PRINT Name: _____

Test Version: `169` One-Answer Multiple Choice 203 Questions − 40 of 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.
☞ 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.
☞ The answer to the last question about reading/doing all these test instructions is: **`Jes`**

---

1. Which command counts the number of Unix permission groups you are in?
   a. **`wc groups`** *b.* **`umask | wc`**
   c. **`id | wc`** *d.* **`echo groups | wc`**
   e. **`groups | wc`**
2. Which of these statements is true?
   a. The "**`ln`**" command takes two arguments, so the maximum number of hard links a file can have is two.
   b. If you give me write permission on a file owned by you, I can then use **`chmod`** to change its permissions.
   c. You can make a hard link to a directory.
   d. To make a hard link to file "**`foo`**" named "**`bar`**", file "**`foo`**" must exist.
   e. You only need "**`r--`**" permission on directory "**`foo`**" for "**`ls -l foo`**" to work.
3. If the current directory contains files **`abc`**, **`bbc`**, **`cbc`**, and **`bbc`** contains just the line **`bbb`**, what is the output of the following command: **`grep bb* bbc`**
   a. **`dbd`** *b.* **`bbc`** *c.* an error message
   d. **`bbb`** *e.* no output
4. 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?
   a. Permissions **`700`** on directory **`dir`** and **`200`** on file **`dir/f`**.
   b. Permissions **`600`** on directory **`dir`** and **`300`** on file **`dir/f`**.
   c. Permissions **`300`** on directory **`dir`** and **`500`** on file **`dir/f`**.
   d. Permissions **`500`** on directory **`dir`** and **`500`** on file **`dir/f`**.
   e. Permissions **`600`** on directory **`dir`** and **`500`** on file **`dir/f`**.
5. Which command line displays all the non-hidden names in the current directory that contain the case-insensitive word **`hi`** (and no other names)?
   a. **`echo *[Hh][Ii]*`** *b.* **`echo ?[HhIiHhIi]?`**
   c. **`echo ?[HhIi]?`** *d.* **`echo *(H,h,I,i)*`**
   e. **`echo *[hiHI]*`**

6. 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
```
   a. **`bob`** can access and write on the file
   b. **`bob`** can rename the file
   c. **`bob`** can list names in the directory
   d. **`pat`** can access and write on the file
   e. **`bob`** can create a new file in the directory
7. 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. **`test: a=123: integer expression expected`**
   b. no output
   c. **`bash: 123: command not found`**
   d. **`test: $a: string expression expected`**
   e. **`123`**
8. What permissions are given to **`newfile`** after this command line:
```
umask 326 ; touch newfile
```
   a. **`-wx-w-rw-`** *b.* **`-wx-w-r-x`** *c.* **`r--r-x--x`**
   d. **`r--r-----`** *e.* **`-wxr-----`**
9. Under what directory are system log files usually stored?
   a. **`/var/log`** *b.* **`/usr/bin`** *c.* **`/log/var`**
   d. **`/bin/`** *e.* **`/etc/log`**
10. In an empty directory, what permissions are on file **`???`** after these commands:
```
touch ??? *** ; chmod 111 *
chmod 222 ??? ; chmod 444 '***'
```
   a. **`rw-rw-rw-`** *b.* **`-w--w--w-`** *c.* **`-wx-wx-wx`**
   d. **`--x--x--x`** *e.* **`r--r--r--`**
11. 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
```
   a. **`pat`** can access and write on the file
   b. **`bob`** can list names in the directory
   c. **`bob`** can access and write on the file
   d. **`pat`** can rename the file
   e. **`bob`** can create a new file in the directory

12. 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
```
a. **bob** can list names in the directory
b. **pat** can rename the file
c. **bob** can access and write on the file
d. **bob** can create a new file in the directory
e. **bob** can rename the file

13. What permissions are given to **newdir** after this command line:
```
umask 156 ; mkdir newdir
```
a. **r-x-w-rw-** *b.* **rw--w----** *c.* **rw--w---x**
d. **--xr-xrw-** *e.* **r-x--x---**

14. Process signals in increasing order of strength:
a. **KILL HUP TERM** *b.* **HUP TERM KILL**
c. **TERM KILL HUP** *d.* **TERM HUP KILL**
e. **HUP KILL TERM**

15. 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
```
a. **bob** can create a new file in the directory
b. **pat** can access and write on the file
c. **pat** can rename the file
d. **bob** can access and write on the file
e. **bob** can list names in the directory

16. 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
```
a. **./1/2/3/a/b** *b.* **./1/2/3/a/b/c**
c. **./1/a** *d.* **./1/2/a**
e. **./1/2/a/b/c**

17. 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. **pat** can access and write on the file
b. **bob** can access and write on the file
c. **bob** can list names in the directory
d. **bob** can create a new file in the directory
e. **pat** can rename the file

18. Dereference the following symlink **bar** into its equivalent absolute path:
```
ln -s ../b/../../a/../foo /tmp/a/b/bar
```
a. **/tmp/b/bar** *b.* **/tmp/b/foo** *c.* **/tmp/a/foo**
d. **/tmp/a/b/bar** *e.* **/tmp/foo**

19. Which expands to the exit status of the previous command?
a. **"$0"** *b.* **"$?"** *c.* **"$@"** *d.* **"$*"** *e.* **"$#"**

20. If the file **foo** in the current directory contains just two lines **dbd**, and **123**, what is the output of the following command: **grep '[:alnum:]' foo**
a. **foo** *b.* no output or an error message
c. **123** *d.* **dbd**
e. both lines

21. Which expands to all the script arguments?
a. **"$0"** *b.* **"$!"** *c.* **"$?"** *d.* **"$*"** *e.* **"$#"**

22. 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
```
a. **bob** can rename the file
b. **bob** can access and write on the file
c. **bob** can list names in the directory
d. **bob** can create a new file in the directory
e. **pat** can access and write on the file

