

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

**One-Answer Multiple Choice      180 Questions      Weight 40%**

- ☞ 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.
- ☞ The answer to the question about reading the test instructions is: **123**

191. Answer **191** is
192. Answer **192** is
193. Answer **193** is
194. Answer **194** is
195. Answer **195** is
196. Answer **196** is

Your Test Version is:

**D C E A B B**

Fill in the bubbles for the above six letters as six answers **191** through **196** on the back side of the Scantron form, in the lower-right-most answer column.

1. If **a=123** and **b=456** then what is the output of the following sequence of commands: `if [$a = $b] ; then echo $a ; fi`
  - a. no output
  - b. **123**
  - c. `test: a=123: integer expression expected`
  - d. `bash: [123: command not found`
  - e. `test: $a: string expression expected`
2. Which of the following commands would result in an error?
  - a. `[ a -eq 4 ]`
  - b. `[ a != 4 ]`
  - c. `[ 3 = 4 ]`
  - d. `[ a = 4 ]`
  - e. `[ 3 -eq 4 ]`
3. What command would you use to see the command that **at** job number **2** will run?
  - a. `atq 2`
  - b. `at -c 2`
  - c. `at -m 2`
  - d. `at -v 2`
  - e. `at -l 2`
4. What value **umask** gives a new directory permissions **rw--w---x**?
  - a. **211**
  - b. **621**
  - c. **432**
  - d. **156**
  - e. **421**
5. Which command line makes 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 ; chmod 333 dir`
  - d. `mkdir dir ; chmod 777 .`
  - e. `mkdir dir ; cd dir ; chmod go+wx .`

6. Which of the following, as first line of a shell script, would mean that when the script is run as a command, `/bin/sh` will be run with the `-u` option to process the script.
  - a. `#!/bin/sh -u`
  - b. `!!/bin/sh -u`
  - c. `!/bin/sh -u`
  - d. `!#/bin/sh -u`
  - e. `#!/bin/sh -u`
7. 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
```

  - a. Yes, because **bird** has write permissions on **foo**
  - b. No, because execute permissions are not set for **bird** on **foo**
  - c. No, because the directory is not accessible to **bird**
  - d. No, because **bird** has no write permission on the directory
  - e. Yes; permissions don't apply because **bird** owns **foo**
8. 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?
  - a. Permissions **100** on directory **dir** and **200** on file **dir/bar**.
  - b. Permissions **500** on directory **dir** and **400** on file **dir/bar**.
  - c. Permissions **300** on directory **dir** and **300** on file **dir/bar**.
  - d. Permissions **100** on directory **dir** and **100** on file **dir/bar**.
  - e. Permissions **300** on directory **dir** and **500** on file **dir/bar**.
9. 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. **bob** can rename the file
  - b. **bob** can list names in the directory
  - c. **bob** can access and write on the file
  - d. **pat** can create a new file in the directory
  - e. **pat** can rename the file
10. What does the `-v` option to the **fgrep** command do?
  - a. turns on the translation of unprintable characters
  - b. selects lines that do not contain a match for the supplied pattern
  - c. selects lines that do not contain unprintable characters
  - d. prints the version number of the **fgrep** command
  - e. turns off the translation of unprintable characters
11. If a shell script **myscript.sh** is called this way:
 

```
./myscript.sh a b c
```

 and the first line inside the script below the script header is
 

```
echo "$#$1" ; shift
```

 what is the output of that line?
  - a. **3a**
  - b. **2b**
  - c. **4c**
  - d. **3b**
  - e. **2a**

