

PRINT Name: _____ LAB Section: **One-Answer Multiple Choice 347 Questions****Weight 15%**

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

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 A D B C C

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. **Did you read all the words of the test instructions on page one?**
 - a. **Jes** (Yes - Esperanto)
 - b. **Sim** (Yes - Portuguese)
 - c. **Tak** (Yes - Polish)
 - d. **Taip** (Yes - Lithuanian)
 - e. **Igen** (Yes - Hungarian)
2. **My three-digit Lab Section number is:**
 - a. My lecture room number, i.e. **C346**
 - b. The Test Version code printed on the question sheet.
 - c. My lecture Section Number, i.e. **010**
 - d. The timetable Section Number of my weekly 2-hour lab period.
 - e. My lab room number, e.g. **T303, A213, B332**
3. What is the output on your screen after this:


```
mkdir dir ; touch dir/.aa dir/.bb ; echo dir/*
```

 - a. **dir/.aa dir/.bb**
 - b. **dir/. dir/.. dir/.aa dir/.bb**
 - c. **dir/**
 - d. no output
 - e. **dir/***
4. Which environment variable contains your HOME directory?
 - a. **\$home**
 - b. **\$/HOME**
 - c. **/home/abcd0001**
 - d. **/home**
 - e. **\$HOME**

5. What is the output of this in an empty directory:


```
touch a .a bc .bc def ; echo [ab]*
```

 - a. **a bc**
 - b. no output
 - c. **a .a bc .bc**
 - d. **[ab]***
 - e. an error message from **echo** saying **[ab]*** does not exist
6. Which can generate a non-empty file?
 - a. **cat foo >foo**
 - b. **sort foo >foo**
 - c. **tail foo >foo**
 - d. **ls foo >foo**
 - e. **fgrep 'foo' foo >foo**
7. What is the link count of directory **dir** after these successful commands?


```
mkdir dir ; cd dir ; touch one ; mkdir two
```

 - a. **5**
 - b. **3**
 - c. **1**
 - d. **2**
 - e. **4**
8. A "dangling symlink" is a symlink to:
 - a. a directory
 - b. a special device file
 - c. the current directory
 - d. a parent directory
 - e. a non-existent target
9. In a directory containing one file named **dog**, what is the output on your screen after this: **2>/dev/null ls nosuchfile**
 - a. **nosuchfile**
 - b. **bash: 2>/dev/null: command not found**
 - c. **ls: nosuchfile: No such file or directory**
 - d. no output
 - e. **dog**
10. How many lines are in the file **bar** after this:


```
echo hi >x ; echo ho >>x ; cat x >bar
```

 - a. **2**
 - b. **4**
 - c. **1**
 - d. **6**
 - e. **0**
11. What is the link count of directory **dir** after these successful commands?


```
mkdir dir ; mkdir dir/foo ; touch dir/bar
```

 - a. **4**
 - b. **3**
 - c. **5**
 - d. **2**
 - e. **1**
12. What is the output on your screen after these command lines:


```
echo 1 >x ; cp x y ; echo 2 >>y
sort x >y ; cat y
```

 - a. **2**
 - b. **1 followed by 2**
 - c. no output
 - d. **1**
 - e. **2 followed by 1**
13. How many arguments are passed to the command by the shell:


```
$ <foo foo " a 'b c' d " e ' f " g " ' >foo h
```

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

14. What is the link count of directory **d** after these successful commands?
`mkdir d d/a d/b d/c d/c/z ; touch d/x d/y`
 a. 4 b. 3 c. 6 d. 2 e. 5
15. In an empty directory, what is the output on your screen after this:
`touch 1 2 .a .b ; echo .*`
 a. `.a .b`
 b. `. . . .a .b`
 c. `.*`
 d. `1 2`
 e. an error message from **echo** saying `.*` does not exist
16. Which shows only lines 6-10 of file **foo**?
 a. `head -10 foo | tail -6` b. `tail -15 foo | head -5`
 c. `tail -10 foo | head -6` d. `head -6 foo | tail -10`
 e. `head -10 foo | tail -5`
17. If `/bin/foo` is a program that outputs **one** and `/usr/bin/foo` is a program that outputs **two**, what is the output on your screen after this:
`PATH=/etc:/usr/bin:/usr:/bin:/dev ; foo`
 a. **two**
 b. **bash: foo: command not found**
 c. **two** followed by **one**
 d. **one**
 e. **one** followed by **two**
18. How many lines are in the file **out** after this:
`date >f ; ls f >>f ; cat f f >out`
 a. 2 b. 6 c. 4 d. 1 e. 0
19. Which option to **ls** displays the directory itself and not its contents?
 a. `-a` b. `-R` c. `-d` d. `-i` e. `-l`
20. If directory **dir** contains four three-character file names: `.on`, `.tw`, `.th`, `.f.`, then what is the output on your screen of this: `echo dir/*`
 a. `dir/ dir/.. dir/.on dir/.tw dir/.th dir/.f.`
 b. `dir/.f.`
 c. `dir/.on dir/.tw dir/.th`
 d. no output
 e. `dir/*`
21. If you are in `/bin` and `ls -l` shows a symbolic link **bar** `-> ../dir/foo` then dereference the absolute path of **bar** with no symbolic links:
 a. `/bin/dir/foo` b. `/bin/bar/dir/foo`
 c. `/bin/dir/foo/bar` d. `/bar/./dir/foo`
 e. `/dir/foo`

22. In an empty directory, what is the output on your screen after this:
`touch a ; ls >wc -l`
 a. no output b. 1 c. 3
 d. 0 e. 2
23. Which tells you the recursive count of all pathnames under the current directory and all subdirectories?
 a. `wc *` b. `ls | wc` c. `wc .`
 d. `find | wc` e. `wc "$PWD"`
24. You enter this `cp a/b c/` and get `cp: a: No such file or directory` because:
 a. pathname **a** exists but is a file, not a directory
 b. directory **a** does not exist
 c. directory **c** does not exist
 d. you forgot to specify the destination file name after **c/**
 e. the command **cp** is not in your search **PATH**
25. How many files are touched? `touch "1 " 2 3" " ' ' 4 5`
 a. 4 b. 3 c. 7 d. 6 e. 5
26. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cp b a | head`
 a. 2 b. 3 c. 2 followed by 3
 d. no output e. 3 followed by 2
27. What is the output of this in an empty directory:
`touch 1 2 3 .a .b .c ; echo .??*`
 a. `. . . 1 2 3 .a .b .c`
 b. `.a .b .c`
 c. `. . . .a .b .c`
 d. `.*?*`
 e. an error message from **echo** saying `.*?*` does not exist
28. In an empty directory, what is the output on your screen after this:
`echo hi >a ; sort * 1>/dev/null`
 a. `sort: 1>/dev/null: No such file or directory`
 b. `sort: *: No such file or directory`
 c. **a**
 d. **hi**
 e. no output

29. In an empty directory, what is the output on your screen after this:
`echo hi >a ; ls nosuchfile 2>/dev/null`
 a. no output
 b. `ls: 2>/dev/null: No such file or directory`
 c. `a`
 d. `ls: nosuchfile: No such file or directory`
 e. `nosuchfile`
30. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are output on your screen by this: `cat b | cat a`
 a. 5
 b. 2
 c. 2 followed by 3
 d. 3 followed by 2
 e. 3
31. What is the resulting link count of empty directory `dir` after these successful commands? `cd dir ; touch foo ; ln foo one ; ln foo two`
 a. 4
 b. 2
 c. 5
 d. 1
 e. 3
32. What is the link count of file `f` after these successful commands?
`rm f ; touch f ; ln f bar ; ln bar x`
`cp bar a ; ln a b ; ln -s x c ; cp c d`
 a. 1
 b. 3
 c. 4
 d. 5
 e. 2
33. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this: `cat a a >c ; head b >>a ; cat c b >c a`
 a. 12
 b. 0
 c. 8
 d. 7
 e. 10
34. Dereference the following symlink `xyz` into its equivalent absolute path:
`ln -s ../../a/./b/./bar /tmp/a/b/xyz`
 a. `/tmp/a/b/bar`
 b. `/tmp/bar`
 c. `/tmp/a/bar`
 d. `/tmp/b/bar`
 e. `/tmp/b/xyz`
35. Which displays only the non-hidden names in the current directory that contain the letter `a` (and no other names)?
 a. `echo a*`
 b. `echo [a]`
 c. `echo *a*`
 d. `echo ?a?`
 e. `echo *a`
36. What is in file `out` after this:
`echo me >a ; ln a b ; echo hi >b ; ln a out ; rm a b`
 a. nothing (empty file)
 b. `me`
 c. `me` followed by `hi`
 d. `hi`
 e. no such file (nonexistent)
37. Which makes pathnames `/usr/local/bin` and `/usr/bin` lead to the same directory?
 a. `touch /usr/local`
 b. `ln . /usr/local`
 c. `rmdir /usr/local`
 d. `ln -s . /usr/local`
 e. `mkdir /usr/local`