23. 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
```
a. **bob** can list names in the directory
b. **bob** can access and write on the file
c. **bob** can rename the file
d. **bob** can create a new file in the directory
e. **pat** can create a new file in the directory

24. If the current directory contains files **abc**, **bbc**, **cbc**, and **bbc** contains just the line **dbd**, what is the output of the following command: **grep "^bb*" bbc**
a. **cbc** *b.* no output *c.* **dbd**
d. **bbc** *e.* an error message

25. Which of the following would result in a "true" exit status?
a. **[00 = 0]** *b.* **['00' = "0"]**
c. **['00' != "00"]** *d.* **['00' -eq "0"]**
e. **['00' -ne "0"]**

26. If the file **foo** in the current directory contains just the line **dbd**, what is the output of the following command: **grep '^[[:alpha:]]' foo**
a. no output *b.* **dbd** *c.* an error message
d. **123** *e.* **foo**

27. Other than **root**, who can change the permissions of the following directory?
```
dr-xrwxrwx 17 foo bar 4096 Apr 15 16:40 .
```
a. only users in group **bar**
b. only user **foo**
c. anyone except user **foo**
d. user **foo** and any user in group **bar**
e. only **root** can change the permissions

28. Which of the following signals is strongest (cannot be handled or ignored)?
a. **SIGHUP** *b.* **SIGTERM** *c.* **SIGSUSP**
d. **SIGINT** *e.* **SIGKILL**

29. Which of the following commands would result in an error?
a. **[a -eq 4]** *b.* **[3 -eq 4]** *c.* **[3 = 4]**
d. **[a = 4]** *e.* **[a != 4]**

30. When a personal **crontab** job runs, the current working directory is set to:
a. the directory with the name **/root**
b. the HOME directory of the user who created the job
c. the system ROOT directory
d. the current directory that was in use when the **crontab** job was created
e. the directory with the name **/home**

31. Given the following shell script statement,
```
if [ "a" = "b" ] ; then echo SAME ; fi
```
which of the following statements is true?
a. an "invalid number" error would result
b. "**[**" is passed four arguments
c. "**SAME**" would be printed
d. "**[**" is part of all "**if**" statements
e. "**fi**" would cause a "command not found" error

32. 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 **foo** has no read permissions for **bird**
b. Yes, because **bird** has write permissions on **foo**
c. Yes; permissions don't apply because **bird** owns **foo**
d. No, because the directory is not readable by **bird**
e. No, because the directory has no write permissions for **bird**

33. Which command line below does not show any lines from inside the file **out**?
a. **sort out** *b.* **tail out** *c.* **head out**
d. **wc out** *e.* **more out**

34. 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"
```
a. **1 2 3 4** *b.* **1** *c.* **1234**
d. **10** *e.* **4321**

35. If the current directory contains files **abc**, **bbc**, **cbc**, and **bbc** contains just the line **dbd**, what is the output of the following command: **grep 'b*$' bbc**
a. no output *b.* **cbc** *c.* an error message
d. **dbd** *e.* **bbc**

36. What value **umask** gives a new file permissions **r--r-----**?
a. **110** *b.* **326** *c.* **447** *d.* **220** *e.* **440**

37. 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. **chmod 333 .**
b. **cd dir ; chmod ugo-rw .**
c. **chmod 222 dir**
d. **chmod 333 dir**
e. **cd dir ; chmod ugo=w .**

38. Which command sequence correctly searches for the **string** and then prints **OK** if it is found inside the password file?
a. **if [test string /etc/passwd] ; then echo OK ; fi**
b. **if test string = /etc/passwd ; then echo OK ; fi**
c. **if test string /etc/passwd ; then echo OK ; fi**
d. **if [grep string /etc/passwd] ; then echo OK ; fi**
e. **if grep string /etc/passwd ; then echo OK ; fi**