12. Which command usually goes in your `.bash_profile` file?
- `source ~/.bashrc`
  - `./ .bash_profile source`
  - `./ .bashrc source`
  - `cat ~/.bashrc`
  - `source ~/.bash_profile`
13. Dereference the following symlink `bar` into its equivalent absolute path:  
`ln -s ../b/../../a/./foo /tmp/a/b/bar`
- `/tmp/foo`
  - `/tmp/a/foo`
  - `/tmp/b/foo`
  - `/tmp/a/b/bar`
  - `/tmp/b/bar`
14. What is the output on your screen of the following sequence of commands:  
`x=ok ; y=ok ; [ x = y ]`
- `test: x: integer expression expected`
  - `1`
  - no output on screen
  - `bash: x: command not found`
  - `0`
15. Which of the following options for `bash` or `sh` might be useful for debugging a shell script?
- `-c`
  - `-z`
  - `-l`
  - `-x`
  - `-r`
16. User `bob` is in groups `bg1` and `bg2`. User `pat` is in group `pgg`.  
`d-wxrw-w- 2 pat ted 60 Jan 1 1:00 foo`  
`-r-xr-xrw 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
  - `pat` can rename the file
  - `pat` can access and write on the file
  - `bob` can create a new file in the directory
17. 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
  - `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
18. To list your personal crontab, type:
- `crontab -l`
  - `/var/log/crontab`
  - `cat crontab`
  - `atq`
  - `/etc/crontab`

19. 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 list names in the directory
  - `bob` can access and write on the file
  - `bob` can create a new file in the directory
20. 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`
- No, because `foo` has no read permissions for `bird`
  - No, because the directory is not accessible to `bird`
  - Yes; permissions don't apply because `bird` owns `foo`
  - No, because the directory has no write permissions for others
  - Yes, because `bird` has write permissions on `foo`
21. 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`
- `bob` can access and write on the file
  - `bob` can rename the file
  - `pat` can access and write on the file
  - `bob` can create a new file in the directory
  - `bob` can list names in the directory
22. Given the following, can user `bird` in group `sesame` remove `./foo`?  
`drwxr-xrw 2 root sesame 4096 Oct 7 14:00 .`  
`-rwxrwxrwx 1 bird sesame 123 Oct 4 14:05 foo`
- Yes, because `bird` matches the writable other permissions
  - No, because the directory is not accessible to `bird`
  - Yes, because `bird` has full permissions on `foo`
  - Yes; permissions don't apply because `bird` owns `foo`
  - No, because `bird` has no write permission on the directory
23. 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 create a new file in the directory
  - `bob` can rename the file
  - `pat` can access and write on the file
  - `bob` can access and write on the file
  - `bob` can list names in the directory

24. 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`
- 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
25. What value to **chmod** would change the permissions on a file to **rw-r--r--**?
- 344
  - 644
  - 211
  - 244
  - 311
26. Which command sequence correctly compares the two numbers and prints OK?
- `if [ 4 -ge 3 ] ; then echo OK ; fi`
  - `if ( 3 < 4 ) ; then echo OK ; fi`
  - `if ( ! 4 < 3 ) ; then echo OK ; fi`
  - `if [ ! 4 -gt 3 ] ; then echo OK ; fi`
  - `if [ 4 > 3 ] ; then echo OK ; fi`
27. The **cron** system can run commands at most every
- hour
  - minute
  - millisecond
  - day
  - second
28. 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 **700** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **500** on directory **dir** and **500** on file **dir/bar**.
  - Permissions **700** on directory **dir** and **500** on file **dir/bar**.
  - Permissions **600** on directory **dir** and **500** on file **dir/bar**.
  - Permissions **600** on directory **dir** and **300** on file **dir/bar**.
29. To bring a background shell job into the foreground, type:
- fg**
  - `kill %1`
  - `[Ctrl-D]`
  - bg**
  - `[Ctrl-Z]`
30. 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`
- No, because **bird** has no write permission on the directory
  - No, because execute permissions are not set for **bird** on **foo**
  - No, because the directory is not accessible to **bird**
  - No, because **bird** has no write permissions on **foo**
  - Yes; permissions don't apply because **bird** owns **foo**

31. 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`
- Yes, because **bird** has write permissions on **foobar**
  - No, because execute permissions are not set for **bird** on **foobar**
  - No, because **sesame** has no write permissions on **foobar**
  - Yes, because **bird** owns **foobar**
  - No, because the directory is not accessible to **bird**
32. 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`
- bob** can list names in the directory
  - 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
33. To send a **KILL** signal to a process with process ID **PID**, which of the following commands would you use?
- `kill -KILL PID`
  - `signal -KILL PID`
  - `kill PID KILL`
  - `send -KILL PID`
  - `send PID KILL`
34. 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 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
  - pat** can rename the file
35. What would the following command do: `at 2pm`
- read commands from stdin to be run every day at 2pm
  - read commands from stdin to be run once at 2pm
  - issue an error message
  - run the user's **crontab** jobs every day at 2pm
  - run the user's **crontab** jobs at 2pm
36. If a shell script **myscript.sh** is called this way:  
`./myscript.sh a b c`  
and the first line inside the script below the script header is  
`shift ; echo "$#$1"`  
what is the output of that line?
- 2a
  - 3b
  - 3a
  - 2b
  - 4c

37. Which command counts the number of Unix permission groups you are in?
- `umask | wc`
  - `id | wc`
  - `echo groups | wc`
  - `groups | wc`
  - `wc groups`
38. 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 *[Hh][Ii]*`
  - `echo *[hiHI]*`
  - `echo *(H,h,I,i)*`
  - `echo *[HhIi]*`
  - `echo *[HhIiHhIi]*`
39. 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 create a new file in the directory
  - bob** can access and write on the file
  - bob** can rename the file
  - pat** can create a new file in the directory
  - bob** can list names in the directory
40. 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
```
- bob** can rename the file
  - pat** can create a new file in the directory
  - bob** can list names in the directory
  - pat** can rename the file
  - bob** can access and write on the file
41. 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 **500** on directory **dir** and **500** on file **dir/f**.
  - Permissions **300** on directory **dir** and **500** on file **dir/f**.
