
- bob** can list names in the directory
 - bob** can create a new file in the directory
 - bob** can access and write on the file
 - pat** can access and write on the file
 - bob** can rename the file

104. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
`drw---x--- 2 pat bg2 60 Jan 1 1:00 foo`
`-r-----w- 1 pat ted 0 Jan 1 1:00 foo/bar`
- bob** can rename the file
 - bob** can list names in the directory
 - pat** can create a new file in the directory
 - pat** can rename the file
 - bob** can access and write on the file
105. Which of these statements is true?
- you can only remove a file name if the file is owned by you
 - you can only make links to files owned by you
 - you can change the permissions of any file to which you can write
 - you may be able to rename a file even if you do not own the file
 - you can only remove a file name if the file is writable by you
106. In a directory containing one file named **dog**, what is the output on your screen after this command line: `1>/dev/null ls *`
- `bash: 1>/dev/null: command not found`
 - `ls: *: No such file or directory`
 - `*`
 - `dog`
 - no output
107. What command manipulates your personal list of repeated scheduled commands:
- `showall`
 - `dmesg`
 - `ps lxww`
 - `crontab`
 - `psmine`
108. Given my directory **dir** and my file **dir/f** owned by me, which permissions allow me to access and change or create new content (data) in the file **dir/f** but not delete the file?
- Permissions **600** on directory **dir** and **700** on file **dir/f**.
 - Permissions **400** on directory **dir** and **400** on file **dir/f**.
 - Permissions **100** on directory **dir** and **200** on file **dir/f**.
 - Permissions **200** on directory **dir** and **200** on file **dir/f**.
 - Permissions **500** on directory **dir** and **100** on file **dir/f**.
109. Which command removes adjacent duplicate lines from a file?
- `dupl`
 - `unique`
 - `duplicate`
 - `uniq`
 - `dup`
110. When a personal **crontab** job runs, the current working directory is set to:
- the directory with the name **/root**
 - the HOME directory of the user who created the job
 - the current directory that was in use when the **crontab** job was created
 - the system ROOT directory
 - the directory with the name **/home**