39. 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
```
a. **pat** 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. **bob** can rename the file

40. 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 **300** on directory **dir** and **500** on file **dir/bar**.
    c. Permissions **300** on directory **dir** and **300** on file **dir/bar**.
    d. Permissions **500** on directory **dir** and **400** on file **dir/bar**.
    e. Permissions **100** on directory **dir** and **100** on file **dir/bar**.

41. The **minimum** permissions you need to append to a file **foo** in directory **a** are:
    a. **wx** on **a**, **w** on **foo**
    b. **wx** on **a**, none on **foo**
    c. **x** on **a**, **w** on **foo**
    d. **rwx** on **a**, **rw** on **foo**
    e. **rwx** on **a**, none on **foo**

42. The **cron** system can run commands at most every
    a. millisecond
    b. hour
    c. day
    d. second
    e. minute

43. What would the following command do: **at 2pm**
    a. run the user's **crontab** jobs at 2pm
    b. issue an error message
    c. read commands from stdin to be run every day at 2pm
    d. run the user's **crontab** jobs every day at 2pm
    e. read commands from stdin to be run once at 2pm

44. 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. **test: a=123: integer expression expected**
    b. **123**
    c. **bash: [123: command not found**
    d. no output
    e. **test: $a: string expression expected**

45. What is the output on your screen of the following sequence of commands:
    a=4 ; b=4 ; [$a -le $b] ; echo $?
    a. **1**
    b. **0**
    c. the number 1 or 0 followed by another 1 or 0 on a new line
    d. **test: $a: integer expression expected**
    e. no output

46. Which of the following regular expressions would match only lines that contain only one or more alphanumeric characters?
    a. **^[[:alnum:]]*$**
    b. **[[:alnum:]][[:alnum:]]***
    c. **[a-z0-9][a-z0-9]***
    d. **[[:alnum:]]***
    e. **^[[:alnum:]][[:alnum:]]*$**

47. If the file **foo** in the current directory contains just the line **dbd**, what is the output of the following command: **grep '[[:alpha:]]' foo**
    a. **foo**
    b. an error message
    c. **dbd**
    d. no output
    e. **123**

48. Which command line would show the inode number of a file?
    a. **ls -i file**
    b. **find -i file**
    c. **ls -l file**
    d. **cat -l file**
    e. **cat -i file**

49. 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?
    a. **"f"**
    b. **$3**
    c. **f**
    d. **c d e**
    e. **a b**

50. Which of the following regular expressions would match only lines that contain no white space?
    a. **[^:space:]***
    b. **[^[:space:]]*$**
    c. **^[[:space:]]*$**
    d. **^[^[:space:]]*$**
    e. **[^[:space:]]***

51. 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 **500** on directory **dir** and **500** on file **dir/bar**.
    b. Permissions **300** on directory **dir** and **200** on file **dir/bar**.
    c. Permissions **100** on directory **dir** and **500** on file **dir/bar**.
    d. Permissions **300** on directory **dir** and **400** on file **dir/bar**.
    e. Permissions **100** on directory **dir** and **300** on file **dir/bar**.

52. 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?
    a. Permissions **200** on directory **dir** and **200** on file **dir/c**.
    b. Permissions **100** on directory **dir** and **100** on file **dir/c**.
    c. Permissions **600** on directory **dir** and **700** on file **dir/c**.
    d. Permissions **400** on directory **dir** and **400** on file **dir/c**.
    e. Permissions **100** on directory **dir** and **200** on file **dir/c**.

53. To bring a background shell job into the foreground, type:
    a. **[Ctrl-D]**
    b. **bg**
    c. **[Ctrl-Z]**
    d. **fg**
    e. **kill %1**

54. If the current directory contains files **abc**, **bbc**, **cbc**, and **bbc** contains just the line **dbd**, what is the output of the following command: **grep '^bb*' bbc**
    a. **dbd**
    b. no output
    c. an error message
    d. **cbc**
    e. **bbc**

55. What command changes a user's password?
a. **chpasswd** *b.* **password** *c.* **passwd**
d. **mkpasswd** *e.* **chsh**

56. A Unix/Linux "tarball" is:
a. a multi-file directory containing individual compressed files
b. a single compressed file containing one uncompressed file
c. a single-file that contains individual uncompressed files
d. a single-file that contains individual compressed files
e. a multi-file directory containing individual uncompressed files

57. 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**?
a. **ln /a/foo /b/new** *b.* **ln /b/new /a/foo**
c. **ln -s /a/foo /b/new** *d.* **ln -s /b/new /a/foo**
e. **ln /a/new /b/foo**

58. Which of these statements is true?
a. you may be able to rename a file even if you do not own the file
b. you can change the permissions of any file to which you can write
c. you can only remove a file name if the file is owned by you
d. you can only remove a file name if the file is writable by you
e. you can only make links to files owned by you

59. In a directory containing one file named **dog**, what is the output on your screen after this command line: **2>/dev/null ls nosuchfile**
a. no output
b. **ls: nosuchfile: No such file or directory**
c. **bash: 2>/dev/null: command not found**
d. **nosuchfile**
e. **dog**

60. 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?
a. 9 iterations *b.* 8 iterations *c.* 7 iterations
d. 6 iterations *e.* 1 iteration

61. If the file **foo** in the current directory contains just two lines **123** and **abc**, what is the output of the following command: **grep '[[:alpha:]]' foo**
a. no output *b.* **123** *c.* **foo**
d. an error message *e.* **abc**

62. Which of the following regular expressions would match only lines that contain exactly one character of any kind?
a. **^.$** *b.* **^.*$** *c.* **^*$** *d.* **^?$** *e.* **^\?$**

63. 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 rename the file
b. **pat** can access and write on the file
c. **bob** can list names in the directory
d. **bob** can create a new file in the directory
e. **bob** can access and write on the file

64. Which command usually goes in your **.bash_profile** file?
a. **source ./.bashrc** *b.* **cat ./.bashrc**
c. **./.bashrc source** *d.* **source ./.bash_profile**
e. **./.bash_profile source**

65. Which of these commands makes a file owned by me, also readable by me?
a. **umask 400 myfile** *b.* **chmod r+u myfile**
c. **umask 300 ./myfile** *d.* **chmod u+r ./myfile**
e. **chmod r=u ./myfile**

66. Inside a shell script, which expands to the name of the script itself?
a. **"$0"** *b.* **"$@"** *c.* **"$?"** *d.* **"$#"** *e.* **"$*"**

67. 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:
a. **root and foo** *b.* **foo** *c.* **root and bob**
d. **root** *e.* **bob**

68. Can three different files have the same inode number on three different file systems?
a. no: you can't have inode numbers on three file systems
b. no: inode numbers are unique across all file systems
c. yes: inode numbers are only unique inside a file system
d. yes: if the files are all names for the same inode
e. no: inode numbers only apply to directories, not files

69. What value **umask** gives a new file permissions **r--r-----**?
a. **446** *b.* **237** *c.* **110** *d.* **440** *e.* **220**

70. The *difference* between the system (**root**) crontab and all the user (personal) crontabs is:
a. the system crontab also has the userid in it
b. the personal crontab also has the userid in it
c. the personal crontab only runs commands once
d. the personal crontab has the date and time in it
e. the system crontab has the date and time in it

71. 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
```
a. **pat** can access and write on the file
b. **bob** can list names in the directory
c. **bob** can create a new file in the directory
d. **pat** can rename the file
e. **bob** can access and write on the file

72. The **minimum** permissions you need to read a file **foo** in directory **a** are:
a. **rwx** on **a**, none on **foo**
b. **wx** on **a**, **w** on **foo**
c. **x** on **a**, **r** on **foo**
d. **rwx** on **a**, **rw** on **foo**
e. **wx** on **a**, none on **foo**

73. What is the output on your screen of the following sequence of commands:
```
x=ok ; y=ok ; [ x = y ]
```
a. `test: x: integer expression expected`
b. no output on screen
c. **0**
d. **1**
e. `bash: x: command not found`