  - Permissions **700** on directory **dir** and **200** on file **dir/f**.
  - Permissions **600** on directory **dir** and **300** on file **dir/f**.
  - Permissions **600** on directory **dir** and **500** on file **dir/f**.
42. 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
```
- pat** can access and write on the file
  - 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

43. The **minimum** permissions you need to read a file **foo** in directory **a** are:
- rx** on **a**, none on **foo**
  - wx** on **a**, **w** on **foo**
  - rx** on **a**, **rw** on **foo**
  - wx** on **a**, none on **foo**
  - x** on **a**, **r** on **foo**
44. Inside a shell script, which correctly expands to be the first script argument without processing any special characters in the argument?
- `$1`
  - `"$1"`
  - `'$1'`
  - `"$1"`
  - `\$1`
45. 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`
  - `./1/2/b/c`
  - `./1/a/b/c`
  - `./1/2/a/b/c`
  - `./1/2/c`
46. If you have a file **crontab.day** of commands in **crontab** format, you could submit that file to be your live **crontab** file by running which of the following commands?
- `crontab < crontab.day`
  - `echo crontab.day | crond`
  - `crontab -l crontab.day`
  - `crontab > crontab.day`
  - `crontab -e crontab.day`
47. 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 **400** on directory **dir** and **400** 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**.
  - Permissions **100** on directory **dir** and **200** on file **dir/f**.
  - Permissions **600** on directory **dir** and **700** on file **dir/f**.
48. The **minimum** permissions you need to link a file **foo** from directory **a** to directory **b** are:
- x** on **a**, **wx** on **b**, none on **foo**
  - wx** on **a**, **wx** on **b**, **r** on **foo**
  - wx** on **a**, **wx** on **b**, **w** on **foo**
  - rx** on **a**, **wx** on **b**, none on **foo**
  - rx** on **a**, **wx** on **b**, **rw** on **foo**
49. When an **at** job runs, the current working directory is set to:
- the directory with the name **/home**
  - the system **ROOT** directory
  - the current directory that was in use when the **at** job was created
  - the directory with the name **/root**
  - the **HOME** directory of the user who created the job

50. 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 **200** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **500** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **600** on directory **dir** and **700** on file **dir/bar**.
  - Permissions **400** on directory **dir** and **400** on file **dir/bar**.
51. If **guru=linus** then which one of the following **case** patterns will match this statement: **case "\$guru" in**
- (\*nus echo yes ;;**
  - "linu?" ) echo yes ;;**
  - \* ) echo yes ;;**
  - lin? ) echo yes ;;**
  - [linus] | [LINUS] ) echo yes ;;**
52. What is the output (if any) of this program fragment? (There are blanks between all the digits in the word list section of the **for** loop.)
- ```
s=0
for i in 1 2 3 4
do
    s=$((s+i))
done
echo "$s"
```
- 1 2 3 4
  - 1234
  - 1
  - 10
  - 4321
53. 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
```
- bob** can list names in the directory
  - pat** can access and write on the file
  - bob** can create a new file in the directory
  - pat** can rename the file
  - bob** can access and write on the file
54. 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 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
  - pat** can access and write on the file

55. 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
```
- 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
56. 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
```
- pat** can rename the file
  - pat** can access and write on 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
57. What is the output on your screen of the following sequence of commands:
- ```
x=pig ; [ -z $x ] ; echo $?
```
- 1
  - the number 0 or 1 followed by another 0 or 1 on a new line
  - 0
  - no output
  - test: \$x: integer expression expected**
58. 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**
  - rwx** on **a**, **wx** on **b**, none on **foo**
  - rx** on **a**, **wx** on **b**, **w** on **foo**
59. The **minimum** permissions you need to append to a file **foo** in directory **a** are:
- wx** on **a**, none on **foo**
  - rw** on **a**, none on **foo**
  - rw** on **a**, **rw** on **foo**
  - wx** on **a**, **w** on **foo**
  - x** on **a**, **w** on **foo**
60. 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
```
- pat** can rename the file
  - pat** can create a new file in the directory
  - bob** can rename the file
  - bob** can list names in the directory
  - bob** can access and write on the file

61. 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`
- 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
62. Which command line would show the inode number of a file?
- `ls -l file`
  - `cat -i file`
  - `ls -i file`
  - `find -i file`
  - `cat -l file`
63. 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`
- Yes, because **bird** owns **foobar**
  - No, because execute permissions are not set for **bird** on **foobar**
  - No, because the directory is not accessible to **bird**
  - Yes, because **bird** has write permissions on **foobar**
  - Yes, because **sesame** has write permissions on **foobar**
64. 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 /a/foo /b/new`
  - `ln -s /b/new /a/foo`
  - `ln -s /a/foo /b/new`
  - `ln /a/new /b/foo`
  - `ln /b/new /a/foo`
65. What command displays the kernel ring buffer of log messages:
- `ps lxww`
  - `dmesg`
  - `crontab`
  - `psmine`
  - `showall`
66. The signal sent to a foreground process by typing the **[Ctrl-C]** key is:
- SIGHUP**
  - SIGTERM**
  - SIGKILL**
  - SIGINT**
  - SIGSTOP**
67. 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:
