

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 **E**
192. Answer **192** is **C**
193. Answer **193** is **B**
194. Answer **194** is **A**
195. Answer **195** is **B**
196. Answer **196** is **E**

Your **Test Version** is:**E C B A B E**

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. **Taip** (Yes - Lithuanian)
 - b. **Jes** (Yes - Esperanto)
 - c. **Sim** (Yes - Portuguese)
 - d. **Igen** (Yes - Hungarian)
 - e. **Tak** (Yes - Polish)
2. **My three-digit Lab Section number is:**
 - a. My lecture Section Number, i.e. **010, 020**
 - b. My lab room number, e.g. **P210, P213, B119**
 - c. The Test Version code printed on the question sheet.
 - d. My lecture room number, i.e. **C346, T119**
 - e. The timetable Section Number of my weekly 2-hour lab period.
3. 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/dir/bar/foo**
 - b. **/foo/dir/bar**
 - c. **/bin/foo/dir/bar**
 - d. **/dir/bar**
 - e. **/bin/dir/bar**
4. 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 command fails because the name **mt/./bar/me** does not exist
 - c. there is a second copy of the file **bar** in the file named **me**
 - d. the **mkdir** fails because **bar** already exists
 - e. the directory **mt** now contains only a file named **me**

5. Which command counts lines with two adjacent asterisk characters (******) inside the file?
 - a. **find file -name ****
 - b. **fgrep -c '**' file**
 - c. **fgrep -c ** file**
 - d. **find -c ** file**
 - e. **find file -name '**'**
6. 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. **3**
 - b. **3** followed by **2**
 - c. **2** followed by **3**
 - d. **2**
 - e. no output
7. 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 size of this directory.
 - b. The number 32 is the size of this directory.
 - c. The number 32 is the count of links (names) this directory has.
 - d. The number 32 is the inode number of this directory.
 - e. The number 128 is the inode number of this directory.
8. How many arguments are passed to the command by the shell:


```
$ <bat bat -b "-a -r" >bat bat bat
```

 - a. **7**
 - b. **6**
 - c. **5**
 - d. **3**
 - e. **4**
9. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are output on your screen by this: **cat a | echo b**
 - a. **5**
 - b. no output
 - c. **2**
 - d. **1**
 - e. **3**
10. 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. **A**
 - c. **A** followed by **B**
 - d. no such file (nonexistent)
 - e. nothing (empty file)
11. 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. **2**
 - b. **5**
 - c. **3**
 - d. **4**
 - e. **1**
12. 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/.aa dir/.ab**
 - c. **dir/???**
 - d. **dir/.aa dir/.ab dir/.a? dir/.a***
 - e. **dir/.a?**
13. 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
```

 - a. **6**
 - b. **6 6**
 - c. **8**
 - d. **1 1**
 - e. **4 4**

14. 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. no output c. 3
 d. 1 e. 2
15. What command will recursively show disk usage in directories?
 a. **du** b. **df** c. **find** d. **tree** e. **ls**
16. 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`
 a. 9 b. 7 c. 6 d. 0 e. 8
17. In an empty directory, what is the output on your screen after this:
`touch A a ; echo * >"*" ; ls`
 a. **A a >A a** b. No output c. *** >***
 d. **A a >*** e. *** A a**
18. In an empty directory, how many arguments are passed to the **cat** command in this: `touch a1 a2 ba ca ; cat a*`
 a. 2 b. 1 c. 4 d. 3 e. none
19. 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?"** e. **rm /a\?**
20. In an empty directory, how many words are in file **pig** after this:
`touch pig pig ; ls >pig`
 a. 1 b. 3 c. 2 d. 4 e. 0
21. 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. **01 01** b. **98** c. **04 04**
 d. **96** e. **96 96**
22. 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. 6 b. 2 c. 10 d. 18 e. 21