74. Which command sequence correctly compares the two numbers and prints **OK**?
a. `if ( 3 < 4 ) ; then echo OK ; fi`
b. `if ( ! 4 < 3 ) ; then echo OK ; fi`
c. `if [ ! 4 -gt 3 ] ; then echo OK ; fi`
d. `if [ 4 -ge 3 ] ; then echo OK ; fi`
e. `if [ 4 > 3 ] ; then echo OK ; fi`

75. To list your personal crontab, type:
a. `crontab -l`
b. `/var/log/crontab`
c. `cat crontab`
d. `/etc/crontab`
e. `atq`

76. Inside a shell script, which correctly expands to be the first script argument without processing any special characters in the argument?
a. `$1`
b. `'$1'`
c. `"\$1"`
d. `\$1`
e. `"$1"`

77. If **archive.tar.gz** is a compressed tar archive, which command could you run to produce a listing of its contents without extracting it?
a. `tar -tgz archive.tar.gz`
b. `tar -tzf archive`
c. `tar -tgz archive`
d. `tar -tzf archive.tar.gz`
e. `tar -xzf archive.tar.gz`

78. What is the output on your screen of the following sequence of commands:
```
i=00 ; [ $i -eq 0 ] ; echo $?
```
a. **1**
b. the number 0 or 1 followed by another 0 or 1 on a new line
c. `test: $i: integer expression expected`
d. no output
e. **0**

79. 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 **100** on directory **dir** and **200** on file **dir/c**.
b. Permissions **300** on directory **dir** and **500** on file **dir/c**.
c. Permissions **300** on directory **dir** and **300** on file **dir/c**.
d. Permissions **100** on directory **dir** and **100** on file **dir/c**.
e. Permissions **500** on directory **dir** and **400** on file **dir/c**.

80. If the file **foo** in the current directory contains just the line **123**, what is the output of the following command: `grep '[[:alpha:]]' foo`
a. **123**
b. no output
c. an error message
d. **foo**
e. **dbd**

81. What command would you use to see the command that **at** job number **2** will run?
a. `at -l 2`
b. `atq 2`
c. `at -c 2`
d. `at -m 2`
e. `at -v 2`

82. What would be the output of the following command line:
```
echo a b c d | awk '{print $2}'
```
a. **c d**
b. **$2**
c. no output
d. **b**
e. **a b**

83. What is the output on your screen of the following command sequence:
```
i=04; test $i = 4 ; echo $?
```
a. the number 0 or 1 followed by another 0 or 1 on a new line
b. no output
c. `test: $i: integer expression expected`
d. **0**
e. **1**

84. 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. `./a/b/c`
b. `./1/2/a/b`
c. `./1/2/3/c`
d. `./1/2/3/b/c`
e. `./1/2/3/a/b`

85. If the current directory contains files **abc**, **bbc**, **cbc**, and **bbc** contains just the line **dbd**, what is the output of the following command: `grep 'bb*$' bbc`
a. no output
b. **bbc**
c. **cbc**
d. an error message
e. **dbd**

86. Which of the following regular expressions would match only lines that contain an **A** or a **B** somewhere?
a. **[^AB]** *b.* **^A*B*$** *c.* **[AB]**
d. **^[AB]$** *e.* **[^[AB]]***

87. 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. No, because **bird** cannot read the 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. Yes, because **bird**'s group matches the group writable directory

88. 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
```
a. **bob** can access and write on the file
b. **pat** can rename the file
c. **bob** can list names in the directory
d. **bob** can create a new file in the directory
e. **pat** can access and write on the file

89. 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
```
a. **bob** can access and write on the file
b. **pat** can rename the file
c. **bob** can list names in the directory
d. **bob** can create a new file in the directory
e. **pat** can access and write on the file

90. Which of the following programs uses file GLOBbing expressions rather than regular expressions for matching:
a. **grep** *b.* **egrep** *c.* **vi**
d. **less** *e.* **find**

91. What is the output on your screen of the following command sequence:
```
a=1 ; b=2 ; test $b -ge $a ; echo $?
```
a. no output on screen
b. the number 1 or 0 followed by another 1 or 0 on a new line
c. **test: $b: integer expression expected**
d. **1**
e. **0**

92. 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
```
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

93. 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?
a. Permissions **600** on directory **dir** and **700** on file **dir/bar**.
b. Permissions **300** on directory **dir** and **200** on file **dir/bar**.
c. Permissions **400** on directory **dir** and **400** on file **dir/bar**.
d. Permissions **500** on directory **dir** and **600** on file **dir/bar**.
e. Permissions **100** on directory **dir** and **100** on file **dir/bar**.

94. 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
```
a. **bob** can access and write on the file
b. **pat** can rename the file
c. **bob** can create a new file in the directory
d. **bob** can list names in the directory
e. **pat** can access and write on the file

95. Which command removes adjacent duplicate lines from a file?
a. **dup** *b.* **dupl** *c.* **unique**
d. **uniq** *e.* **duplicate**

96. What minimal permissions must you have on a directory to be able to execute successfully the command **ls .** from *inside* the directory?
a. **-wx** *b.* **--x** *c.* **rw-** *d.* **r-x** *e.* **r--**

97. 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. ***/.**
b. 5 directory names
c. an error message because ***/.** does not exist
d. 15 pathnames
e. no output