- foo**
  - root and foo**
  - root and bob**
  - bob**
  - root**
68. Can three different files have the same inode number on three different file systems?
- yes: inode numbers are only unique inside a file system
  - no: inode numbers are unique across all file systems
  - no: you can't have inode numbers on three file systems
  - no: inode numbers only apply to directories, not files
  - yes: if the files are all names for the same inode

69. Process signals in increasing order of strength:
- HUP TERM KILL**
  - HUP KILL TERM**
  - TERM KILL HUP**
  - KILL HUP TERM**
  - TERM HUP KILL**
70. A **crontab** entry of `0 6 * * * /sbin/somescript` would run **somescript** when and how often?
- at 12:06am every day
  - at 12:06am every business day and Saturday
  - at 6:00am every day
  - at 6:00am every business day
  - at 12:06am every business day
71. What value **umask** gives a new file permissions **r--r-----**?
- 110
  - 446
  - 337
  - 440
  - 220
72. 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 create a new file in the directory
  - pat** can rename the file
  - pat** can access and write on the file
  - bob** can access and write on the file
  - bob** can list names in the directory
73. Under what directory are system configuration files usually stored?
- /usr/bin**
  - /bin/**
  - /log/var/**
  - /etc**
  - /var/log/**
74. Which of these statements is true?
- 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 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 can only remove a file name if the file is writable by you
75. 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`
- pat** 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
  - bob** can access and write on the file

76. 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`  
`-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
77. A Unix/Linux "tarball" is:
- a single compressed file containing one uncompressed file
  - a multi-file directory containing individual uncompressed files
  - a single-file that contains individual compressed files
  - a multi-file directory containing individual compressed files
  - a single-file that contains individual uncompressed files
78. 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 **200** on directory **dir** and **200** 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 **600** on directory **dir** and **700** on file **dir/c**.
  - Permissions **400** on directory **dir** and **400** on file **dir/c**.
79. 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`
- Yes; permissions don't apply because **bird** owns **foo**
  - No, because execute permissions are not set for **bird** on **foo**
  - Yes, because **bird** has write permissions on **foo**
  - No, because **bird** has no write permission on the directory
  - No, because the directory is not accessible to **bird**
80. 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 create a new file in the directory
  - pat** can rename the file
  - pat** can access and write on the file
  - bob** can access and write on the file
  - bob** can list names in the directory

81. If the line, `exit 2` is executed in a shell script, what is the result?
- the script breaks out of up to 2 levels of loops
  - an invalid argument error message
  - termination with an exit status of 0
  - termination with an exit status of 2
  - termination after sleeping for 2 seconds
82. If a script named **bar** contains a loop that starts: `for i do` and the script is executed using this command line:  
`./bar a ' b d ' e f " g h " a`  
 how many times will the loop iterate?
- 6 iterations
  - 1 iteration
  - 9 iterations
  - 7 iterations
  - 8 iterations
83. 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`
- pat** can access and write on the file
  - bob** 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
84. 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`
- pat** can create a new file in the directory
  - 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
85. What command changes a user's password?
- `chpasswd`
  - `passwd`
  - `chsh`
  - `password`
  - `mkpasswd`
86. Dereference the following symlink **bar** into its equivalent absolute path:  
`ln -s ../b/../../a/./foo /tmp/a/b/bar`
- `/tmp/foo`
  - `/tmp/b/foo`
  - `/tmp/b/bar`
  - `/tmp/a/foo`
  - `/tmp/a/b/bar`

87. Other than **root**, who can change the permissions of the following directory?  
**dr-xrwxrwx 17 foo bar 4096 Apr 15 16:40 .**
- anyone except user **foo**
  - user **foo** and any user in group **bar**
  - only user **foo**
  - only **root** can change the permissions
  - only users in group **bar**
88. 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**
  - rxw** on **a**, **wx** on **b**, **rw** on **foo**
  - rxw** on **a**, **wx** on **b**, none on **foo**
  - wx** on **a**, **wx** on **b**, **r** on **foo**
89. How does system logging work under Unix/Linux?
- processes write log entries directly into the system log directory
  - processes send messages to a central **rsyslog** program that writes log files
  - processes send messages to the **init** process that inherits orphan processes
  - processes write log files into each user's **\$HOME** directory
  - processes copy logs from your **\$HOME** directory to the **/var/spool** directory
90. 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** and **foo**
  - bob**
  - root**
  - root** and **bob**
  - foo**
91. 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**
- bob** can list names in the directory
  - pat** can rename the file
  - bob** can create a new file in the directory
  - pat** can access and write on the file
  - bob** can access and write on the file
92. 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**
- 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 rename the file
  - pat** can access and write on the file