38. What is the link count of directory `z` after these successful commands?
`mkdir z ; cd z ; touch a ; ln a b ; ln a c`
 a. 3
 b. 4
 c. 1
 d. 2
 e. 5
39. If directory `cow` contains four three-character file names: `.AA`, `.A1`, `.BB`, `.B.`, then what is the output on your screen of this: `echo cow/*`
 a. `cow/.B.`
 b. `cow/.AA cow/.A1 cow/.BB cow/.B.`
 c. no output
 d. `cow/.AA cow/.A1 cow/.BB`
 e. `cow/*`
40. File `foo` contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9). What is the output on your screen of this:
`cat foo foo | sort -r | head -n 4 | tail -n 1`
 a. 5
 b. 9
 c. 8
 d. 7
 e. 6
41. What is the link count of file `f` after these successful commands?
`rm f ; touch f ; cp f x`
`ln f a ; ln x y ; ln a z ; ln x b`
 a. 5
 b. 3
 c. 4
 d. 2
 e. 6
42. If `/bin/foo` is a program that outputs `hi` and `/usr/bin/foo` is a program that outputs `mom` what is the output on your screen after this:
`PATH=/etc:/usr/bin:/bin ; foo`
 a. `hi` followed by `mom`
 b. `hi`
 c. `mom` followed by `hi`
 d. `bash: foo: command not found`
 e. `mom`
43. If `mt` is an empty sub-directory, what is true after this:
`touch mt/bar ; mkdir bar ; mv mt/bar mt/./bar/me`
 a. the directory `mt` is now empty
 b. the `mkdir` fails because `bar` already exists
 c. the directory `mt` now contains only a file named `me`
 d. there is a second copy of the file `bar` in the file named `me`
 e. the command fails because the name `mt/./bar/me` does not exist
44. What does *quoting* mean on a shell command line?
 a. using a leading tilde ("`~`") on a pathname to mean your `HOME` directory
 b. using more than one pathname argument to a command, e.g. `rm a b c`
 c. typing a "control" character using the `[CTRL]` key
 d. setting the `PS1` variable to be your shell prompt
 e. turning off the special meaning of shell meta-characters
45. How many lines are in file `out` after this: `date >wc >cat >out`
 a. 0 0 0
 b. 2
 c. 1 6 29
 d. 0
 e. 1

46. What is the link count of directory **dir** after these successful commands?
`mkdir dir ; cd dir ; touch a b c ; mkdir d e`
 a. 7 b. 3 c. 2 d. 5 e. 4
47. Which command recursively finds all things named **foo**?
 a. `echo -name foo` b. `find -name foo`
 c. `fgrep -name foo` d. `ls -name foo`
 e. `cat -name foo`
48. What displays on your screen given this command:
`date >date ; pwd >pwd ; head date | tail pwd`
 a. only the **pwd** displays because **tail** ignores the pipe
 b. only the **date** displays because **tail** ignores the pipe
 c. **tail** reads the pipe and the **pwd** and displays both together
 d. nothing displays because **tail** ignores the pipe
 e. **head** displays the **date** and **tail** displays the **pwd**
49. If the file **pig** contained the word **bar**, what is the output on your screen after this: `PATH=/etc/passwd:/bin/ls:/bin/who ; /bin/cat pig`
 a. `bash: /bin/cat: command not found`
 b. **bar**
 c. no output on screen
 d. **pig**
 e. `/bin/cat: pig: No such file or directory`
50. What is in file **c** after this:
`echo B >b ; ln b a ; echo A >a ; ln a c ; rm a b`
 a. **B** b. nothing (empty file)
 c. **A** followed by **B** d. no such file (nonexistent)
 e. **A**
51. If I have a directory named **a/b**, which action would increase its *link count* by exactly one?
 a. create a hard link to directory **b** named **b2**
 b. create a directory named **a/b2**
 c. create a file named **a/b/c**
 d. create a directory named **a/b/c**
 e. create a file named **a/b2**
52. Which command line has exactly one argument?
 a. `echo "It's "'funny how' " it's done."`
 b. `echo 'It's "funny how" it's done.'`
 c. `echo 'It's "'funny how'" it's done.'`
 d. `echo "It's " 'funny how'" it's done."`
 e. `echo "It's "'funny how'" it's done."`

53. In an empty directory, what is in file **foo** after this:
`echo hi >foo ; ls nosuchfile | cat >foo`
 a. **foo**
 b. **hi**
 c. **nosuchfile**
 d. `ls: cannot access nosuchfile`
 e. nothing (empty file)
54. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
`111 -rw-r--r-- 3 me me 100 Jan 1 1:00 a`
`111 -rw-r--r-- 3 me me 100 Jan 1 1:00 b`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 c`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 d`
 a. 0 b. 1 c. 3 d. 2 e. 4
55. What is the output on your screen after these command lines:
`echo one >x ; ln x y ; echo two >y`
`echo ten >x ; cat y`
 a. **one** b. **ten**
 c. no output on screen d. **one** followed by **two** and **ten**
 e. **two**
56. In an empty directory, what is in file **count** after this:
`ls ??? | wc -w >count`
 a. 1 1 1 b. 0
 c. 1 1 2 d. nothing (empty file)
 e. 1
57. How many lines are in the file **out** after this:
`echo hi >x ; echo ho >>x ; cat x x x >out`
 a. 0 b. 2 c. 6 d. 1 e. 3
58. What is the link count of file **f** after these successful commands?
`rm f ; touch f ; ln f b ; cp f g`
`cp b a ; ln a d ; ln b c ; cp c g`
 a. 4 b. 1 c. 3 d. 2 e. 5
59. In an empty directory, what is the output on your screen after this:
`ls l>/dev/null nosuchfile`
 a. `ls: /dev/null: No such file or directory`
 b. **nosuchfile**
 c. `ls: l>/dev/null nosuchfile: No such file or directory`
 d. `ls: nosuchfile: No such file or directory`
 e. no output
60. In an empty directory, how many words are in file **out** after this:
`echo hi >a ; ls >out`
 a. 3 b. 0 c. 1 d. 2 e. 4

61. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `ln a d ; ln d c ; cp c b ; sort a b d >c`
 a. 4 b. 6 c. 5 d. 0 e. 2
62. What is the output on your screen after this: `echo hi >out | wc -w`
 a. 3 b. no output c. 1
 d. 2 e. 0
63. What is the link count of directory **d** after these successful commands?
`mkdir d ; mkdir d/a d/b ; touch d/c d/e`
 a. 5 b. 1 c. 4 d. 3 e. 2
64. Which displays only the non-hidden names in the current directory that contain the case-insensitive word **me** (and no other names)?
 a. `echo *[MmEe]*` b. `echo ?[MmEe]?`
 c. `echo *(M,m,E,e)*` d. `echo *[Mm][Ee]*`
 e. `echo *[me]*`
65. If `/bin/pig` is a program that outputs **hi** and `/usr/bin/pig` is a program that outputs **foo** what is the output on your screen after this:
`PATH=/etc:/usr/bin:/bin ; pig`
 a. **hi** followed by **foo**
 b. **foo**
 c. **foo** followed by **hi**
 d. **hi**
 e. **bash: pig: command not found**
66. In an empty directory, how many words are in file **out** after this:
`touch 1 2 3 2 1 ; ls >out`
 a. 5 b. 3 c. 0 d. 4 e. 6
67. How many arguments are passed to the command by the shell:
`$ <foo foo " a 'b c' d " e ' f " g " ' >foo`
 a. 5 b. 3 c. 4 d. 2 e. 6
68. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cat a | cat b`
 a. no output b. 0 c. 5
 d. 2 e. 3
69. What is the link count of file **f** after these successful commands?
`rm f ; touch f ; ln f bar`
`cp bar x ; ln x y ; ln y z`
 a. 4 b. 3 c. 1 d. 2 e. 0
70. How many arguments are passed to the command by the shell:
`$ <pig pig -x " " -z -r" " >pig pig pig`
 a. 8 b. 5 c. 7 d. 9 e. 6

71. What is the usual output on your screen of this:
`mkdir dir ; cd dir >dir/foo ; cat foo`
 a. **foo**
 b. **dir**
 c. no output
 d. **bash: dir/foo: No such file or directory**
 e. **cat: foo: No such file or directory**
72. What is the usual output on your screen of this:
`mkdir dir ; cd dir >foo ; cat foo`
 a. **bash: cd: dir: No such file or directory**
 b. **dir**
 c. **cat: foo: No such file or directory**
 d. no output
 e. **foo**
73. In a directory containing one file named **dog**, what is the output on your screen after this: `1>/dev/null ls *`
 a. **bash: 1>/dev/null: command not found**
 b. no output
 c. **dog**
 d. **ls: *: No such file or directory**
 e. *****
74. If you are in `/bin` and `ls -l` shows a symbolic link **foo** `-> /bar` then dereference the absolute path of **foo** with no symbolic links:
 a. `/bin/bar` b. `/bin/foo/bar` c. `/foo/bar`
 d. `/bar` e. `/bin/bar/foo`
75. Which shows the file in `/bin` with the largest checksum?
 a. `cat /bin | sum | sort -nr | head -n 1`
 b. `cat /bin/* | sum | sort -nr | head -n 1`
 c. `sum /bin | sort -nr | head -n 1`
 d. `ls /bin/* | sum | sort -nr | head -n 1`
 e. `sum /bin/* | sort -nr | head -n 1`
76. Which command counts lines containing the string **refused** in only the month **October 2016** in the **denyhosts** log file?
 a. `fgrep refused denyhosts | fgrep -c October 2016`
 b. `fgrep refused denyhosts ; fgrep -c 2016-10`
 c. `fgrep '2016-10 refused' denyhosts`
 d. `fgrep -c 2016-10 denyhosts | fgrep refused`
 e. `fgrep 'refused' denyhosts | fgrep -c '2016-10'`
77. What is the link count of an empty directory?
 a. 0 b. 1 c. 2 d. 3 e. 4