23. 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/***
 d. **rm /a/a*** e. **rm /a/a?**
24. Which tells you the recursive count of all pathnames under the current directory and all subdirectories?
 a. **ls | wc** b. **wc .** c. **wc ***
 d. **find | wc** e. **wc "\$PWD"**

25. 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. 0 b. 5 c. 4 d. 2 e. 3
26. 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 .1 .2**
 d. **. .. .1 .2**
 e. **a b**
27. Which can generate a non-empty file?
 a. **cat foo >foo** b. **sort foo >foo**
 c. **ls foo >foo** d. **tail foo >foo**
 e. **fgrep 'foo' foo >foo**
28. In an empty directory, what is the output on your screen after this:
`ls 1>/dev/null nosuchfile`
 a. **ls: /dev/null: No such file or directory**
 b. **nosuchfile**
 c. no output
 d. **ls: nosuchfile: No such file or directory**
 e. **ls: 1>/dev/null nosuchfile: No such file or directory**
29. How many arguments are passed to the command by the shell:
`$ echo " 1 '2 3' 4 "5 6 ' 7 "8 ' >out`
 a. 5 b. 4 c. 2 d. 6 e. 3
30. What displays on your screen given this command:
`date >date ; pwd >pwd ; head date | tail pwd`
 a. only the **date** displays because **tail** ignores the pipe
 b. only the **pwd** displays because **tail** ignores the pipe
 c. nothing displays because **tail** ignores the pipe
 d. **head** displays the **date** and **tail** displays the **pwd**
 e. **tail** reads the pipe and the **pwd** and displays both together
31. 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`
 a. 6 b. 10 c. 5 d. 0 e. 3
32. 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`
 a. there is a second copy of the file **bar** in file **/tmp/me**
 b. there is a second copy of the file **bar** in directory **mt**
 c. the directory **mt** is now empty
 d. the directory **mt** now contains only a file named **me**
 e. the command fails because the name **mt/./me** does not exist

33. 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. 02 e. 96
34. In a directory containing one file named **mt**, what is the output on your screen after this: `ls 2>/dev/null nosuchfile`
 a. **bash: 2>/dev/null: command not found**
 b. no output
 c. **nosuchfile**
 d. **ls: nosuchfile: No such file or directory**
 e. **mt**
35. In an empty directory, how many arguments are passed to the **rm** command in this:
`touch a a1 a2 ba ca ; rm a*`
 a. 2 b. 3 c. none d. 4 e. 1
36. Which of the following **PATH** statements makes the most sense?
 a. **PATH=/bin/bash:/usr/bin:/bin**
 b. **PATH=/bin:/etc/passwd:/usr/bin**
 c. **PATH=/bin:/usr/bin:/etc/passwd**
 d. **PATH=/bin:/usr/bin**
 e. **PATH=/bin/ls:/etc/passwd:/usr/bin**
37. What is the output on your screen after this:
`mkdir dir ; touch dir/.aa dir/.bb ; echo dir/*`
 a. **dir/***
 b. **dir/ dir/.. dir/.aa dir/.bb**
 c. **dir/.aa dir/.bb**
 d. **dir/**
 e. no output
38. How many files are touched? `touch 1 "2 3 ' 4 '" 5`
 a. 2 b. 1 c. 5 d. 4 e. 3
39. What is the output of this in an empty directory: `cat *`
 a. **.**
 b. no output
 c. *****
 d. an error message from **cat** saying ***** does not exist
 e. **. . .**
40. 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. 3 b. 4 c. 5 d. 1 e. 2

41. How many arguments are passed to the command by the shell:
`$ <foo foo " a 'b c' d " e ' f " g " ' >foo h`
 a. 6 b. 5 c. 4 d. 3 e. 2
42. How do you execute the program **foo** in the current directory?
 a. **/foo** b. **\$HOME/foo** c. **foo/**
 d. **./foo** e. **foo/.**
43. 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}'**
44. 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. 6 b. no output c. 5
 d. 3 e. 1
45. Which of these will make file **foo** contain all of the content of file **a** followed by all of the content of file **b**?