98. 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
```
a. **bob** can list names in the directory
b. **pat** can create a new file in the directory
c. **pat** can access and write on the file
d. **bob** can rename the file
e. **bob** can access and write on the file

99. What command manipulates your personal list of repeated scheduled commands:
a. **crontab** *b.* **dmesg** *c.* **showall**
d. **ps lxww** *e.* **psmine**

100. What command line shows only your own processes, not all processes?
a. **ps lxww** *b.* **psmine** *c.* **showall**
d. **dmesg** *e.* **crontab**

101. Which of the following options for **bash** or **sh** might be useful for debugging a shell script?
a. **-z** *b.* **-x** *c.* **-c** *d.* **-l** *e.* **-r**

102. 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/b/c** *c.* **./a/b/c**
d. **./1/2/c** *e.* **./1/a/b**

103. 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. **test: a=123: integer expression expected**
b. **bash: 123: command not found**
c. no output
d. **123**
e. **test: $a: string expression expected**

104. 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 list names in the directory
b. **pat** 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 rename the file

105. 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
```
a. **./1/a/b/c** *b.* **./1/2/3/a/b/c**
c. **./1/2/b/c** *d.* **./1/2/c**
e. **./1/2/a/b/c**

106. 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"
```
a. **the 'animal' ate** *b.* **the 'dog' ate**
c. **the $animal ate** *d.* **the 'pig' ate**
e. **the '$animal' ate**

107. What command displays the kernel ring buffer of log messages:
a. **dmesg** *b.* **showall** *c.* **psmine**
d. **crontab** *e.* **ps lxww**

108. 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
```
a. No, because execute permissions are not set for **bird** on **foobar**
b. Yes, because **sesame** has write permissions on **foobar**
c. Yes, because **bird** owns **foobar**
d. Yes, because **bird** has write permissions on **foobar**
e. No, because the directory is not accessible to **bird**

109. 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
```
a. **bob** can create a new file in the directory
b. **bob** can list names in the directory
c. **pat** can rename the file
d. **pat** can access and write on the file
e. **bob** can access and write on the file

110. In a directory containing one file named **dog**, what is the output on your screen after this command line: **1>/dev/null ls ***
a. *****
b. **dog**
c. **ls: *: No such file or directory**
d. no output
e. **bash: 1>/dev/null: command not found**

111. 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. **bob** can access and write on the file
b. **pat** can access and write on the file
c. **pat** can rename the file
d. **bob** can create a new file in the directory
e. **bob** can list names in the directory

112. What does the **-v** option to the **grep** command do?
   a. prints the version number of the grep command
   b. turns on the translation of unprintable characters
   c. turns off the translation of unprintable characters
   d. selects lines that do not contain unprintable characters
   e. selects lines that do not contain a match for the supplied pattern

113. What value to **chmod** would change the permissions on a file to **r-----rw-**?
   a. **406** *b.* **122** *c.* **654** *d.* **102** *e.* **322**

114. 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 create a new file in the directory
   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 rename the file

115. 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:
   a. **root and bob** *b.* **bob** *c.* **root and foo**
   d. **root** *e.* **foo**

116. 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?
   a. Permissions **100** on directory **dir** and **200** on file **dir/f**.
   b. Permissions **400** on directory **dir** and **400** on file **dir/f**.
   c. Permissions **500** on directory **dir** and **100** on file **dir/f**.
   d. Permissions **200** on directory **dir** and **200** on file **dir/f**.
   e. Permissions **600** on directory **dir** and **700** on file **dir/f**.

117. The **minimum** permissions you need to move a file **foo** from directory **a** to directory **b** are:
   a. **wx** on **a**, **wx** on **b**, **w** on **foo**
   b. **rwx** on **a**, **wx** on **b**, none on **foo**
   c. **wx** on **a**, **wx** on **b**, none on **foo**
   d. **rwx** on **a**, **wx** on **b**, **rw** on **foo**
   e. **wx** on **a**, **wx** on **b**, **r** on **foo**

118. What value **umask** gives a new directory permissions **rw--w---x**?
   a. **156** *b.* **421** *c.* **432** *d.* **621** *e.* **211**

119. How does system logging work under Unix/Linux?
   a. processes send messages to the **init** process that inherits orphan processes
   b. processes write log files into each user's **$HOME** directory
   c. processes write log entries directly into the system log directory
   d. processes send messages to a central **rsyslog** program that writes log files
   e. processes copy logs from your **$HOME** directory to the **/var/spool** directory

120. 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
```
   a. **bob** can access and write on the file
   b. **bob** can list names in the directory
   c. **pat** can access and write on the file
   d. **pat** can rename the file
   e. **bob** can create a new file in the directory

121. What is the output of this command line in an empty directory:
```
touch .a .b .c ; echo [.]*
```
   a. **[.]***
   b. **.a .b .c**
   c. no output
   d. an error message from **echo** saying **[.]*** does not exist
   e. **. .. .a .b .c**

122. The shadow password file is used:
   a. to hide encrypted passwords from viewing by ordinary users
   b. to allow passwords to exist on partitions other than the ROOT
   c. to store secondary passwords for times when you forget your main one
   d. to keep a back-up of the main password file in case of corruption
   e. to reduce the size of the main password file for faster access

123. The password **:x:** in **/etc/passwd** means:
   a. the encrypted password is **"x"**
   b. the account is locked
   c. the password is locked
   d. the encrypted password is stored in the shadow file
   e. the unencrypted password is stored in the group file

124. If the current directory contains files **abc**, **bbc**, **cbc**, and **bbc** contains just the line **dbd**, what is the output of the following command: **grep bb* bbc**
   a. no output *b.* **bbc** *c.* an error message
   d. **cbc** *e.* **dbd**

125. In a shell **case** structure, the **case** segment that will GLOB match the text **a**, **b**, or **c**, is coded as
   a. **a/b/c)** *b.* **a\b\c)** *c.* **a|b|c)**
   d. **a,b,c)** *e.* **a:b:c)**

126. 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
```
a. **pat** can access and write on the file
b. **pat** can rename the file
c. **bob** can list names in the directory
d. **bob** can access and write on the file
e. **bob** can create a new file in the directory

127. 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?
a. `./foo bar bar`
b. `./foo "$1" '$3'`
c. `./foo $3 $3`
d. `./foo '$3' bar`
e. `./foo "bar" 'bar'`

128. Given the following, can user **bird** in group **sesame** append to **foobar**?
```
drwx--xrwx 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. No, because execute permissions are not set for **bird** on **foobar**
d. Yes, because **bird** owns **foobar**
e. Yes, because **bird** has write permissions on **foobar**