93. Which of the following could you use as options for the **tar** command to extract a **gzip**-compressed archive?
- czf**
  - ezf**
  - tgz**
  - egf**
  - xzf**
94. What command displays the groups you are in?
- gpasswd**
  - lstgroups**
  - groups**
  - mkgroups**
  - groupprint**
95. What value **umask** gives a new file permissions **r--r-----**?
- 220**
  - 440**
  - 110**
  - 446**
  - 237**
96. Inside a shell script, which expands to the name of the script itself?
- "\$@"**
  - "\$#"**
  - "\$?"**
  - "\$\*"**
  - "\$0"**
97. Which command line below does not show any lines from inside the file **out**?
- sort out**
  - wc out**
  - tail out**
  - more out**
  - head out**
98. In a directory containing one file named **dog**, what is the output on your screen after this command line: **1>/dev/null ls \***
- dog**
  - \***
  - bash: 1>/dev/null: command not found**
  - ls: \*: No such file or directory**
  - no output
99. Which command removes adjacent duplicate lines from a file?
- dup**
  - duplicate**
  - dupl**
  - unique**
  - uniq**
100. What value to **chmod** would change the permissions on a file to **r-----rw-**?
- 406**
  - 654**
  - 122**
  - 322**
  - 102**
101. What would be the output of the following command line:  
**echo a b c d | awk '{print \$NF}'**
- \$NF**
  - no output
  - a b c d**
  - 4**
  - d**
102. 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 list names in the directory
  - pat** can create a new file in the directory
  - pat** can rename the file
  - bob** can rename the file
  - bob** can access and write on the file
103. What minimal permissions must you have on a directory to be able to execute successfully the command **ls .** from *inside* the directory?
- wx**
  - r-x**
  - x**
  - r--**
  - rw-**

104. Which command sequence correctly searches for the **string** and then prints **OK** if it is found inside the password file?
- `if fgrep string /etc/passwd ; then echo OK ; fi`
  - `if [ fgrep string /etc/passwd ] ; then echo OK ; fi`
  - `if [ test string /etc/passwd ] ; then echo OK ; fi`
  - `if test string /etc/passwd ; then echo OK ; fi`
  - `if test string = /etc/passwd ; then echo OK ; fi`
105. If variable **a** might contain nothing (a null value - defined but empty), which command sequence correctly tests for this and prints the date?
- `if [ '' = $a ] ; then date ; fi`
  - `if test "" -eq $a ; then date ; fi`
  - `if [ $a = /dev/null ] ; then date ; fi`
  - `if test "" = "$a" ; then date ; fi`
  - `if [ "$a" = * ] ; then date ; fi`
106. 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
```
- 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
  - pat** can create a new file in the directory
107. Which command line makes 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 ; chmod 222 dir`
  - `mkdir dir ; cd dir ; chmod ugo=w .`
  - `mkdir dir ; chmod 333 dir`
  - `mkdir dir ; cd dir ; chmod ugo=rw .`
108. 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 **600** on directory **dir** and **700** on file **dir/foo**.
  - Permissions **300** on directory **dir** and **200** on file **dir/foo**.
  - Permissions **100** on directory **dir** and **100** 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**.
109. Which of the following would result in a "true" exit status?
- `[ '00' -ne "0" ]`
  - `[ '00' = "0" ]`
  - `[ '00' -eq "0" ]`
  - `[ '00' != "00" ]`
  - `[ 00 = 0 ]`

110. The shadow password file is used:
- 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 hide encrypted passwords from viewing by ordinary users
  - to allow passwords to exist on partitions other than the **ROOT**
  - to reduce the size of the main password file for faster access
111. 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 **300** on file **dir/bar**.
  - Permissions **100** on directory **dir** and **500** on file **dir/bar**.
  - Permissions **300** on directory **dir** and **200** on file **dir/bar**.
  - Permissions **300** on directory **dir** and **400** on file **dir/bar**.
  - Permissions **500** on directory **dir** and **500** on file **dir/bar**.
112. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
- ```
d--rwx--x 2 bob ted 60 Jan 1 1:00 foo
----rw--w- 1 bob bg1 0 Jan 1 1:00 foo/bar
```
- bob** can list names in the directory
  - 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
113. 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 .`
  - `ls /home/foo/bar/.`
  - `ls ..`
  - `ls /home/foo/bar/..`
  - `ls /home/foo/bar`
114. 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
```
- bob** can rename the file
  - bob** can list names in the directory
  - bob** can access and write on the file
  - pat** can create a new file in the directory
  - pat** can access and write on the file