 a. **cp a b >foo**
 b. **mv a b >foo**
 c. **echo a b >foo**
 d. **cat a >foo ; cat b >>foo**
 e. **cp a >foo ; cp b >>foo**
46. In an empty directory, what is the output on your screen after this:
`echo hi >a ; ls | wc -w`
 a. 1 b. 2 c. 0
 d. a e. no output
47. What is the output of this in an empty directory:
`touch x .a .ab .cde .fghi ; echo .??*`
 a. **.cde .fghi**
 b. **. . . .a .ab .cde .fghi**
 c. **.??***
 d. **.ab .cde .fghi**
 e. an error message from **echo** saying **.??*** does not exist
48. What is the output on your screen after this:
`echo hi >ls ; cat ls > wc`
 a. **ls** b. no output c. 1 1 2
 d. 1 1 3 e. hi
49. 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`
 a. 7 b. 6 c. 2 d. 0 e. 4

50. 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`
- `mom`
 - `dad`
 - `dad` followed by `mom`
 - `bash: /bin/foo: command not found`
 - `mom` followed by `dad`
51. In an empty directory, what is the output on your screen after this:
`touch a ; ls | wc -w`
- 3
 - no output
 - 0
 - 1
 - 2
52. In an empty directory, how many arguments are passed to the `wc` command in this:
`touch xx yy >zz 123 .a b. ; wc ??`
- 5
 - 4
 - 2
 - 1
 - 0
53. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are output on your screen by this: `echo a ; echo b`
- no output
 - 2
 - 5
 - 3
 - 1
54. Which command recursively finds all things with names beginning with `foo`?
- `ls -name foo*`
 - `ls foo*`
 - `fgrep -name foo?`
 - `find -name foo?`
 - `find -name 'foo*'`
55. What is the resulting link count of empty directory `dir` after these successful commands? `cd dir ; touch foo ; ln foo one ; ln foo two`
- 5
 - 2
 - 3
 - 1
 - 4
56. What is the output on your screen after this:
`echo 1 >x ; ln x y ; echo 2 >>y ; sort x`
- no output
 - 1
 - 1 followed by 2
 - 2 followed by 1
 - 2
57. If your terminal type is `xterm`, what is the output of this: `echo '$TERM'`
- `$TERM`
 - `xterm`
 - no output on screen
 - `'$TERM'`
 - `'xterm'`
58. Which command pipeline outputs the count of the number of manual page titles that contain the keyword "sort"?
- `wc man sort`
 - `man sort | wc`
 - `wc -k sort`
 - `man sort ; wc`
 - `man -k sort | wc`