78. What is the link count of directory **d** after these successful commands?
`mkdir d ; mkdir d/a ; mkdir d/b ; mkdir d/b/c`
 a. 3 b. 2 c. 1 d. 5 e. 4
79. In an empty directory, how many arguments are passed to the **cat** command in this: `touch a1 a2 ac ba .a ; cat a*`
 a. 2 b. none c. 1 d. 4 e. 3
80. In a directory containing one file named **mt**, what is the output on your screen after this: `ls 2>/dev/null nosuchfile`
 a. `ls: nosuchfile: No such file or directory`
 b. `bash: 2>/dev/null: command not found`
 c. `nosuchfile`
 d. `mt`
 e. no output
81. Create a symbolic link under **/usr** named **bar** that has target **xy**:
 a. `ln -s 'xy' /bar/usr`
 b. `ln -s /usr/bar 'xy'`
 c. `ln -s '/usr/xy' /usr/bar`
 d. `ln -s /usr/bar '/usr/xy'`
 e. `ln -s 'xy' '/usr/bar'`
82. What command will recursively show disk usage in directories?
 a. `tree` b. `ls` c. `df` d. `find` e. `du`
83. What is the output of this in an empty directory:
`touch .12 .345 .6789 ; echo .??*`
 a. an error message from **echo** saying `.??*` does not exist
 b. `.12 .345 .6789`
 c. `. . . .12 .345 .6789`
 d. no output
 e. `.??*`
84. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
`111 -rw-r--r-- 2 me me 100 Jan 1 1:00 a`
`111 -rw-r--r-- 2 me me 100 Jan 1 1:00 b`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 c`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 d`
 a. 0 b. 1 c. 2 d. 3 e. 4
85. File **foo** contains 99 lines, each of which is the two-digit line number of the line in the file (01 through 99). What is the output on your screen of this:
`sort -r foo foo | tail -n 4 | head -n 1`
 a. `04 04` b. `98` c. `96 96`
 d. `96` e. `02`

86. File **foo** contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9). What is the output on your screen of this:
`cat foo foo | cat | tail -n 4 | head -n 1`
 a. 6 b. 9 c. 7 d. 5 e. 8
87. If **/bin/xxx** is a program that outputs **one** and **/usr/bin/xxx** is a program that outputs **two**, what is the output on your screen after this:
`PATH=/usr:/usr/bin:/etc:/bin ; xxx`
 a. `bash: xxx: command not found`
 b. **two** followed by **one**
 c. **one**
 d. **one** followed by **two**
 e. **two**
88. The correct syntax to assign to a shell variable is:
 a. `V = foo bar` b. `V = "foo bar"`
 c. `V="foo bar"` d. `"V=foo bar"`
 e. `V=foo bar`
89. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cp a b >c ; cat a b c`
 a. 7 b. 6 c. 5 d. 4 e. 10
90. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
`111 -rw-r--r-- 2 me me 100 Jan 1 1:00 a`
`111 -rw-r--r-- 2 me me 100 Jan 1 1:00 b`
`222 -rw-r--r-- 2 me me 100 Jan 1 1:00 c`
`222 -rw-r--r-- 2 me me 100 Jan 1 1:00 d`
 a. 4 b. 0 c. 3 d. 1 e. 2
91. In an empty directory, what is the output on your screen after this:
`touch 1 2 3 ; cow="*" ; echo $cow`
 a. `*` b. `$cow` c. `1 2 3`
 d. `"1 2 3"` e. `"*"`
92. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cat a | ls b`
 a. no output b. 3 c. 2
 d. 5 e. 1
93. What is the output of this in an empty directory:
`touch .a .b .c ; echo .*`
 a. `.a .b .c`
 b. `.*`
 c. an error message from **echo** saying `.*` does not exist
 d. `. . . .a .b .c`
 e. no output

94. How many arguments are passed to the command by the shell:
`$ <bar bar -b "-a" '-r' >bar bar bar`
 a. 4 b. 3 c. 7 d. 5 e. 6
95. If `/bin/foo` is a program that outputs **one** and `/usr/bin/foo` is a program that outputs **two**, what is the output on your screen after this:
`PATH=/bin/ls:/home:/usr/bin/cat:/etc ; foo`
 a. **two** followed by **one**
 b. **one**
 c. **one** followed by **two**
 d. **two**
 e. **bash: foo: command not found**
96. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `ln a d ; ln d c ; cat a b >c`
 a. 5 b. 4 c. 0 d. 3 e. 2
97. What is true about this output from `ls -il foo bar`
`15 -r-x----- 2 me me 3 Jan 1 1:00 foo`
`99 -r-x----- 2 me me 3 Jan 1 1:00 bar`
 a. **foo** and **bar** are names for different files
 b. **foo** and **bar** are names for the same file
 c. **foo** and **bar** are two of three names for the same file
 d. **foo** and **bar** each have three names (six names total)
 e. this output is not possible
98. In an empty directory, how many words are in file **out** after this:
`touch a ; ls >out`
 a. 0 b. 2 c. 3 d. 1 e. 4
99. Which shows just the count of words in the file?
 a. `wc file | awk '{print #2}'`
 b. `wc file | awk '{print $2}'`
 c. `wc file | awk '[print #2]'`
 d. `wc file | awk '[print $2]'`
 e. `wc file | awk '{print 2}'`
100. In an empty directory, what is the output on your screen after this:
`ls out 2>/dev/null`
 a. `ls: out 2>/dev/null: No such file or directory`
 b. `out`
 c. no output
 d. `ls: out: No such file or directory`
 e. `ls: /dev/null: No such file or directory`
101. File **foo** contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9). What is the output on your screen of this:
`sort foo foo | tail -n 2 | head -n 1`
 a. 8 b. 8 8 c. 9 d. 2 2 e. 1

102. Which one of these names is usually a shell environment variable?
 a. **fooBar** b. **foobar** c. **FooBar**
 d. **FOOBAR** e. **FooBar**
103. How many lines are in the file **bar** after this:
`echo hi >x ; echo ho >>x ; cat x x >bar`
 a. 6 b. 4 c. 0 d. 2 e. 1
104. If you want a user-defined alias in all your **bash** shells, what do you do?
 a. put the alias into the `/etc/passwd` file for next log in
 b. put the alias into the `/bin/bash` file for next log in
 c. define the alias in my file `$HOME/.bashrc`
 d. put the alias into the `/etc/group` file for next log in
 e. create the alias and then type **save** to save it to all shells
105. What is the output on your screen after this:
`PATH=/bin/cat:/bin/sh:/bin/ls ; ls nosuchfile`
 a. `ls: /bin/ls: command not found`
 b. `bash: /bin/sh: No such file or directory`
 c. `bash: ls: command not found`
 d. `bash: /bin/ls: command not found`
 e. `ls: nosuchfile: No such file or directory`
106. Rewrite as a simplified absolute path:
`../../../../var/./a/../../../../var/b/../../../../etc/./bar/./foo`
 a. `/var/foo` b. `/etc/bar/foo` c. `/var/a/foo`
 d. `/var/b/foo` e. `/etc/foo`
107. Given this `ls -il` long listing:
`123 drwxr-xr-x 456 me me 789 Jan 1 1:00 dir`
 How many subdirectories lie immediately under **dir**?
 a. 456 b. 787 c. 454 d. 789 e. 123
108. If you are in `/etc` and `ls -l` shows a symbolic link **bar** `-> foo` then dereference the absolute path of **bar** with no symbolic links:
 a. `/etc/foo` b. `/etc/foo/bar` c. `/foo`
 d. `/bar/foo` e. `/etc/bar/foo`
109. If your **PATH** variable contains `/bin:/usr/bin`, what is the output of this:
`echo '$PATH'`
 a. `'$PATH'`
 b. `/bin:/usr/bin`
 c. `'/bin:/usr/bin'`
 d. `$PATH`
 e. `echo: $PATH: No such file or directory`
110. How many files are touched? `touch 1 "2 3" ' 4 ' 5`
 a. 6 b. 7 c. 3 d. 4 e. 5

111. What is the link count of file **f** after these successful commands?
`rm f ; touch f ; ln f bar ; ln bar x`
`cp bar a ; ln a b ; ln x c ; cp c d`
 a. 1 b. 3 c. 5 d. 4 e. 2
112. In an empty directory, what is the output on your screen after this:
`date >.foo >.bar ; ls *`
 a. . . .foo .bar
 b. *
 c. an error message from `ls` saying * does not exist
 d. .foo .bar
 e. no output
113. If directory **/a** contains seven two-character names: **aa, ab, ac, ad, a*, a?, ??**, then which removes *only* the single two-character name **a?** from the directory?
 a. `rm "/a?"` b. `rm /a/a?` c. `rm /a\?`
 d. `rm '/a/a?'` e. `rm /a/?\?`
114. In an empty directory, what is the output on your screen after this:
`touch 1 2 3 ; cow="*" ; echo "$cow"`
 a. 1 2 3 b. * c. "1 2 3"
 d. "\$cow" e. \$cow
115. What is the output of this in an empty directory:
`touch x .a .ab .cde .fghi ; echo .??*`
 a. an error message from `echo` saying .??* does not exist
 b. .??*
 c. .ab .cde .fghi
 d. . . .a .ab .cde .fghi
 e. .cde .fghi
116. File **foo** contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9). What is the output on your screen of this:
`cat foo foo | sort | tail -n 4 | head -n 1`
 a. 4 b. no output c. 8
 d. 1 e. 6
117. In an empty directory, what is the output on your screen after this:
`echo one >.bar ; echo .*`
 a. .*
 b. one
 c. .bar
 d. an error message from `echo` saying .* does not exist
 e. . . .bar
118. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `echo a ; cat b | echo a`
 a. 5 b. 3 c. 2
 d. 1 e. no output

119. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
 111 -rw-r--r-- 2 me me 1 Jan 1 1:00 a
 111 -rw-r--r-- 2 me me 1 Jan 1 1:00 b
 222 -rw-r--r-- 3 me me 1 Jan 1 1:00 c
 222 -rw-r--r-- 3 me me 1 Jan 1 1:00 d
 a. 1 b. 4 c. 3 d. 0 e. 2
120. In an empty directory, how many words are in file **a** after this:
`echo It's redirected >b isn't it\? ; ls >a`
 a. 1 b. 0 c. 3 d. 2 e. 4
121. What is the link count of directory **dir** after these successful commands?
`mkdir dir ; cd dir ; touch foo ; mkdir a b c`
 a. 5 b. 4 c. 2 d. 3 e. 1
122. Which command recursively finds all things with names beginning with **foo**?
 a. `find -name 'foo*'` b. `fgrep -name foo?`
 c. `ls -name foo*` d. `ls foo*`
 e. `find -name foo?`
123. What is the link count of directory **d** after these successful commands?
`mkdir d ; cd d ; touch a ; mkdir b c`
 a. 2 b. 4 c. 5 d. 6 e. 3
124. What is the link count of directory **d** after these successful commands?
`mkdir d ; touch f ; cd d ; ln ../f x`
 a. 5 b. 3 c. 1 d. 4 e. 2
125. In an empty directory, how many arguments are passed to the **rm** command in this:
`touch a a1 a2 ba ca ; rm a*`
 a. none b. 1 c. 4 d. 2 e. 3
126. Which command shows the name of the current computer:
 a. `hostname` b. `history` c. `whois`
 d. `find` e. `comname`
127. What is the link count of directory **a** after these successful commands?
`mkdir a ; mkdir a/b ; mkdir a/c ; mkdir a/b/c`
 a. 1 b. 2 c. 4 d. 5 e. 3
128. How many arguments are passed to the command by the shell:
`$ echo " 1 2 "three ' 4 ' five"6"`
 a. 3 b. 9 c. 1 d. 4 e. 5
129. What is the output on your screen after these command lines:
`echo one >x ; ln x y ; echo two >>y`
`sort x >y ; cat y`
 a. **two** followed by **one** b. **one** followed by **two**
 c. no output d. **two**
 e. **one**

130. How many arguments are passed to the command by the shell:
`$ <cow cow "-x" "-y" "-z" >cow cow`
 a. 4 b. 6 c. 3 d. 7 e. 5
131. What is the link count of directory **d** after these successful commands?
`mkdir d ; mkdir d/a ; touch d/b`
 a. 4 b. 2 c. 3 d. 5 e. 1
132. How many arguments are passed to the command by the shell:
`$ echo 'It's a bird! No! It's a plane!'`
 a. 2 b. 5 c. 3 d. 1 e. 4
133. Which of these statements is true?
 a. Only single quotes are strong enough to stop GLOB patterns from expanding.
 b. If `/x` is an empty directory, `sort /x/*` produces an error message.
 c. Only backslashes are strong enough to stop GLOB patterns from expanding.
 d. Only double quotes are strong enough to stop GLOB patterns from expanding.
 e. If `/y` is an empty directory, `echo /y/*` produces an error message.
134. If directory `/a` contains seven two-character names: **aa**, **ab**, **ac**, **ad**, **a?**, **a***, **a.**, then which removes *only* the single two-character name **a*** from the directory?
 a. `rm /a/a?` b. `rm /a*` c. `rm /a/a*`
 d. `rm /a/a*` e. `rm /a/*`
135. If file **one** occupies one disk block, how many disk blocks are in use after this:
`cp one foo ; ln foo two ; ln two bar ; ln one cow`
 a. 1 b. 5 c. 3 d. 2 e. 4
136. If you are in `/bin` and `ls -l` shows a symbolic link **foo** `-> dir/bar` then dereference the absolute path of **foo** with no symbolic links:
 a. `/bin/foo/dir/bar` b. `/bin/dir/bar`
 c. `/dir/bar` d. `/bin/dir/bar/foo`
 e. `/foo/dir/bar`
137. How many arguments are passed to the command by the shell:
`$ <cow cow "-x" -y "-z" >cow cow`
 a. 3 b. 5 c. 4 d. 2 e. 6
138. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:

```
111 -rw-r--r-- 1 me me 100 Jan 1 1:00 a
222 -rw-r--r-- 1 me me 100 Jan 1 1:00 b
333 -rw-r--r-- 1 me me 100 Jan 1 1:00 c
444 -rw-r--r-- 2 me me 100 Jan 1 1:00 d
```

 a. 0 b. 3 c. 4 d. 2 e. 1
139. In an empty directory, how many arguments are passed to the `cat` command in this: `touch a1 a2 ba ca ; cat a*`
 a. none b. 4 c. 1 d. 3 e. 2

140. What command shows all the lines in file **cow** that contain the string **pig**?
 a. `fgrep pig >cow` b. `cat cow > fgrep pig`
 c. `fgrep pig <cow` d. `fgrep cow pig`
 e. `fgrep cat cow pig`
141. What is the output on your screen after this:
`mkdir dir ; rmdir dir | wc -w`
 a. 1 b. 2 c. no output
 d. 3 e. 0
142. What is the output on your screen after this:
`echo hi >ls ; cat ls > wc`
 a. **hi** b. 1 1 3 c. **ls**
 d. no output e. 1 1 2
143. How many arguments are passed to the command by the shell:
`$ echo "cow "y " bat 'man x' " pig'a "hop' a b`
 a. 6 b. 11 c. 7 d. 5 e. 4
144. If directory **dir** contains four three-character file names: **.aa**, **.ab**, **.a?**, **.a***, then what is the output on your screen of this: `echo dir/???`
 a. no output
 b. `dir/.a?`
 c. `dir/.aa dir/.ab`
 d. `dir/.aa dir/.ab dir/.a? dir/.a*`
 e. `dir/???`
145. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cat b | sort a`
 a. 2 followed by 3 b. 3 followed by 2 c. 4
 d. 3 e. 2
146. In an empty directory, how many arguments are passed to the `rm` command in this:
`date >a1 ; touch a2 ba ca >all ; rm a*`
 a. 2 b. none c. 3 d. 1 e. 4
147. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `sort a b >c ; cat a b c >c`
 a. 10 b. 5 c. 7 d. 6 e. 0
148. If `/bin/bat` is a program that outputs **foo** and `/usr/bin/bat` is a program that outputs **hi** what is the output on your screen after this:
`PATH=/usr:/usr/bin:/bin ; bat`
 a. **hi**
 b. **bash: bat: command not found**
 c. **foo** followed by **hi**
 d. **foo**
 e. **hi** followed by **foo**

149. Rewrite as a simplified absolute path:
`/home/me/../../you/../../../../etc/../../home/me/../../you/../../me/../../foo`
 a. `/home/me/foo` b. `/foo`
 c. `/home/foo` d. `/home/you/foo`
 e. `/etc/foo`
150. If `/bin/bat` is a program that outputs `foo` and `/usr/bin/bat` is a program that outputs `bar` what is the output on your screen after this:
`PATH=/usr:/usr/bin:/bin ; bat`
 a. `foo` followed by `bar`
 b. `bar`
 c. `bash: bat: command not found`
 d. `bar` followed by `foo`
 e. `foo`
151. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
`111 -rw-r--r-- 1 me me 100 Jan 1 1:00 a`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 b`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 c`
`222 -rw-r--r-- 3 me me 100 Jan 1 1:00 d`
 a. 3 b. 1 c. 4 d. 2 e. 0
152. What is true about this output from `ls -il foo bar`
`35 -rw-rw-r-- 2 me me 3 Jan 1 1:00 foo`
`36 -rw-rw-r-- 2 me me 3 Jan 1 1:00 bar`
 a. this output is not possible
 b. `foo` and `bar` are two of three names for this file
 c. `foo` and `bar` are names for the same file
 d. `foo` and `bar` each have two names (four names total)
 e. `foo` and `bar` each have three names (six names total)
153. What is the output on your screen of this:
`echo pig >one ; echo bat | tail one`
 a. an error message b. `pig` followed by `bat`
 c. `bat` d. `bat` followed by `pig`
 e. `pig`
154. How many files are touched? `touch '1 '2 3 '4'" '5`
 a. 4 b. 5 c. 3 d. 2 e. 1
155. Which outputs inode/filename pairs for names in the current directory, sorted by inode number?
 a. `ls -ai | sort -n` b. `ls -node * > sort -n`
 c. `sort -n | ls -ai` d. `ls -i * > sort -n`
 e. `ls /* | sort -node`

156. What is the link count of file `a` after these successful commands?
`ln a d ; cp a f ; ln d c ; ln f g ; ln c e`
 a. 2 b. 1 c. 4 d. 3 e. 5
157. What is the output of this in an empty directory:
`touch 1 .1 23 .23 456 ; echo [12]*`
 a. `1 .1 23 .23`
 b. `1 .1 23 .23 456`
 c. an error message from `echo` saying `[ab]*` does not exist
 d. `1 23`
 e. `[12]*`
158. In an empty directory, how many lines are in file `out` after this:
`ls . .. nosuchfile 2>out`
 a. 4 b. 1 c. 2 d. 3 e. 0
159. In an empty directory, how many arguments are passed to the `wc` command in this:
`date >o1 ; touch a1 b2 out >o1 ; wc o*`
 a. 2 b. 5 c. 3 d. 4 e. 1
160. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this: `sort a b >c ; cat a >>b ; cat c b >c a`
 a. 7 b. 12 c. 8 d. 0 e. 5
161. How many arguments are passed to the command by the shell:
`$ echo 'It's "1 2" isn't it? I can't decide.`
 a. 2 b. 4 c. 6 d. 5 e. 3
162. If `/bin/xxx` is a program that outputs `one` and `/usr/bin/xxx` is a program that outputs `two`, what is the output on your screen after this:
`PATH=/bin/xxx:/usr/bin/xxx:/etc/passwd ; xxx`
 a. `one`
 b. `one` followed by `two`
 c. `two` followed by `one`
 d. `two`
 e. `bash: xxx: command not found`
163. If the file `bat` contained the word `foo`, what is the output on your screen after this: `PATH=/bin/cat:/bin/who:/bin/ls ; cat bat`
 a. `bash: cat: command not found`
 b. `bat`
 c. no output on screen
 d. `foo`
 e. `cat: bat: No such file or directory`
164. How many arguments are passed to the command by the shell:
`$ echo " 1 '2 3' 4 "5 6 ' 7 "8 ' >out`
 a. 6 b. 3 c. 5 d. 2 e. 4