129. 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
```
a. **bob** can access and write on the file
b. **pat** can create a new file in the directory
c. **pat** can rename the file
d. **bob** can list names in the directory
e. **bob** can rename the file

130. To show all your one-time scheduled commands, type:
a. `cat crontab`
b. `/var/log/crontab`
c. `/etc/crontab`
d. `crontab -l`
e. `atq`

131. 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?
a. Permissions **600** on directory **dir** and **700** on file **dir/bar**.
b. Permissions **200** on directory **dir** and **200** on file **dir/bar**.
c. Permissions **100** on directory **dir** and **200** on file **dir/bar**.
d. Permissions **400** on directory **dir** and **400** on file **dir/bar**.
e. Permissions **100** on directory **dir** and **100** on file **dir/bar**.

132. 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?
a. `crontab -e crontab.day`
b. `crontab -l crontab.day`
c. `crontab < crontab.day`
d. `echo crontab.day | crond`
e. `crontab > crontab.day`

133. If the file **foo** in the current directory contains just two lines **dbd**, and **123**, what is the output of the following command: `grep '[[:alnum:]]' foo`
a. **dbd**
b. no output or an error message
c. **foo**
d. **123**
e. both lines

134. 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. **pat** can rename the file
b. **bob** can access and write on the file
c. **bob** can list names in the directory
d. **bob** can create a new file in the directory
e. **pat** can access and write on the file

135. 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 the directory is not accessible to **bird**
b. No, because execute permissions are not set for **bird** on **foo**
c. No, because **bird** has no write permission on the directory
d. No, because **bird** has no write permissions on **foo**
e. Yes; permissions don't apply because **bird** owns **foo**

136. 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 **700** on directory **dir** and **200** on file **dir/bar**.
b. Permissions **600** on directory **dir** and **300** on file **dir/bar**.
c. Permissions **500** on directory **dir** and **500** on file **dir/bar**.
d. Permissions **700** on directory **dir** and **500** on file **dir/bar**.
e. Permissions **600** on directory **dir** and **500** on file **dir/bar**.

137. If variable **a** might contain nothing (a null value - defined but empty), which command sequence correctly tests for this and prints the date?
   a. `if [ "$a" = * ] ; then date ; fi`
   b. `if [ '''' = $a ] ; then date ; fi`
   c. `if test "" = "$a" ; then date ; fi`
   d. `if [ $a = /dev/null ] ; then date ; fi`
   e. `if test "" -eq $a ; then date ; fi`

138. User **bob** is in groups **bg1** and **bg2**. User **pat** is in group **pgg**.
```
d--xrwx-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
```
   a. **bob** can list names in the directory
   b. **bob** can access and write on the file
   c. **pat** can rename the file
   d. **bob** can create a new file in the directory
   e. **pat** can access and write on the file

139. Inside a shell script, which expands to the number of script arguments?
   a. `"$@"` *b.* `"$?"` *c.* `"$#"` *d.* `"$*"` *e.* `"$0"`

140. Which of the following could you use as options for the **tar** command to extract a **gzip**-compressed archive?
   a. `-tgz` *b.* `-czf` *c.* `xzf` *d.* `ezf` *e.* `egf`

141. 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?
   a. `ls .` *b.* `ls /home/foo/bar`
   c. `ls /home/foo/bar/..` *d.* `ls ..`
   e. `ls /home/foo/bar/.`

142. What value **umask** gives a new file permissions **r--r-----**?
   a. `440` *b.* `446` *c.* `220` *d.* `110` *e.* `337`

143. 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
```
   a. **pat** can rename the file
   b. **bob** can access and write on the file
   c. **bob** can list names in the directory
   d. **pat** can access and write on the file
   e. **bob** can create a new file in the directory

144. 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?
   a. `4c` *b.* `2a` *c.* `3a` *d.* `2b` *e.* `3b`

145. To change your own account password, use this exact command line:
   a. `$ passwd .`
   b. `$ passwd *`
   c. `$ passwd`
   d. `$ passwd cst8207`
   e. `$ passwd idallen-ubuntu`

146. Which **crontab** line executes at **13:54** every day?
   a. `54 13 * * * command` *b.* `* * * 13 54 command`
   c. `13 * * * 54 command` *d.* `* * * 54 13 command`
   e. `13 54 * * * command`

147. 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 access and write on the file
   b. **pat** can rename the file
   c. **pat** can create a new file in the directory
   d. **bob** can list names in the directory
   e. **bob** can create a new file in the directory

148. A **crontab** entry of `0 6 * * * /sbin/somescript` would run **somescript** when and how often?
   a. at 12:06am every business day and Saturday
   b. at 6:00am every business day
   c. at 12:06am every business day
   d. at 12:06am every day
   e. at 6:00am every day

149. What command terminates processes based on their name (not safe!):
   a. `dmesg` *b.* `kill` *c.* `crontab`
   d. `killall` *e.* `ps lxww`

150. 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?
   a. Permissions **500** on directory **dir** and **100** on file **dir/bar**.
   b. Permissions **400** on directory **dir** and **400** on file **dir/bar**.
   c. Permissions **500** on directory **dir** and **200** on file **dir/bar**.
   d. Permissions **600** on directory **dir** and **700** on file **dir/bar**.
   e. Permissions **200** on directory **dir** and **200** on file **dir/bar**.