59. 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`
- `one` followed by `ten` and `two`
 - `one`
 - `two`
 - no output on screen
 - `ten`
60. 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`
- 5
 - 0
 - 7
 - 8
 - 12
61. In an empty directory, how many words are in file `a` after this:
`echo It's redirected >b isn't it\? ; ls >a`
- 0
 - 3
 - 2
 - 4
 - 1
62. How many files are touched? `touch '1 '2 3 '4'" '5`
- 5
 - 4
 - 3
 - 2
 - 1
63. How many arguments are passed to the command by the shell:
`$ echo 'And it's not hard, it's just logical.'`
- 4
 - 7
 - 5
 - 6
 - 3
64. If your `PATH` variable contains `/bin:/usr/bin`, what is the output of this:
`echo '$PATH'`
- `echo: $PATH: No such file or directory`
 - `'/bin:/usr/bin'`
 - `$PATH`
 - `/bin:/usr/bin`
 - `'$PATH'`
65. What is the link count of directory `z` after these successful commands?
`mkdir z ; mkdir z/a ; touch z/b z/c z/d`
- 5
 - 2
 - 3
 - 4
 - 1
66. 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 5 | head -n 1`
- 1 1
 - 9
 - 5 5
 - 1
 - 7
67. In an empty directory, what is in file `count` after this:
`ls ??? | wc -w >count`
- nothing (empty file)
 - 1 1 1
 - 1
 - 0
 - 1 1 2
68. 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`
- 1
 - 5
 - 4
 - 3
 - 2

69. What is the output on your screen after this:
`echo hi >a ; ls a > wc`
 a. 1 1 2 b. no output c. 1 1 3
 d. 3 e. 2
70. Create a symbolic link under `/usr` named `bar` that has target `xy`:
 a. `ln -s '/usr/xy' /usr/bar`
 b. `ln -s /usr/bar 'xy'`
 c. `ln -s 'xy' /bar/usr`
 d. `ln -s 'xy' '/usr/bar'`
 e. `ln -s /usr/bar '/usr/xy'`
71. 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. 2 b. 5 c. 4 d. 3 e. 0
72. How many arguments are passed to the command by the shell:
`$ <wc wc " 1 '2 3' 4 " 5 6 ' 7 " 8 " ' >wc 9`
 a. 6 b. 2 c. 5 d. 3 e. 4
73. How many words are in file `out` after this:
`echo one >two >three >out`
 a. 4 b. 0 c. 2 d. 3 e. 1
74. 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. 2 c. 0 d. 1 e. 4
75. 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`
 a. 0 b. 2 c. 4 d. 1 e. 3
76. If I have a directory named `/1/2`, which action would increase its *link count* by exactly one?
 a. create one file named `/1/22`
 b. create a directory named `/1/2`
 c. create a directory named `/1/22`
 d. create one file named `/1/2/3`
 e. create a directory named `/1/2/3`

77. How many lines are in the file `bar` after this:
`echo hi >x ; echo ho >>x ; cat x x >bar`
 a. 4 b. 2 c. 6 d. 0 e. 1
78. 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."`
79. 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/sh: command not found`
 b. `cat: /etc/passwd: command not found`
 c. `bash: cat: command not found`
 d. `bash: /bin/cat: no such file or directory`
 e. `cat: bash: no such file or directory`
80. 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. 2 b. 5 c. 1 d. 4 e. 3
81. 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. `"$cow"` e. `1 2 3`
82. Which command recursively finds all things named `foo`?
 a. `find -name foo` b. `cat -name foo`
 c. `echo -name foo` d. `fgrep -name foo`
 e. `ls -name foo`
83. In an empty directory, what is in file `out` after this:
`ls nosuchfile | wc -w >out`
 a. nothing (empty file) b. `nosuchfile`
 c. `out` d. 1
 e. 0
84. In an empty directory, how many words are in file `c` after this:
`touch a b 1 b a ; ls >c`
 a. 1 b. 0 c. 2 d. 4 e. 3

85. 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
```
- 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
  - `foo` and `bar` each have three names (six names total)
  - `foo` and `bar` are names for the same file
86. What is the output on your screen of this unquoted command line:
- ```
mkdir a ; touch b a/b1 a/b2 ; find a -name b*
```
- no output
 - `b`
 - `b a/b1 a/b2`
 - `b1 b2`
 - `a/b1 a/b2`
87. How many files are touched? `touch 1 "2 3" '4' 5`
- 3
 - 5
 - 6
 - 7
 - 4
88. What is the link count of directory `dir` after these successful commands?
- ```
mkdir dir ; cd dir ; touch one ; mkdir two
```
- 5
  - 3
  - 2
  - 1
  - 4
89. Which of these statements is true?
- If `/y` is an empty directory, `echo /y/*` produces an error message.
  - If `/x` is an empty directory, `sort /x/*` produces an error message.
  - Only backslashes are strong enough to stop GLOB patterns from expanding.
  - Only single quotes are strong enough to stop GLOB patterns from expanding.
  - Only double quotes are strong enough to stop GLOB patterns from expanding.
90. 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
```
- `two` followed by `one`
 - `two`
 - `one`
 - `bash: /bin/xxx: command not found`
 - `one` followed by `two`
91. 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/*`
- no output
 - `dir/.on dir/.tw dir/.th`
 - `dir/.f.`
 - `dir/*`
 - `dir/ dir/.. dir/.on dir/.tw dir/.th dir/.f.`

92. Which of the following statements is true about this:
- ```
$ <dir/c cat dir/d
```
- The command is always invalid.
  - The command `cat` sees two arguments.
  - The command `cat` sees only one argument.