165. In an empty directory, what is the output on your screen after this:
`echo hi >a ; ls >wc -l`
 a. 2 b. no output c. 1
 d. a e. 0
166. If `/bin/foo` is a program that outputs **one** and `/usr/bin/foo` is a program that outputs **two**, what is the output on your screen after this:
`PATH=/dev:/usr/bin:/usr:/bin:/etc ; /bin/foo`
 a. **one** followed by **two**
 b. **bash: /bin/foo: command not found**
 c. **two** followed by **one**
 d. **one**
 e. **two**
167. Which command below is the best way to find a line containing a question mark (?) in the file `/etc/passwd`?
 a. `fgrep /etc/passwd '?'`
 b. `fgrep '?' /etc/passwd`
 c. `search '?' /etc/passwd`
 d. `find '?' /etc/passwd`
 e. `fgrep './?' /etc/passwd`
168. Which command counts lines with two adjacent asterisk characters (**) inside the file?
 a. `fgrep -c ** file` b. `find file -name '**'`
 c. `find -c ** file` d. `find file -name **`
 e. `fgrep -c '**' file`
169. In an empty directory, what is the output on your screen after this:
`ls 2>/dev/null nosuchfile`
 a. `ls: /dev/null: No such file or directory`
 b. `ls: nosuchfile: No such file or directory`
 c. `ls: 2>/dev/null nosuchfile: No such file or directory`
 d. `nosuchfile`
 e. no output
170. What is the link count of directory **d** after these successful commands?
`mkdir d ; cd d ; touch f ; ln f a ; ln f b`
 a. 2 b. 5 c. 3 d. 4 e. 1
171. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `cat a b >c ; sort c a b >c`
 a. 10 b. 7 c. 6 d. 5 e. 0
172. What is the link count of file **f** after these successful commands?
`rm f ; touch f ; ln f bar`
`cp bar x ; ln x y ; ln bar z`
 a. 4 b. 2 c. 1 d. 3 e. 5

173. In an empty directory, what is the output on your screen after this:
`touch A a ; echo * ">*"`
 a. No output b. **A a >A a** c. *** >***
 d. **A a** e. **A a >***
174. File **foo** contains 99 lines, each of which is the two-digit line number of the line in the file (01 through 99). What is the output on your screen of this:
`sort foo foo | tail -n 4 | head -n 1`
 a. 04 04 b. 98 c. 96 96
 d. 96 e. 01 01
175. Which of the following **PATH** statements makes the most sense?
 a. `PATH=/bin:/usr/bin`
 b. `PATH=/bin/bash:/usr/bin:/bin`
 c. `PATH=/bin:/etc/passwd:/usr/bin`
 d. `PATH=/bin:/usr/bin:/etc/passwd`
 e. `PATH=/bin/ls:/etc/passwd:/usr/bin`
176. Which command copies a directory:
 a. `cp dir1 dir2` b. `mv -r dir1 dir2`
 c. `cp -r dir1 dir2` d. `mv -f dir1 dir2`
 e. `mv -rf dir1 dir2`
177. If `/bin/foo` is a program that outputs **mom** and `/usr/bin/foo` is a program that outputs **dad** what is the output on your screen after this:
`PATH=/bin/foo:/usr/bin/foo:/usr ; foo`
 a. **mom** followed by **dad**
 b. **mom**
 c. **dad** followed by **mom**
 d. **bash: foo: command not found**
 e. **dad**
178. In an empty directory, what is the output on your screen after this:
`touch a b .1 .2 ; echo .??*`
 a. an error message from **echo** saying `.??*` does not exist
 b. `.??*`
 c. **a b**
 d. `. . . .1 .2`
 e. `. . . a b .1 .2`
179. What is the output of this in an empty directory:
`touch 1 13 .13 2 213 3 30 39 .31 ; echo [13]?`
 a. `[13]?`
 b. `1 13 3 30 39`
 c. `13`
 d. `13 30 39`
 e. an error message from **echo** saying `[13]?` does not exist

180. How many arguments are passed to the command by the shell:
`$ echo 'And it's not hard, it's just logical.'`
 a. 5 b. 4 c. 3 d. 6 e. 7
181. If your terminal type is `xterm`, what is the output of this: `echo '$TERM'`
 a. no output on screen b. 'xterm'
 c. \$TERM d. '\$TERM'
 e. xterm
182. What is the output on your screen after these command lines:
`echo one >x ; ln x y ; echo ten >y`
`echo two >x ; cat y`
 a. ten b. two
 c. no output on screen d. one
 e. one followed by ten and two
183. In an empty directory, how many lines are in file `foo` after this:
`ls nosuchfile . .. 2>foo`
 a. 2 b. 1 c. 0 d. 3 e. 4
184. If directory `dir` contains five two-character names: `a?`, `11`, `?1`, `1*`, `.1`, then which removes *only* the single two-character name `?1` from the directory?
 a. `rm dir/?1` b. `rm dir/1*` c. `rm dir/\??`
 d. `rm dir/*1` e. `rm dir/??`
185. Which of the following is true, given this long directory listing:
`drwxr-x--x 128 me me 32 Jan 1 1:00 dir`
 a. The number 128 is the inode number of this directory.
 b. The number 32 is the inode number of this directory.
 c. The number 32 is the size of this directory.
 d. The number 128 is the size of this directory.
 e. The number 32 is the count of links (names) this directory has.
186. How many files are touched? `touch '1' '2 3' '4'"' ' 5`
 a. 2 b. 3 c. 1 d. 5 e. 4
187. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this: `ln a e ; ln b d ; ln d c ; cat e b >c`
 a. 0 b. 3 c. 2 d. 4 e. 5
188. Which command removes *only* this five-character name containing a special character: `date?`
 a. `rm ./date\?` b. `rm date*` c. `rm ./date?`
 d. `rm date\\?` e. `rm date/?`
189. What is the output on your screen of this:
`echo pig >one ; echo cow | head -n 2 one`
 a. an error message b. pig
 c. cow followed by pig d. pig followed by cow
 e. cow

190. What is in the file `bar` after this:
`echo hi >x ; echo ho >x ; mv x y >bar`
 a. no such file (nonexistent) b. ho
 c. hi d. hi followed by ho
 e. nothing (empty file)
191. Which command removes *only* this four-character name containing a special character: `?xyz`
 a. `rm '?xyz'` b. `rm ?xyz` c. `rm ''?xyz''`
 d. `rm '?xyz'` e. `rm '?xyz'`
192. How many arguments are passed to the command by the shell:
`$ <wc wc " 1 '2 3' 4 " 5 6 ' 7 " 8 " ' >wc 9`
 a. 3 b. 4 c. 6 d. 5 e. 2
193. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this: `ln a d ; ln b e ; cp d e >c`
 a. 3 b. 0 c. 4 d. 2 e. 5
194. How many arguments are passed to the command by the shell:
`$ <bat bat -b "-a -r" >bat bat bat`
 a. 7 b. 6 c. 3 d. 5 e. 4
195. Your current directory is `dir1`. The parent directory contains another directory, `dir2`. Which command copies file `foo` from the current directory into the `dir2` directory?
 a. `cp foo dir2/..` b. `cp foo dir2`
 c. `cp .. dir2 foo` d. `cp foo dir2 ..`
 e. `cp foo ../dir2`
196. What is the link count of file `f` after these successful commands?
`rm f ; touch f ; cp f x`
`ln f a ; ln x y ; ln a z ; ln z q`
 a. 5 b. 4 c. 6 d. 2 e. 3
197. In an empty directory, what is the output on your screen after this:
`echo hi >foo ; cp foo bar | wc -w`
 a. 2 b. 3 c. 0
 d. 1 e. no output
198. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this: `sort a b >c ; cat a >>b ; sort c b >c a`
 a. 5 b. 0 c. 8 d. 7 e. 12
199. What is the output of this in an empty directory: `cat *`
 a. no output
 b. .
 c. . .
 d. *
 e. an error message from `cat` saying `*` does not exist

200. What is the output on your screen after this:
`echo one >x ; ln x y ; echo two >>y ; sort x`
 a. one
 b. two
 c. one followed by two
 d. two followed by one
 e. no output
201. What is the output of this 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
202. If `mt` is an empty sub-directory, what is true after this:
`touch foo ; mkdir bar ; mv foo bar/mt`
 a. the command fails because `bar/mt` is not a directory
 b. the directory `mt` is still empty
 c. the directory `mt` now contains a directory named `bar`
 d. the directory `mt` now contains a file named `foo`
 e. the directory `bar` now contains a file named `foo`
203. If the current directory contains 10 visible files and 15 visible sub-directories, what is the output on your screen of this: `ls -d */.`
 a. no output
 b. 25 pathnames
 c. 15 directory names
 d. an error message because `*/.` does not exist
 e. `*/.`
204. Which shows the current date?
 a. `echo date | bash`
 b. `bash date`
 c. `bash >date ; cat date`
 d. `date | bash`
 e. `bash <date`
205. What is the link count of file `f` after these successful commands?
`rm f ; touch f ; ln f bar`
`cp bar a ; ln a b ; ln bar c ; cp c a`
 a. 5
 b. 3
 c. 2
 d. 1
 e. 4
206. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:

```
111 -rw-r--r-- 2 me me 100 Jan 1 1:00 a
222 -rw-r--r-- 1 me me 100 Jan 1 1:00 b
333 -rw-r--r-- 2 me me 100 Jan 1 1:00 c
333 -rw-r--r-- 2 me me 100 Jan 1 1:00 d
```