115. What is the output on your screen of the following command sequence:  
`i=04; test $i = 4 ; echo $?`
- 0
  - no output
  - the number 0 or 1 followed by another 0 or 1 on a new line
  - 1
  - `test: $i: integer expression expected`
116. What permissions are given to `newfile` after this command line:  
`umask 326 ; touch newfile`
- `r--r-----`
  - `-wx-w-r-x`
  - `r--r-x--x`
  - `-wx-w-rw-`
  - `-wxr-----`
117. The *difference* between the system (`root`) crontab and all the user (personal) crontabs is:
- the personal crontab only runs commands once
  - the personal crontab also has the userid in it
  - the system crontab also has the userid in it
  - the personal crontab has the date and time in it
  - the system crontab has the date and time in it
118. User `bob` is in groups `bg1` and `bg2`. User `pat` is in group `pgg`.  
`d--x--xrw- 2 bob pgg 60 Jan 1 1:00 foo`  
`-r-xrwx-w- 1 bob bg2 0 Jan 1 1:00 foo/bar`
- `bob` can access and write on the file
  - `pat` can rename the file
  - `bob` can create a new file in the directory
  - `bob` can list names in the directory
  - `pat` can access and write on the file
119. Inside a shell script, which expands to the number of script arguments?
- `"$#"`
  - `"$*"`
  - `"$?"`
  - `"$@"`
  - `"$0"`
120. 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 list names in the directory
  - `bob` can rename the file
  - `bob` can create a new file in the directory

121. 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 `100` on directory `dir` and `100` 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`.
122. What command line shows only your own processes, not all processes?
- `psmine`
  - `crontab`
  - `ps lxww`
  - `dmesg`
  - `showall`
123. What command manipulates your personal list of repeated scheduled commands:
- `dmesg`
  - `crontab`
  - `showall`
  - `ps lxww`
  - `psmine`
124. To change your own account password, use this exact command line:
- `$ passwd`
  - `$ passwd *`
  - `$ passwd cst8207`
  - `$ passwd .`
  - `$ passwd idallen-ubuntu`
125. If `browser=lynx` then which one of the following `case` patterns will match this statement: `case "$browser" in`
- `(*ynx echo yes ;;`
  - `l?n? ) echo yes ;;`
  - `?lynx? ) echo yes ;;`
  - `[lynx] | [LYNX] ) echo yes ;;`
  - `@ ) echo yes ;;`
126. 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`
- `./1/2/3/a/b/c`
  - `./1/2/3/a/b`
  - `./1/2/a/b/c`
  - `./1/2/a`
  - `./1/a`
127. 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`
- No, because `foo` has no read permissions for `bird`
  - No, because the directory has no write permissions for `bird`
  - Yes, because `bird` has write permissions on `foo`
  - No, because the directory is not readable by `bird`
  - Yes; permissions don't apply because `bird` owns `foo`

128. 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/a/b`      b. `./1/2/a/b`      c. `./a/b/c`  
d. `./1/2/3/c`      e. `./1/2/3/b/c`
129. What would be the output of the following command line:  
`echo a b c d | awk '{print $2}'`
- a. `b`      b. `c d`      c. no output  
d. `a b`      e. `$2`
130. What is the output of this command line in an empty directory:  
`touch .a .b .c ; echo [.]*`
- a. an error message from `echo` saying `[.]*` does not exist  
b. `... .a .b .c`  
c. `[.]*`  
d. no output  
e. `.a .b .c`
131. If `a=123` and `b=456` then what is the output of the following sequence of commands: `if $a = $b ; then echo $a ; fi`
- a. no output  
b. `test: a=123: integer expression expected`  
c. `test: $a: string expression expected`  
d. `123`  
e. `bash: 123: command not found`
132. Which of these statements is true?
- a. To make a hard link to file "`foo`" named "`bar`", file "`foo`" must exist.  
b. You only need "`r--`" permission on directory "`foo`" for "`ls -l foo`" to work.  
c. The "`ln`" command takes two arguments, so the maximum number of hard links a file can have is two.  
d. You can make a hard link to a directory.  
e. If you give me write permission on a file owned by you, I can then use `chmod` to change its permissions.
133. 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 */.`
- a. an error message because `*/.` does not exist  
b. no output  
c. `*/.`  
d. 5 directory names  
e. 15 pathnames
134. To show all your one-time scheduled commands, type:
- a. `/var/log/crontab`      b. `atq`  
c. `cat crontab`      d. `crontab -l`  
e. `/etc/crontab`

135. Which `crontab` line executes at `13:54` every day?
- a. `13 54 * * * command`      b. `54 13 * * * command`  
c. `13 * * * 54 command`      d. `* * * 54 13 command`  
e. `* * * 13 54 command`
136. When a personal `crontab` job runs, the current working directory is set to:
- a. the system `ROOT` directory  
b. the current directory that was in use when the `crontab` job was created  
c. the `HOME` directory of the user who created the job  
d. the directory with the name `/root`  
e. the directory with the name `/home`
137. What permissions are given to `newdir` after this command line:  
`umask 156 ; mkdir newdir`
- a. `rw--w---x`      b. `rw--w----`      c. `r-x--x---`  
d. `--xr-xrw-`      e. `r-x-w-rw-`
138. 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`
- a. `bob` can create a new file in the directory  
b. `bob` can list names in the directory  
c. `bob` can access and write on the file  
d. `pat` can rename the file  
e. `pat` can access and write on the file
139. 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?