111. The *difference* between the system (**root**) crontab and all the user (personal) crontabs is:
- the system crontab has the date and time in it
 - the personal crontab has the date and time in it
 - the system crontab also has the userid in it
 - the personal crontab also has the userid in it
 - the personal crontab only runs commands once
112. The **minimum** permissions you need to move a file **foo** from directory **a** to directory **b** are:
- wx** on **a**, **wx** on **b**, none on **foo**
 - wx** on **a**, **wx** on **b**, **w** on **foo**
 - rw**x on **a**, **wx** on **b**, none on **foo**
 - rw**x on **a**, **wx** on **b**, **rw** on **foo**
 - wx** on **a**, **wx** on **b**, **r** on **foo**
113. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
dr--xr----x 2 bob ted 60 Jan 1 1:00 foo
-r-x-w-rwx 1 pat bg2 0 Jan 1 1:00 foo/bar
```
- bob** can create a new file in the directory
  - pat** can rename the file
  - bob** can access and write on the file
  - bob** can list names in the directory
  - pat** can access and write on the file
114. What value **umask** gives a new file permissions **r--r-----**?
- 220
  - 440
  - 226
  - 110
  - 446
115. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
dr-x-wx--x 2 bob ted 60 Jan 1 1:00 foo
-r-x-w-r-x 1 bob bg1 0 Jan 1 1:00 foo/bar
```
- bob** can access and write on the file
 - bob** can create a new file in the directory
 - pat** can access and write on the file
 - bob** can list names in the directory
 - pat** can rename the file
116. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
dr---wx--- 2 pat bg2 60 Jan 1 1:00 foo
-rw-rw-r-x 1 pat ted 0 Jan 1 1:00 foo/bar
```
- bob** can rename the file
  - bob** can access and write on the file
  - bob** can list names in the directory
  - pat** can rename the file
  - pat** can create a new file in the directory

117. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to delete the file **dir/bar** from the directory, but not change the content (data) in the file?
- Permissions **300** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **500** on directory **dir** and **500** on file **dir/bar**.
  - Permissions **100** on directory **dir** and **300** on file **dir/bar**.
  - Permissions **300** on directory **dir** and **400** on file **dir/bar**.
  - Permissions **100** on directory **dir** and **500** on file **dir/bar**.
118. When an **at** job runs, the current working directory is set to:
- the system **ROOT** directory
  - the directory with the name **/root**
  - the current directory that was in use when the **at** job was created
  - the **HOME** directory of the user who created the job
  - the directory with the name **/home**
119. The **minimum** permissions you need to append to a file **foo** in directory **a** are:
- rw**x on **a**, none on **foo**
  - wx** on **a**, none on **foo**
  - wx** on **a**, **w** on **foo**
  - rw**x on **a**, **rw** on **foo**
  - x** on **a**, **w** on **foo**
120. The **minimum** permissions you need to copy a file **foo** from directory **a** to directory **b** are:
- wx** on **a**, **wx** on **b**, **rw** on **foo**
  - wx** on **a**, **wx** on **b**, none on **foo**
  - x** on **a**, **wx** on **b**, **r** on **foo**
  - rw**x on **a**, **wx** on **b**, none on **foo**
  - rx** on **a**, **wx** on **b**, **w** on **foo**
121. Given the following, can user **bird** in group **sesame** append to **./foo**?
- ```
dr-xr--r-x 2 root sesame 4096 Oct 7 14:00 .
-rw-rw-r-- 1 bird sesame 123 Oct 4 14:05 foo
```
- No, because **bird** has no write permission on the directory
 - No, because the directory is not accessible to **bird**
 - No, because execute permissions are not set for **bird** on **foo**
 - Yes; permissions don't apply because **bird** owns **foo**
 - Yes, because **bird** has write permissions on **foo**
122. What permissions are given to **newfile** after this command line:
- ```
umask 326 ; touch newfile
```
- wx-w-r-x**
  - r--r-----**
  - wxr-----**
  - r--r-x--x**
  - wx-w-rw-**

123. The **minimum** permissions you need to link a file **foo** from directory **a** to directory **b** are:
- wx** on **a**, **wx** on **b**, **r** on **foo**
  - wx** on **a**, **wx** on **b**, **w** on **foo**
  - rw** on **a**, **wx** on **b**, none on **foo**
  - rw** on **a**, **wx** on **b**, **rw** on **foo**
  - x** on **a**, **wx** on **b**, none on **foo**
124. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to access and change or create new content (data) in the file **dir/bar** but not delete the file?
- Permissions **600** on directory **dir** and **700** on file **dir/bar**.
  - Permissions **100** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **200** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **100** on directory **dir** and **100** on file **dir/bar**.
  - Permissions **400** on directory **dir** and **400** on file **dir/bar**.
125. In an empty directory, what is output on your screen by:
- ```
mkdir -p a/b/c 1/2/3 ; mv a/b/c 1/2 ; find . -name c
```
- `./1/2/3/a/b/c`
 - `./1/a/b/c`
 - `./1/2/c`
 - `./1/2/a/b/c`
 - `./1/2/b/c`
126. Which command line would show the inode number of a file?
- `cat -l file`
 - `ls -i file`
 - `find -i file`
 - `ls -l file`
 - `cat -i file`
127. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
d--x--xrwx- 2 bob pgg 60 Jan 1 1:00 foo
-r-xrwx-w- 1 bob bg2 0 Jan 1 1:00 foo/bar
```
- bob** can access and write on the file
  - pat** can rename the file
  - bob** can list names in the directory
  - bob** can create a new file in the directory
  - pat** can access and write on the file
128. Which command line displays all the non-hidden names in the current directory that contain the case-insensitive word **hi** (and no other names)?
- `echo *(H,h,I,i)*`
  - `echo ?[HhIiHhIi]?`
  - `echo *[Hh][Ii]*`
  - `echo *[hiHI]*`
  - `echo ?[HhIi]?`
129. What command line shows only your own processes, not all processes?
- `dmesg`
  - `crontab`
  - `psmine`
  - `ps lxww`
  - `showall`

130. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
dr-xr-x-w- 2 bob pgg 60 Jan 1 1:00 foo
-r-xrwxr-x 1 bob bg1 0 Jan 1 1:00 foo/bar
```
- bob** can access and write on the file
 - pat** can access and write on the file
 - pat** can rename the file
 - bob** can list names in the directory
 - bob** can create a new file in the directory
131. What does the **-v** option to the **grep** command do?
- prints the version number of the **grep** command
 - selects lines that do not contain a match for the supplied pattern
 - turns on the translation of unprintable characters
 - turns off the translation of unprintable characters
 - selects lines that do not contain unprintable characters
132. The signal sent to a foreground process by typing the **[Ctrl-C]** key is:
- SIGKILL**
 - SIGSTOP**
 - SIGINT**
 - SIGTERM**
 - SIGHUP**
133. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
d--x-----x 2 pat pgg 60 Jan 1 1:00 foo
-r-xrwx-w- 1 bob bg1 0 Jan 1 1:00 foo/bar
```
- pat** can rename the file
  - bob** can access and write on the file
  - bob** can create a new file in the directory
  - pat** can access and write on the file
  - bob** can list names in the directory
134. Which command line below does not show any lines from inside the file **bat**?
- `more bat`
  - `ls bat`
  - `less bat`
  - `head bat`
  - `tail bat`