151. 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
```
a. No, because **foo** has no write permissions for **bird**
b. Yes; permissions don't apply because **bird** owns **foo**
c. Yes, because **bird** has read permissions on **foo**
d. No, because the directory is not accessible to **bird**
e. No, because the directory has no write permissions for **bird**

152. What command displays the groups you are in?
a. **gpasswd** *b.* **mkgroups** *c.* **groups**
d. **lstgroups** *e.* **grouprint**

153. In an empty directory, what permissions are on file **???** after these commands:
```
touch ??? *** ; chmod 111 *
chmod 222 ? ; chmod 444 '*'
```
a. **--x--x--x** *b.* **-w--w--w-** *c.* **rw-rw-rw-**
d. **-wx-wx-wx** *e.* **r--r--r--**

154. 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
```
a. **bob** can rename the file
b. **pat** can create a new file in the directory
c. **bob** can access and write on the file
d. **pat** can rename the file
e. **bob** can list names in the directory

155. The **minimum** permissions you need to copy a file **foo** from directory **a** to directory **b** are:
a. **rx** on **a**, **wx** on **b**, **w** on **foo**
b. **x** on **a**, **wx** on **b**, **r** on **foo**
c. **wx** on **a**, **wx** on **b**, none on **foo**
d. **rwx** on **a**, **wx** on **b**, none on **foo**
e. **wx** on **a**, **wx** on **b**, **rw** on **foo**

156. If **guru=linus** then which one of the following **case** patterns will match this statement: **case "$guru" in**
a. ***) echo yes ;;**
b. **lin?) echo yes ;;**
c. **(*nus echo yes ;;**
d. **"linu?") echo yes ;;**
e. **[linus] | [LINUS]) echo yes ;;**

157. 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
```
a. **bob** can rename the file
b. **bob** can access and write on the file
c. **pat** can rename the file
d. **bob** can list names in the directory
e. **pat** can create a new file in the directory

158. 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
```
a. **pat** can rename the file
b. **bob** can create a new file in the directory
c. **pat** can access and write on the file
d. **bob** can access and write on the file
e. **bob** can list names in the directory

159. When an **at** job runs, the current working directory is set to:
a. the HOME directory of the user who created the job
b. the current directory that was in use when the **at** job was created
c. the directory with the name **/home**
d. the system ROOT directory
e. the directory with the name **/root**

160. 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
```
a. **pat** can rename the file
b. **bob** can create a new file in the directory
c. **pat** can access and write on the file
d. **bob** can access and write on the file
e. **bob** can list names in the directory

161. The **minimum** permissions you need to delete a file **foo** from directory **a** are:
a. **rwx** on **a**, none on **foo** *b.* **rwx** on **a**, **rw** on **foo**
c. **wx** on **a**, **w** on **foo** *d.* **wx** on **a**, **r** on **foo**
e. **wx** on **a**, none on **foo**

162. The output of the **whoami** command is:
a. a list of users logged in to the system
b. a list of accounts in the password file
c. your userid
d. the current directory
e. your HOME directory

163. The signal sent to a foreground process by typing the **[Ctrl-C]** key is:
a. **SIGKILL** *b.* **SIGTERM** *c.* **SIGSTOP**
d. **SIGINT** *e.* **SIGHUP**

164. 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?
a. Permissions **300** on directory **dir** and **200** on file **dir/foo**.
b. Permissions **400** on directory **dir** and **400** on file **dir/foo**.
c. Permissions **600** on directory **dir** and **700** on file **dir/foo**.
d. Permissions **500** on directory **dir** and **600** on file **dir/foo**.
e. Permissions **100** on directory **dir** and **100** on file **dir/foo**.

165. The **-v** option to the **grep** command does what?
a. turns off the translation of unprintable characters
b. selects lines that do not contain a match for the supplied pattern
c. turns on the translation of unprintable characters
d. prints the version number of the grep command
e. selects lines that do not contain unprintable characters

166. Which command line below does not show any lines from inside the file **bat**?
a. **tail bat** *b.* **ls bat** *c.* **head bat**
d. **more bat** *e.* **less bat**

167. To send a **KILL** signal to a process with process ID *PID*, which of the following commands would you use?
a. **kill** *PID* **KILL** *b.* **send -KILL** *PID*
c. **kill -KILL** *PID* *d.* **signal -KILL** *PID*
e. **send** *PID* **KILL**

168. 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. Yes, because **bird** has write permissions on **foo**
b. No, because **foo** has no read permissions for **bird**
c. No, because the directory has no write permissions for others
d. Yes; permissions don't apply because **bird** owns **foo**
e. No, because the directory is not accessible to **bird**

169. 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. No, because execute permissions are not set for **bird** on **foo**
b. Yes, because **bird** has write permissions on **foo**
c. Yes; permissions don't apply because **bird** owns **foo**
d. No, because the directory is not accessible to **bird**
e. No, because **bird** has no write permission on the directory

170. 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
```
a. **bob** can access and write on the file
b. **bob** can create a new file in the directory
c. **bob** can list names in the directory
d. **pat** can access and write on the file
e. **pat** can rename the file

171. 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
```
a. Yes; permissions don't apply because **bird** owns **foo**
b. Yes, because **bird** has full permissions on **foo**
c. Yes, because **bird** matches the writable other permissions
d. No, because **bird** has no write permission on the directory
e. No, because the directory is not accessible to **bird**

172. 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
```
a. **bob** can list names in the directory
b. **bob** can create a new file in the directory
c. **pat** can rename the file
d. **pat** can access and write on the file
e. **bob** can access and write on the file

173. What would be the output of the following command line:
```
echo a b c d | awk '{print $NF}'
```
a. no output *b.* **4** *c.* **a b c d**
d. **$NF** *e.* **d**

174. If the file **foo** in the current directory contains just the line **dbd**, what is the output of the following command: **grep '[b1]' foo**
a. **123** *b.* **no output** *c.* **dbd**
d. an error message *e.* **foo**

175. Dereference the following symlink **bar** into its equivalent absolute path:
```
ln -s ../b/../b/../../foo /tmp/a/b/bar
```
a. **/tmp/foo** *b.* **/tmp/b/foo** *c.* **/tmp/b/bar**
d. **/tmp/a/b/bar** *e.* **/tmp/a/foo**