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

207. If your `PATH` contained only the file names `/bin/sh`, `/bin/cat`, and `/bin/ls`, then what is the output on your screen of this: `cat /etc/passwd`
 a. `bash: /bin/cat: no such file or directory`
 b. `bash: /bin/sh: command not found`
 c. `cat: /etc/passwd: command not found`
 d. `bash: cat: command not found`
 e. `cat: bash: no such file or directory`
208. In an empty directory, what is the output on your screen after this:
`ls nosuchfile 2>out`
 a. `nosuchfile`
 b. `2 not found`
 c. `nosuchfile not found`
 d. no output
 e. `nosuchfile 2 not found`
209. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:

```
111 -rw-r--r-- 2 me me 100 Jan 1 1:00 a
222 -rw-r--r-- 2 me me 100 Jan 1 1:00 b
333 -rw-r--r-- 2 me me 100 Jan 1 1:00 c
444 -rw-r--r-- 2 me me 100 Jan 1 1:00 d
```

 a. 0
 b. 2
 c. 4
 d. 3
 e. 1
210. What is the link count of file `f` after these successful commands?
`rm f ; touch f ; ln f a ; ln a b`
`cp f c ; ln c x ; rm b ; mv a b`
 a. 4
 b. 2
 c. 1
 d. 3
 e. 0
211. Which command appends directory `/bin` to your search path?
 a. `$PATH=PATH:/bin`
 b. `PATH=$PATH:/bin`
 c. `PATH=PATH:/bin`
 d. `PATH=PATH+/bin`
 e. `$PATH=$PATH:/bin`
212. How many arguments are passed to the command by the shell:
`$ echo 'It's a bird! It's a plane!'`
 a. 1
 b. 4
 c. 2
 d. 3
 e. 5
213. How many files are touched? `touch 1 "2 3" ' ' 4 5`
 a. 5
 b. 4
 c. 3
 d. 6
 e. 7
214. Given an existing file of yours named `cat`, what is the output on your screen of this: `echo xx >cat ; head cat >cat ; wc cat`
 a. 0 0 0 `cat`
 b. no output
 c. 2 2 4 `cat`
 d. 1 1 3 `cat`
 e. 1 1 2 `cat`

215. If the file **bat** contained the word **foo**, what is the output on your screen after this: `PATH=/etc/passwd:/bin/ls:/bin/cat ; /bin/ls bat`
- `/bin/ls: bat: No such file or directory`
 - `bat`
 - no output on screen
 - `foo`
 - `bash: /bin/ls: command not found`
216. What is the link count of directory **z** after these successful commands?
`mkdir z ; cd z ; touch a b ; mkdir c d e`
- 5
 - 6
 - 4
 - 3
 - 7
217. In an empty directory, what is the output on your screen after this:
`touch 1 2 3 ; cow="*" ; echo "$cow"`
- `"$cow"`
 - `1 2 3`
 - `$cow`
 - `"1 2 3"`
 - `*`
218. How many arguments are passed to the command by the shell:
`$ echo ' one two ' three ' four ' 5'6'`
- 4
 - 1
 - 9
 - 6
 - 5
219. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `cp a c ; cat a b c >c`
- 6
 - 4
 - 5
 - 7
 - 0
220. What is the output on your screen after this:
`echo hi >a ; ls a > wc`
- 3
 - no output
 - 1 1 2
 - 1 1 3
 - 2
221. What is true about this: `date >ls ; ls -ls ls >wc`
- The `ls` command receives the output of `date` on standard input.
 - The shell finds and executes three different commands.
 - The `ls` command is executed more than once.
 - The file `wc` has one line in it.
 - The `wc` command counts the output of the `ls` command.
222. What is true about this output from `ls -il foo bar?`
`15 -rwxrwxrwx 2 bin bin 3 Jul 31 12:33 foo`
`15 -rwxrwxrwx 3 bin bin 3 Jul 31 12:33 bar`
- `foo` and `bar` are names for the same file
 - `foo` and `bar` each have three names (six names total)
 - this output is not possible
 - `foo` and `bar` are two of three names for the same file
 - `foo` and `bar` are names for different files
223. What is the link count of directory **foo** after these successful commands?
`mkdir foo ; cd foo ; touch a b c`
- 2
 - 4
 - 5
 - 3
 - 1

224. What is the output of this in an empty directory:
`touch 1 2 3 .a .ab .abc ; echo [.]*`
- `. . . .a .ab .abc`
 - `.a .ab .abc`
 - `[.]*`
 - no output
 - an error message from `echo` saying `[.]*` does not exist
225. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
`111 -rw-r--r-- 1 me me 1 Jan 1 1:00 a`
`222 -rw-r--r-- 1 me me 1 Jan 1 1:00 b`
`333 -rw-r--r-- 1 me me 1 Jan 1 1:00 c`
`444 -rw-r--r-- 2 me me 1 Jan 1 1:00 d`
- 4
 - 0
 - 3
 - 1
 - 2
226. What is in file **foo** after this: `echo 1 2 >foo 3`
- 1 2
 - 3
 - nothing (empty file)
 - `echo 1 2`
 - 1 2 3
227. What is the possible output on your screen of this:
`echo wc >date ; sort date >date ; cat date`
- no output
 - `wc`
 - `1 6 28 date`
 - `1 6 29 date`
 - `Fri Mar 2 12:00:00 EST 2018`
228. If **mt** is an empty sub-directory, what is true after this:
`touch bar ; mkdir foo ; mv mt/./bar mt/foo`
- the command fails because `mt/foo` is not a directory
 - the directory `foo` now contains a file named `bar`
 - the directory `mt` now contains a file named `bar`
 - the directory `mt` now contains a file named `foo`
 - the directory `mt` is still empty
229. If file **foo** occupies one disk block, how many disk blocks are in use after this:
`cp foo bar ; ln bar one ; cp one two ; ln one pig`
- 1
 - 5
 - 2
 - 4
 - 3
230. Which of the following commands will leave **file1** non-empty?
- `tail file1 > file1`
 - `cat file1 > file1`
 - `wc file1 > file1`
 - `head file1 > file1`
 - `sort file1 > file1`

231. What is in the local variable `$$` ?
- the first argument of the previous command line
 - `$$` is not a valid variable name
 - the process ID of the current shell
 - the cpu cost of the current session, in dollars
 - the command name of the previous command line
232. What is in file `c` after this:
- ```
echo A >a ; ln a b ; echo B >b ; ln a c ; rm a b
```
- `A` followed by `B`
  - nothing (empty file)
  - `B`
  - no such file (nonexistent)
  - `A`
233. Which always prints just the two characters `$x` on the screen?
- `echo $$x`
  - `echo "$$x"`
  - `echo $x`
  - `echo "$x"`
  - `echo '$x'`
234. What is the link count of directory `dir` after these successful commands?
- ```
mkdir dir ; touch foo ; cd dir ; ln ../foo bar
```
- 2
 - 4
 - 3
 - 1
 - 5
235. If file `nine` contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9), what is the output on your screen of this:
- ```
sort nine nine | tail -n 3 | head -n 1
```
- 9
  - 2 2
  - 1
  - 8 8
  - 8
236. In an empty directory, what is the output on your screen after this:
- ```
echo one >.bar ; echo .*
```
- `.bar`
 - an error message from `echo` saying `.*` does not exist
 - `.*`
 - `one`
 - `.. .bar`
237. How many words are in the file `x` after this:
- ```
echo 1 2 >x ; echo 3 >x ; echo 4 >>x
```
- 2
  - 4
  - 1
  - 3
  - 0
238. What is the output on your screen of this unquoted command line:
- ```
mkdir a ; touch b a/b1 a/b2 ; find a -name b*
```
- `b a/b1 a/b2`
 - `a/b1 a/b2`
 - `b1 b2`
 - no output
 - `b`
239. What is the link count of file `f` after these successful commands?
- ```
cp f x ; ln f a ; ln x y ; ln a z ; ln a b
```
- 5
  - 4
  - 3
  - 2
  - 6
240. In an empty directory, how many words are in file `cow` after this:
- ```
touch dog dog cat ; ls >cow
```
- 1
 - 0
 - 2
 - 4
 - 3

241. How many arguments are passed to the command by the shell:
- ```
$ echo " one '2 three' 4 "five 6 ' 7 "8 ' >out
```
- 6
  - 3
  - 4
  - 5
  - 2
242. How many arguments are passed to the command by the shell:
- ```
$ echo " 1 2 " three ' 4 ' five"6"
```
- 3
 - 4
 - 5
 - 9
 - 1
243. What is the link count of directory `z` after these successful commands?
- ```
mkdir z ; mkdir z/a z/a/b z/a/c z/a/d
```
- 3
  - 5
  - 2
  - 1
  - 4
244. What is the output on your screen of this:
- ```
echo wc >wc ; wc wc >wc ; cat wc
```
- 0 0 0 wc
 - no output
 - 1 1 3 wc
 - 1 1 2 wc
 - wc
245. If I am in directory `/tmp` and `mt` is an empty sub-directory, what is true after this:
- ```
touch mt/bar ; mkdir mt/me ; cp mt/bar mt/../me
```
- the directory `mt` now contains only a file named `me`
  - the command fails because the name `mt/../me` does not exist
  - the directory `mt` is now empty
  - there is a second copy of the file `bar` in file `/tmp/me`
  - there is a second copy of the file `bar` in directory `mt`
246. In an empty directory, what is the output on your screen after this:
- ```
echo one >.bar ; ls .????*
```
- `.bar`
 - an error message from `ls` saying `????*` does not exist
 - `????*`
 - `one`
 - `.. .bar`
247. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this:
- ```
cp a b >z ; cp a b >a ; sort a b z >c
```
- 8
  - 9
  - 0
  - 6
  - 7
248. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `out` after this:
- ```
cat a b >c ; head c >c ; sort a b c >out
```
- 6
 - 5
 - 3
 - 0
 - 10
249. In an empty directory, what is the output on your screen after this:
- ```
echo one >.bar ; echo .????*
```
- `????*`
  - `.. .bar`
  - `.bar`
  - an error message from `echo` saying `????*` does not exist
  - `one`