- a. Permissions `500` on directory `dir` and `400` on file `dir/c`.  
b. Permissions `100` on directory `dir` and `100` on file `dir/c`.  
c. Permissions `100` on directory `dir` and `200` on file `dir/c`.  
d. Permissions `300` on directory `dir` and `300` on file `dir/c`.  
e. Permissions `300` on directory `dir` and `500` on file `dir/c`.
140. 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`
- a. `bob` can create a new file in the directory  
b. `bob` can list names in the directory  
c. `bob` can access and write on the file  
d. `pat` can access and write on the file  
e. `pat` can rename the file
141. Which command line below does not show any lines from inside the file `bat`?
- a. `head bat`      b. `ls bat`      c. `less bat`  
d. `more bat`      e. `tail bat`

142. 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
  - bob** can create a new file in the directory
  - bob** can list names in the directory
  - pat** can rename the file
143. Which of the following commands would result in an error?
- `[ 3 -e 3 ]`
  - `[ a != 4 ]`
  - `[ a = 4 ]`
  - `[ 3 -eq 4 ]`
  - `[ 3 = f ]`
144. 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`
- No, because **foo** has no write permissions for **bird**
  - Yes; permissions don't apply because **bird** owns **foo**
  - No, because the directory has no write permissions for **bird**
  - No, because the directory is not accessible to **bird**
  - Yes, because **bird** has read permissions on **foo**
145. 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`
- 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
  - pat** can rename the file
146. The output of the **whoami** command is:
- your userid
  - your HOME directory
  - a list of accounts in the password file
  - a list of users logged in to the system
  - the current directory
147. If **archive.tar.gz** is a compressed tar archive, which command could you run to produce a listing of its contents without extracting it?
- `tar -tgz archive.tar.gz`
  - `tar -tgz archive`
  - `tar -tzf archive`
  - `tar -xzf archive.tar.gz`
  - `tar -tzf archive.tar.gz`

148. 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
  - 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
149. What is the output on your screen of the following sequence of commands:  
`a=4 ; b=4 ; [ $a -le $b ] ; echo $?`
- `test: $a: integer expression expected`
  - 1
  - 0
  - the number 1 or 0 followed by another 1 or 0 on a new line
  - no output
150. Given the following, can user **bird** in group **sesame** rename `./foo` to `bar`?  
`d----wx--- 2 root sesame 4096 Oct 7 14:00 .`  
`----- 1 bird sesame 123 Oct 4 14:05 foo`
- No, because **bird** has no permissions on **foo**
  - No, because **bird** cannot read the directory
  - Yes; permissions don't apply because **bird** owns **foo**
  - No, because the directory has no permissions for other users
  - Yes, because **bird**'s group matches the group writable directory
151. If I mount **sda1** on **/one** and **sda2** on **/two**, how can I link the existing file **/one/foo** to the new pathname **/two/bar**?
- `ln -s /one/foo /two/bar`
  - `ln /one/foo /two/bar`
  - `ln -s /two/bar /one/foo`
  - `ln /two/bar /one/foo`
  - `ln /one/bar /two/foo`
152. In an empty directory, what permissions are on file **???** after these commands:  
`touch ??? *** ; chmod 111 *`  
`chmod 222 ? ; chmod 444 '*'`
- `--x--x--x`
  - `rw-rw-rw-`
  - `-wx-wx-wx`
  - `-w--w--w-`
  - `r--r--r--`
153. Dereference the following symlink **bar** into its equivalent absolute path:  
`ln -s ../b/./b/./b/./b/./foo /tmp/a/b/bar`
- `/tmp/b/bar`
  - `/tmp/b/foo`
  - `/tmp/a/b/bar`
  - `/tmp/foo`
  - `/tmp/a/foo`
154. Which expands to the exit status of the previous command?
- `"$?"`
  - `"$@"`
  - `"$#"`
  - `"$*"`
  - `"$0"`

155. 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`
- bob** can list names 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 create a new file in the directory
156. 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`
- pat** can rename the file
  - bob** 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
157. Which of these commands makes a file owned by me, also readable by me?
- `umask 400 myfile`
  - `chmod r=u ./myfile`
  - `chmod u+r ./myfile`
  - `chmod r+u myfile`
  - `umask 300 ./myfile`
158. A shell script named **bar** is executed as follows:  
`./bar "a b" "c d e" f`  
 Inside the script is the line: `echo "$3"`  
 What is the output on your screen from this line?