176. 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
```
a. Yes, because **bird** has write permissions on **foo**
b. No, because the directory is not accessible to **bird**
c. No, because **bird** has no write permission on the directory
d. No, because execute permissions are not set for **bird** on **foo**
e. Yes; permissions don't apply because **bird** owns **foo**

177. What is the output on your screen of the following sequence of commands:
```
x=pig ; [ -z $x ] ; echo $?
```
a. the number 0 or 1 followed by another 0 or 1 on a new line
b. **0**
c. **1**
d. no output
e. **test: $x: integer expression expected**

178. If the file **foo** in the current directory contains just the line **abc123**, what is the output of the following command: **grep '^[[:alpha:]]' foo**
a. no output *b.* an error message *c.* **abc123**
d. **123** *e.* **abc**

179. 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
```
a. **pat** 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

180. Under what directory are system configuration files usually stored?
a. **/etc** *b.* **/var/log/** *c.* **/bin/**
d. **/usr/bin** *e.* **/log/var/**

181. Which of the following commands would result in an error?
a. **[a = 4]** *b.* **[a != 4]** *c.* **[3 = f]**
d. **[3 -eq 4]** *e.* **[3 -e 3]**

182. 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.* **4c** *c.* **3b** *d.* **2b** *e.* **2a**

183. 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 /tmp/dir/..** *b.* **ls .**
c. **ls /tmp/dir** *d.* **ls ..**
e. **ls /tmp/dir/.**

184. 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
```
a. **pat** can rename the file
b. **pat** 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. **bob** can access and write on the file

185. 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
```
a. **pat** can rename 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 access and write on the file

186. 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
```
a. **pat** can access and write on the file
b. **bob** can list names in the directory
c. **bob** can create a new file in the directory
d. **pat** can rename the file
e. **bob** can access and write on the file

187. If the file **foo** in the current directory contains just the line **dbd**, what is the output of the following command: **grep '[b1]$' foo**
a. **foo** *b.* **dbd** *c.* an error message
d. no output *e.* **123**

188. What value to **chmod** would change the permissions on a file to **rw-r--r--**?
a. **211** *b.* **244** *c.* **344** *d.* **311** *e.* **644**

189. If the line, **`exit 2`**
is executed in a shell script, what is the result?
a. termination after sleeping for 2 seconds
b. an invalid argument error message
c. termination with an exit status of 2
d. the script breaks out of up to 2 levels of loops
e. termination with an exit status of 0

190. If the current directory contains files **`abc`**, **`bbc`**, **`cbc`**, and **`bbc`** contains just the line **`dbd`**, what is the output of the following command: **`grep 'bb*' bbc`**
a. **`dbd`** *b.* **`bbc`** *c.* **`cbc`**
d. no output *e.* an error message

191. 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 access and write on the file
b. **`pat`** 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

192. 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
```
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

193. 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
```
a. **`pat`** can create a new file in the directory
b. **`bob`** can access and write on the file
c. **`pat`** can rename the file
d. **`bob`** can list names in the directory
e. **`bob`** can rename the file

194. If **`browser=lynx`** then which one of the following **`case`** patterns will match this statement: **`case "$browser" in`**
a. **`(*ynx  echo yes ;;`**
b. **`?lynx? ) echo yes ;;`**
c. **`l?n? ) echo yes ;;`**
d. **`[lynx] | [LYNX] ) echo yes ;;`**
e. **`@ ) echo yes ;;`**

195. 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. **`chmod 777 .`** *b.* **`chmod 333 dir`**
c. **`chmod 777 dir`** *d.* **`cd dir ; chmod go-x .`**
e. **`cd dir ; chmod go+wx .`**

196. 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`**

197. 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 -s /one/foo /two/bar`**
b. **`ln /one/bar /two/foo`**
c. **`ln /two/bar /one/foo`**
d. **`ln /one/foo /two/bar`**
e. **`ln -s /two/bar /one/foo`**

198. What value **`umask`** gives a new file permissions **`r--r-----`**?
a. **`220`** *b.* **`226`** *c.* **`440`** *d.* **`110`** *e.* **`446`**

199. The **minimum** permissions you need to link a file **`foo`** from directory **`a`** to directory **`b`** are:
a. **`rwx`** on **`a`**, **`wx`** on **`b`**, **`rw`** on **`foo`**
b. **`rwx`** on **`a`**, **`wx`** on **`b`**, none on **`foo`**
c. **`wx`** on **`a`**, **`wx`** on **`b`**, **`w`** on **`foo`**
d. **`wx`** on **`a`**, **`wx`** on **`b`**, **`r`** on **`foo`**
e. **`x`** on **`a`**, **`wx`** on **`b`**, none on **`foo`**

200. If the current directory contains files **`abc`**, **`bbc`**, **`cbc`**, and **`bbc`** contains just the line **`dbd`**, what is the output of the following command: **`grep "bb*" bbc`**
a. **`bbc`** *b.* no output *c.* an error message
d. **`dbd`** *e.* **`cbc`**

201. Dereference the following symlink **`bar`** into its equivalent absolute path:
`ln -s ../b/../../a/./foo /tmp/a/b/bar`
a. **`/tmp/a/foo`** *b.* **`/tmp/b/bar`** *c.* **`/tmp/a/b/bar`**
d. **`/tmp/b/foo`** *e.* **`/tmp/foo`**

202. 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
```
a. **`pat`** can access and write on the file
b. **`bob`** can list names in the directory
c. **`bob`** can access and write on the file
d. **`pat`** can rename the file
e. **`bob`** can create a new file in the directory

203. **Did you write the Test Version number on the Scantron Form?**
Did you read all the other words of the test instructions on page one?

a. **`Tak`** *(Yes - Polish)*
b. **`Jes`** *(Yes - Esperanto)*
c. **`Igen`** *(Yes - Hungarian)*
d. **`Sim`** *(Yes - Portuguese)*
e. **`Taip`** *(Yes - Lithuanian)*

This page intentionally left blank.