266. How do you execute the program **foo** in the current directory?  
 a. `./foo`                      b. `/foo`                      c. `foo/`  
 d. `foo/.`                      e. `$HOME/foo`
267. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `sort a b >c ; cat a >>b ; sort c b a >c`  
 a. 8                      b. 7                      c. 12                      d. 5                      e. 0
268. What is the output on your screen after these command lines:  
`echo 1 >x ; ln x y ; echo 2 >>y`  
`head -1 x >y ; cat y`  
 a. 2                      b. 1                      c. 1 followed by 2  
 d. no output                      e. 2 followed by 1
269. How many arguments are passed to the command by the shell:  
`$ <foo foo -x " " -z -r" " >foo 'foo foo'`  
 a. 9                      b. 5                      c. 7                      d. 6                      e. 8
270. How many files are touched? `touch "1" 2 3 " " 4 5`  
 a. 5                      b. 4                      c. 6                      d. 3                      e. 7
271. How many files are touched? `touch 1 "2 3 ' 4 '" 5`  
 a. 4                      b. 5                      c. 3                      d. 1                      e. 2
272. In an empty directory, how many arguments are passed to the **wc** command in this:  
`touch xx yy >zz 123 .a b. ; wc ??`  
 a. 2                      b. 5                      c. 0                      d. 1                      e. 4
273. In an empty directory, how many words are in file **c** after this:  
`touch a b 1 b a ; ls >c`  
 a. 0                      b. 1                      c. 4                      d. 3                      e. 2
274. What is the output of this in an empty directory:  
`touch .a .b .c ; echo [.]*`  
 a. `. . . .a .b .c`  
 b. an error message from **echo** saying `[.]*` does not exist  
 c. no output  
 d. `[.]*`  
 e. `.a .b .c`
275. If **/bin/xxx** is a program that outputs **one** and **/usr/bin/xxx** is a program that outputs **two**, what is the output on your screen after this:  
`PATH=/etc:/usr/bin:/usr:/bin ; /bin/xxx`  
 a. **one** followed by **two**  
 b. **bash: /bin/xxx: command not found**  
 c. **two** followed by **one**  
 d. **one**  
 e. **two**

276. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `echo a ; echo b`  
 a. 2                      b. 3                      c. no output  
 d. 5                      e. 1
277. Which makes a file executable?  
 a. `chmod -x file`                      b. `umask u+x file`  
 c. `chmod u+x file`                      d. `umask -x file`  
 e. `umask u=x file`
278. How many words are in file **out** after this:  
`echo one >two >three >out`  
 a. 0                      b. 4                      c. 3                      d. 2                      e. 1
279. Which file is a DOS/Windows file?  
 a. ASCII text, with CRLF line terminators  
 b. ASCII text, with no line terminators  
 c. ASCII text  
 d. ASCII text, with CR line terminators  
 e. ASCII text, with LF line terminators
280. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:  
`111 -rw-r--r-- 1 me me 100 Jan 1 1:00 a`  
`222 -rw-r--r-- 1 me me 100 Jan 1 1:00 b`  
`444 -rw-r--r-- 2 me me 100 Jan 1 1:00 c`  
`444 -rw-r--r-- 2 me me 100 Jan 1 1:00 d`  
 a. 1                      b. 3                      c. 0                      d. 2                      e. 4
281. If file **foo** occupies one disk block, how many disk blocks are in use after this:  
`cp foo bar ; ln bar one ; cp one two ; cp one xxx`  
 a. 4                      b. 5                      c. 1                      d. 2                      e. 3
282. What is the link count of directory **z** after these successful commands?  
`mkdir z ; mkdir z/a ; touch z/b z/c z/d`  
 a. 3                      b. 4                      c. 2                      d. 1                      e. 5
283. In an empty directory, how many words are in file **c** after this:  
`touch a ; mv b a >b ; ls >c`  
 a. 2                      b. 4                      c. 3                      d. 1                      e. 0
284. In an empty directory, what is the output on your screen after this:  
`touch A a ; echo * >"*" ; ls`  
 a. No output                      b. `A a >A a`                      c. `A a >*`  
 d. `* >*`                      e. `* A a`

285. Which of the following statements is true about this:  
`$ <dir/c cat dir/d`  
 a. The command `cat` sees two arguments.  
 b. The command `dir/c` sees two arguments.  
 c. The command `dir/c` sees only one argument.  
 d. The command `cat` sees only one argument.  
 e. The command is always invalid.
286. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `d` (not in `c`) after this:  
`ln a d ; ln d c ; ln c e ; cat a a b b c c d d e e >c`  
 a. 2                      b. 18                      c. 10                      d. 6                      e. 21
287. How many arguments are passed to the command by the shell:  
`$ <f z " a 'b c' d " 1 2 ' g " h " ' >z`  
 a. 3                      b. 6                      c. 2                      d. 4                      e. 5
288. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this: `sort a b >z ; tail a >a ; sort a b z >c`  
 a. 6                      b. 8                      c. 10                      d. 5                      e. 0
289. How many lines are in file `out` after this: `echo hi >dog >out >cat`  
 a. 1                      b. 2                      c. 4                      d. 3                      e. 0
290. What is the link count of file `f` after these successful commands?  
`rm f ; touch f ; ln f bar`  
`cp bar x ; ln x y ; ln bar z ; ln z a`  
 a. 1                      b. 2                      c. 3                      d. 4                      e. 5
291. What is the output on your screen after this:  
`echo 1 >x ; ln x y ; echo 2 >>y ; sort x`  
 a. 2 followed by 1                      b. 1                      c. no output  
 d. 2                      e. 1 followed by 2
292. In an empty directory, what is the output on your screen after this:  
`echo hi >a ; ls | wc -w`  
 a. 2                      b. 0                      c. 1  
 d. no output                      e. a
293. Which outputs only lines 11-15 of the Unix password file?  
 a. `head -n 10 /etc/passwd | tail -n 15 /etc/passwd`  
 b. `tail -n 10 /etc/passwd | head -n 15 /etc/passwd`  
 c. `tail -n 15 /etc/passwd | head -n 10`  
 d. `head -n 15 /etc/passwd | tail -n 5 /etc/passwd`  
 e. `head -n 15 /etc/passwd | tail -n 5`
294. In an empty directory, how many words are in file `foo` after this:  
`date >.bar >.out ; ls >foo`  
 a. 1                      b. 0                      c. 3                      d. 2                      e. 4

295. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are output on your screen by this: `cat a a b | date`  
 a. 6                      b. 1                      c. 5                      d. 7                      e. 8
296. If `/bin/foo` is a program that outputs `mom` and `/usr/bin/foo` is a program that outputs `dad`, what is the output on your screen after this:  
`PATH=/dev:/usr/bin:/usr:/bin:/etc ; /bin/foo`  
 a. `dad` followed by `mom`  
 b. `mom`  
 c. `mom` followed by `dad`  
 d. `dad`  
 e. `bash: /bin/foo: command not found`
297. What is the output on your screen of this:  
`echo hi >hi ; head hi >hi ; wc hi`  
 a. 1 1 2 hi                      b. 0 0 0 hi                      c. no output  
 d. 2 2 4 hi                      e. 1 1 3 hi
298. What is true about this output from `ls -il foo bar`  
`15 -r-x----- 2 me me 3 Jan 1 1:00 foo`  
`15 -rwxrwxrwx 2 me me 3 Jan 1 1:00 bar`  
 a. this output is not possible  
 b. `foo` and `bar` are names for different files  
 c. `foo` and `bar` each have three names (six names total)  
 d. `foo` and `bar` are two of three names for the same file  
 e. `foo` and `bar` are names for the same file
299. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are output on your screen by this: `cat a b | pwd`  
 a. 3                      b. no output                      c. 5  
 d. 6                      e. 1
300. What is usually in the environment variable `$HOME`?  
 a. the absolute path of your login home directory  
 b. the relative path of the `ROOT` directory  
 c. the absolute path of the system `/home` directory  
 d. the relative path of the system `/home` directory  
 e. the relative path of your login home directory
301. In an empty directory, how many arguments are passed to the `cat` command in this: `date >a1 ; touch a2 ba ca ; cat a*`  
 a. 1                      b. none                      c. 2                      d. 4                      e. 3

302. Which of these will make file **foo** contain all of the content of file **a** followed by all of the content of file **b**?
- `echo a b >foo`
  - `cp a b >foo`
  - `mv a b >foo`
  - `cp a >foo ; cp b >>foo`
  - `cat a >foo ; cat b >>foo`
303. If `/bin/foo` is a program that outputs **dad** and `/usr/bin/foo` is a program that outputs **mom** what is the output on your screen after this:
- ```
PATH=/usr:/etc:/bin:/usr/bin ; foo
```
- mom**
 - bash: foo: command not found**
 - mom** followed by **dad**
 - dad** followed by **mom**
 - dad**
304. What is the output of this in an empty directory:
- ```
date >.date ; users >.users ; echo .?*
```
- .date .users**
  - .date**
  - an error message from **echo** saying **.?\* does not exist**
  - .. .date .users**
  - .?\***
305. Which command shows names under directory **oldnotes** containing **RTFM** anywhere in the name?
- `ls oldnotes*RTFM*`
  - `ls oldnotes/*RTFM*`
  - `ls oldnotes/RTFM*`
  - `ls oldnotes *RTFM*`
  - `ls oldnotes RTFM *`
306. Rewrite as a simplified absolute path:
- ```
/usr/./bin/./lib/./../etc/./usr/./lib/./bin/./bar
```
- `/bar`
 - `/usr/bar`
 - `/etc/bar`
 - `/usr/bin/bar`
 - `/usr/lib/bar`
307. What is the link count of directory **x** after these successful commands?
- ```
mkdir x ; mkdir x/y ; mkdir x/z ; mkdir x/y/z
```
- 2
  - 5
  - 3
  - 4
  - 1
308. Which would show the index (inode) number of a file?
- `find -i file`
  - `cat -l file`
  - `ls -i file`
  - `cat -i file`
  - `ls -l file`
309. If directory `/a` contains seven two-character names: **aa**, **ab**, **ac**, **ad**, **a?**, **a\***, **a..**, then which removes *only* the single two-character name **a?** from the directory?
- `rm /a/a\?`
  - `rm /a/a?`
  - `rm /a?`
  - `rm /a/a[*]`
  - `rm /a/a*`