  - The command `dir/c` sees two arguments.
  - The command `dir/c` sees only one argument
93. What is the output of this in an empty directory:
- ```
touch a .a bc .bc def ; echo [ab]*
```
- `a .a bc .bc`
 - `a bc`
 - `[ab]*`
 - an error message from `echo` saying `[ab]*` does not exist
 - no output
94. What is the possible output on your screen of this:
- ```
echo wc >date ; sort date >date ; cat date
```
- `Fri Mar 2 12:00:00 EST 2018`
  - `1 6 29 date`
  - `1 6 28 date`
  - `wc`
  - no output
95. 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
```
- `2`
 - `1`
 - `1` followed by `2`
 - `2` followed by `1`
 - no output
96. In an empty directory, how many words are in file `out` after this:
- ```
touch a ; ls >out
```
- `2`
  - `0`
  - `3`
  - `1`
  - `4`
97. In an empty directory, what is the output on your screen after this:
- ```
echo hi >foo ; cp foo bar | wc -w
```
- no output
 - `0`
 - `1`
 - `2`
 - `3`
98. 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`

99. What is the output of this in an empty directory:
`touch 1 .1 23 .23 456 ; echo [12]*`
 a. `1 .1 23 .23 456`
 b. an error message from `echo` saying `[ab]*` does not exist
 c. `1 23`
 d. `[12]*`
 e. `1 .1 23 .23`
100. In an empty directory, what is the output on your screen after this:
`ls nosuchfile 2>out`
 a. `nosuchfile 2 not found` b. `nosuchfile`
 c. no output d. `2 not found`
 e. `nosuchfile not found`
101. 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. 1 b. 2 c. 0 d. 4 e. 3
102. 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
103. 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. `bash: bat: command not found`
 b. `bar`
 c. `foo`
 d. `bar` followed by `foo`
 e. `foo` followed by `bar`
104. Rewrite as a simplified absolute path:
`/home/me/../../you/../../../../etc/../../../../home/me/../../you/../../me/../../../../foo`
 a. `/home/you/foo` b. `/home/foo`
 c. `/etc/foo` d. `/foo`
 e. `/home/me/foo`
105. 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. 5 c. 3 d. 1 e. 4
106. 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. 0 c. 6 d. 5 e. 7

107. In an empty directory, what is the output on your screen after this:
`touch 1 2 3 ; cow="*" ; echo $cow`
 a. `$cow` b. `"1 2 3"` c. `1 2 3`
 d. `"*"` e. `*`
108. Which displays only the names in the current directory that are exactly three digits long (and no other names)?
 a. `echo [0-9][0-9][0-9]` b. `echo ???`
 c. `echo [1-3][1-3][1-3]` d. `echo [3][3][3]`
 e. `echo [?][?][?]`
109. How many lines are in the file `out` after this:
`date >f ; ls f >>f ; cat f f >out`
 a. 4 b. 2 c. 1 d. 6 e. 0
110. Which command shows *only* names that match `rtfm`, case-insensitive?
 a. `echo [rR][tT][fF][mM]` b. `echo *[rRtTfFmM]*`
 c. `echo [rRtTfFmM]` d. `echo [rR,tT,fF,mM]`
 e. `echo *rtfmRTFM*`
111. Which makes a file executable?
 a. `umask -x file` b. `chmod -x file`
 c. `umask u+x file` d. `umask u=x file`
 e. `chmod u+x file`
112. 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. 4 b. 0 c. 5 d. 2 e. 3
113. How many arguments are passed to the command by the shell:
`$ <foo foo -x " " -z -r" " >foo 'foo foo'`
 a. 7 b. 5 c. 9 d. 6 e. 8
114. 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`
115. Which displays only the non-hidden names in the current directory that contain the case-insensitive word `me` (and no other names)?