- `f`
  - `"f"`
  - `$3`
  - `c d e`
  - `a b`
159. Given the following shell script statement,  
`if [ "a" = "b" ] ; then echo SAME ; fi`  
 which of the following statements is true?
- an "invalid number" error would result
  - SAME** would be printed
  - fi** would cause a "command not found" error
  - `[` is passed four arguments
  - `[` is part of all **if** statements
160. 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 access and write on the file
  - bob** can access and write on the file
  - bob** can create a new file in the directory
  - pat** can rename the file

161. 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 **300** on directory **dir** and **200** on file **dir/bar**.
  - 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 **100** on directory **dir** and **100** on file **dir/bar**.
162. In a directory containing one file named **dog**, what is the output on your screen after this command line: `2>/dev/null ls nosuchfile`
- `bash: 2>/dev/null: command not found`
  - `dog`
  - `nosuchfile`
  - `ls: nosuchfile: No such file or directory`
  - no output
163. 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`
- `./1/2/a/b`
  - `./1/2/b/c`
  - `./1/2/c`
  - `./a/b/c`
  - `./1/a/b`
164. What command terminates processes based on their name (not safe!):
- `crontab`
  - `kill`
  - `dmesg`
  - `killall`
  - `ps lxww`
165. What is the output on your screen of the following command sequence:  
`a=1 ; b=2 ; test $b -ge $a ; echo $?`
- `test: $b: integer expression expected`
  - `0`
  - the number 1 or 0 followed by another 1 or 0 on a new line
  - `1`
  - no output on screen
166. In a shell **case** structure, the **case** segment that will GLOB match the text **a, b**, or **c**, is coded as
- `a|b|c )`
  - `a/b/c )`
  - `a\b\c )`
  - `a:b:c )`
  - `a,b,c )`
167. The password **:x:** in `/etc/passwd` means:
- the encrypted password is stored in the shadow file
  - the account is locked
  - the unencrypted password is stored in the group file
  - the encrypted password is **"x"**
  - the password is locked

168. The **minimum** permissions you need to delete a file **foo** from directory **a** are:
- wx** on **a**, **w** on **foo**
  - wx** on **a**, **r** on **foo**
  - wx** on **a**, none on **foo**
  - rw** on **a**, **rw** on **foo**
  - rw** on **a**, none on **foo**
169. What value **umask** gives a new file permissions **r--r-----**?
- 440**
  - 226**
  - 446**
  - 220**
  - 110**
170. Which expands to all the script arguments?
- "\$!"**
  - "\$#"**
  - "\$0"**
  - "\$?"**
  - "\$\*"**
171. If a shell script named **foo** contains the line:
- ```
if [ '$3' = "$1" ] ; then echo SAME ; fi
```
- then which of the following command lines will produce **SAME** as output?
- `./foo "$1" '$3'`
  - `./foo '$3' bar`
  - `./foo $3 $3`
  - `./foo "bar" 'bar'`
  - `./foo bar bar`
172. Under what directory are system log files usually stored?
- `/var/log`
  - `/log/var`
  - `/usr/bin`
  - `/etc/log`
  - `/bin/`
173. 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?
- `ls ..`
  - `ls /tmp/dir/..`
  - `ls .`
  - `ls /tmp/dir`
  - `ls /tmp/dir/.`
174. In an empty directory, what permissions are on file **???** after these commands:
- ```
touch ??? *** ; chmod 111 *
chmod 222 ??? ; chmod 444 '***'
```
- `--x--x--x`
  - `-wx-wx-wx`
  - `r--r--r--`
  - `-w--w--w-`
  - `rw-rw-rw-`
175. What value **umask** gives a new file permissions **r--r-----**?
- 447**
  - 326**
  - 220**
  - 110**
  - 440**
176. If **a=123** and **b=456** then what is the output of the following sequence of commands: `if [ $a = $b ] ; then echo $a ; fi`
- test: \$a: string expression expected**
  - bash: 123: command not found**
  - no output
  - 123**
  - test: a=123: integer expression expected**

177. If **bar** is an executable script containing the line **animal=dog** then what is the **bash** output of this sequence of three commands:
- ```
animal=pig ; ./bar ; echo "the '$animal' ate"
```
- the 'dog' ate
  - the 'animal' ate
  - the 'pig' ate
  - the '\$animal' ate
  - the \$animal ate
178. Which of the following signals is strongest (cannot be handled or ignored)?
- SIGINT**
  - SIGHUP**
  - SIGTERM**
  - SIGSUSP**
  - SIGKILL**
179. What is the output on your screen of the following sequence of commands:
- ```
i=00 ; [ $i -eq 0 ] ; echo $?
```
- the number 0 or 1 followed by another 0 or 1 on a new line
  - 0**
  - no output
  - test: \$i: integer expression expected**
  - 1**
180. **Did you read all the words of the test instructions on page one?**
- 132**
  - 231**
  - 123**
  - 312**
  - 321**