310. In an empty directory, what is the output on your screen after this:
- ```
touch a ; ls | wc -w
```
- 1
 - 0
 - 2
 - 3
 - no output
311. If files occupy one disk block, how many disk blocks will the system free up if I remove these four file names:
- ```
111 -rw-r--r-- 1 me me 100 Jan 1 1:00 a
222 -rw-r--r-- 1 me me 100 Jan 1 1:00 b
333 -rw-r--r-- 1 me me 100 Jan 1 1:00 c
444 -rw-r--r-- 1 me me 100 Jan 1 1:00 d
```
- 2
  - 0
  - 1
  - 4
  - 3
312. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `ln a e ; ln b d ; ln d c ; cp d e ; sort a b e d >c`
- 4
  - 7
  - 10
  - 6
  - 12
313. What is the link count of file **f** after these successful commands?
- ```
rm f ; touch f ; ln f b ; cp f c
cp b x ; ln x y ; ln b z ; ln z a
```
- 3
 - 4
 - 5
 - 1
 - 2
314. What is the link count of file **f** after these successful commands?
- ```
rm f ; touch f ; cp f x
ln -s f a ; ln x y ; ln a z ; ln x b
```
- 2
  - 4
  - 3
  - 5
  - 1
315. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `sort a | echo b`
- 3 followed by 2
  - 2 followed by 3
  - 4
  - 2 followed by 1
  - 1
316. To change to the parent directory, do this:
- `cd`
  - `pwd`
  - `cd .`
  - `cd ..`
  - `pwd ..`
317. Which command finds your account login userid in the password file?
- `fgrep $USER /etc/passwd`
  - `fgrep /etc/passwd $USER`
  - `find /etc/passwd -name $USER`
  - `find $USER /etc/passwd`
  - `cat $USER /etc/passwd`
318. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `sort a b | cat a | cat`
- 0
  - 4
  - 6
  - 7
  - 2
319. File **foo** contains 9 lines, each of which is the one-digit line number of the line in the file (1 through 9). What is the output on your screen of this:
- ```
sort foo foo | tail -n 4 | head -n 1
```
- 8
 - 4 4
 - 6
 - 6 6
 - 1 1

320. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cat a | sort b`
 a. 4 b. 2 c. 0 d. 3 e. 5
321. If you are in `/etc` and `ls -l` shows a symbolic link `bar -> ../foo` then dereference the absolute path of `bar` with no symbolic links:
 a. `/foo` b. `/etc/foo` c. `/etc/bar/foo`
 d. `/bar/foo` e. `/etc/foo/bar`
322. What is the output on your screen after this:
`echo hi >a ; cp a b | wc -w`
 a. 2 b. 0 c. 1
 d. 3 e. no output
323. What is in file `foo` after this:
`echo hi >a ; ln a b ; echo me >b ; ln a foo ; rm a b`
 a. `hi` followed by `me` b. nothing (empty file)
 c. no such file (nonexistent) d. `me`
 e. `hi`
324. If directory `/a` contains seven two-character names: `aa`, `ab`, `ac`, `ad`, `a?`, `a*`, `a.`, then which removes *only* the single two-character name `a*` from the directory?
 a. `rm /a*` b. `rm /a/a*` c. `rm /a/a?`
 d. `rm /a/*` e. `rm "/a/a*"`
325. What is the output on your screen of this:
`echo bat >pig ; echo one | tail pig`
 a. `one` followed by `bat` b. `one`
 c. an error message d. `bat`
 e. `bat` followed by `one`
326. If `/bin/prg` is a program that outputs `hi` and `/usr/bin/prg` is a program that outputs `foo` what is the output on your screen after this:
`PATH=/etc:/usr/bin:/bin ; prg`
 a. `bash: prg: command not found`
 b. `foo` followed by `hi`
 c. `foo`
 d. `hi`
 e. `hi` followed by `foo`
327. What is usually in the environment variable `$PATH`?
 a. a colon-separated list of your `passwd` file fields
 b. a colon-separated list of directories containing command names
 c. the absolute path of your login home directory
 d. the absolute path of the system `/path` directory
 e. the absolute path of your login shell

328. If `foo` were a readable empty file, what is the output on your screen after this:
`PATH=/etc/passwd:/bin/ls:/bin/cat ; /bin/cat foo`
 a. `bash: /bin/cat: command not found`
 b. no output on screen
 c. `/bin/cat: foo: No such file or directory`
 d. `bash: cat: command not found`
 e. `bash: ls: command not found`
329. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **e** after this: `ln a d ; cp a f ; ln d c ; ln c e ; cat a b d f >e`
 a. 5 b. 6 c. 3 d. 2 e. 9
330. How many arguments are passed to the command by the shell:
`$ <bar bar -b"-a '-r' >bar" bar >out`
 a. 2 b. 5 c. 4 d. 6 e. 3
331. If `/bin/pig` is a program that outputs `xx` and `/usr/bin/pig` is a program that outputs `foo` what is the output on your screen after this:
`PATH=/home:/bin:/dev:/usr/bin ; pig`
 a. `xx` followed by `foo`
 b. `foo`
 c. `foo` followed by `xx`
 d. `xx`
 e. `bash: pig: command not found`
332. What displays on your screen given this command:
`ls >ls ; wc ls >wc ; sort ls | cat wc`
 a. only the `wc` displays because `cat` ignores the pipe
 b. `sort` displays the `ls` and `cat` displays the `wc`
 c. only the `ls` displays because `cat` ignores the pipe
 d. `cat` reads the pipe and the `wc` and displays both together
 e. nothing displays because `cat` ignores the pipe
333. In an empty directory, how many words are in file `pig` after this:
`touch pig pig ; ls >pig`
 a. 3 b. 4 c. 0 d. 2 e. 1
334. Which allows programs in the current directory to execute without preceding the names with `./?` (P.S. Security Risk! Don't do this!)
 a. `PATH=/usr/bin/./:$HOME` b. `$PATH=/usr/bin:./bin`
 c. `$PATH=./:$HOME:/usr/bin` d. `PATH=/bin:/usr/bin:.`
 e. `PATH=./$HOME:/usr/bin`
335. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: `cat a | echo b ; echo a`
 a. 5 b. 7 c. 1 d. 3 e. 2

336. What is in file **c** after this:
`echo foo >a ; ln a b ; echo bar >>b ; ln a c ; rm a`
 a. **foo** followed by **bar** b. **foo**
 c. nothing (empty file) d. **bar**
 e. no such file (nonexistent)
337. The option to **ls** that shows inode (index) numbers is:
 a. **-a** b. **-i** c. **-l** d. **-x** e. **-1**
338. Which displays only the names in the current directory that are exactly three digits long (and no other names)?
 a. `echo [?][?][?]` b. `echo [0-9][0-9][0-9]`
 c. `echo [3][3][3]` d. `echo ???`
 e. `echo [1-3][1-3][1-3]`
339. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **c** after this: `ln a d ; ln d e ; ln b f >c`
 a. **2** b. **0** c. **4** d. **3** e. **5**
340. Which command usually goes in your **.bash_profile** file?
 a. `cat .bashrc` b. `source ./bashrc`
 c. `.bashrc source` d. `.bash_profile source`
 e. `source ./bash_profile`
341. In an empty directory, what is the output on your screen after this:
`echo one >.bar ; echo .??*`
 a. **.??***
 b. **.bar**
 c. **one**
 d. an error message from **echo** saying **.??*** does not exist
 e. **.. .bar**
342. Which command pipeline outputs the count of the number of manual page titles that contain the keyword "sort"?
 a. `wc -k sort` b. `man -k sort | wc`
 c. `man sort | wc` d. `man sort ; wc`
 e. `wc man sort`
343. If directory **dir** contains three four-character file names: **.123**, **.124**, **.???**, then what is the output on your screen of this: `echo dir/????`
 a. **dir/.123 dir/.124 dir/.???**
 b. **dir/????**
 c. **dir/.123 dir/.124**
 d. **echo: dir/????: No such file or directory**
 e. no output

344. In an empty directory, what is the output on your screen of this:
`echo hi >foo >bar ; cat foo`
 a. **cat: foo: No such file or directory**
 b. **hi**
 c. no output
 d. **hi >foo >bar**
 e. **hi >foo**
345. What is usually in the environment variable **\$SHELL**?
 a. the absolute path of the system **/shell** directory
 b. the relative path of the **/home/shell** directory
 c. the relative path of the system **/shell** directory
 d. the relative path of your login shell
 e. the absolute path of your login shell
346. In an empty directory, what is the output on your screen after this:
`touch 1 2 3 ; cow="*" ; echo '$cow'`
 a. **'1 2 3'** b. **\$cow** c. **'\$cow'**
 d. ***** e. **1 2 3**
347. **Did you read all the words of the test instructions on page one?**
 a. **Taip** (Yes - Lithuanian) b. **Tak** (Yes - Polish)
 c. **Igen** (Yes - Hungarian) d. **Sim** (Yes - Portuguese)
 e. **Jes** (Yes - Esperanto)