 a. `echo *(M,m,E,e)*` b. `echo ?[MmEe]?`
 c. `echo *[me]*` d. `echo *[MmEe]*`
 e. `echo *[Mm][Ee]*`
116. 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. 2 b. 5 c. 1 d. 3 e. 4

117. 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. 10 b. 6 c. 5 d. 4 e. 7
118. 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. 2 b. 4 c. 3 d. 1 e. 5
119. 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`
 a. 9 b. 1 c. 8 d. 2 2 e. 8 8
120. 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. **bash: foo: command not found**
 c. **one**
 d. **one** followed by **two**
 e. **two**
121. 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. 0 c. 6 d. 2 e. 5
122. Rewrite as a simplified absolute path:
`../../../../var/./a/../../../../var/b/../../../../etc/./bar/./foo`
 a. `/var/foo` b. `/etc/bar/foo` c. `/etc/foo`
 d. `/var/b/foo` e. `/var/a/foo`
123. 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`
124. What is true about this: `date >ls ; ls -ls ls >wc`
 a. The **wc** command counts the output of the **ls** command.
 b. The **ls** command receives the output of **date** on standard input.
 c. The file **wc** has one line in it.
 d. The shell finds and executes three different commands.
 e. The **ls** command is executed more than once.
125. 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`

126. Which always prints just the two characters `$x` on the screen?
 a. `echo "$x"` b. `echo $x` c. `echo "$$x"`
 d. `echo $$x` e. `echo '$x'`
127. What is the output of this in an empty directory:
`touch .1 .2 .3 4 5 6 ; echo .*`
 a. `.1 .2 .3 4 5 6`
 b. an error message from **echo** saying `.*` does not exist
 c. `.*`
 d. `4 5 6`
 e. `. . . .1 .2 .3`
128. What is the output on your screen of this:
`echo wc >wc ; wc wc >wc ; cat wc`
 a. no output b. `1 1 2 wc` c. `0 0 0 wc`
 d. `1 1 3 wc` e. `wc`
129. 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 the same file
 b. this output is not possible
 c. **foo** and **bar** each have three names (six names total)
 d. **foo** and **bar** are names for different files
 e. **foo** and **bar** are two of three names for the same file
130. 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. 1 b. 2 c. 4 d. 5 e. 3
131. How many lines are in the file **out** after this:
`echo hi >x ; echo ho >>x ; cat x x x >out`
 a. 0 b. 3 c. 1 d. 2 e. 6
132. What is in the local variable `$$` ?
 a. the command name of the previous command line
 b. the first argument of the previous command line
 c. the cpu cost of the current session, in dollars
 d. the process ID of the current shell
 e. `$$` is not a valid variable name
133. 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. `A a`
 d. `* >*` e. `A a >*`
134. What is the output on your screen after this:
`echo hi >a ; cp a b | wc -w`
 a. 2 b. 0 c. 3
 d. no output e. 1

135. 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. 0 b. 5 c. 7 d. 12 e. 8
136. What displays on your screen given this command:
`ls >ls ; wc ls >wc ; sort ls | cat wc`
 a. nothing displays because **cat** ignores the pipe
 b. **cat** reads the pipe and the **wc** and displays both together
 c. only the **wc** displays because **cat** ignores the pipe
 d. **sort** displays the **ls** and **cat** displays the **wc**
 e. only the **ls** displays because **cat** ignores the pipe
137. In an empty directory, what is the output on your screen after this:
`echo one >.bar ; echo .????*`
 a. .????*
 b. one
 c. .bar
 d. .. .bar
 e. an error message from **echo** saying .????* does not exist
138. What is usually in the environment variable **\$PATH**?
 a. the absolute path of your login home directory
 b. the absolute path of the system **/path** directory
 c. a colon-separated list of directories containing command names
 d. the absolute path of your login shell
 e. a colon-separated list of your **passwd** file fields
139. 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. 1 b. 0 c. 3 d. 4 e. 2
140. What is the output of this in an empty directory: `echo *`
 a. .
 b. ..
 c. *
 d. an error message from **echo** saying * does not exist
 e. no output
141. Which makes pathnames **/usr/local/bin** and **/usr/bin** lead to the same directory?
 a. `ln . /usr/local` b. `rmdir /usr/local`
 c. `mkdir /usr/local` d. `ln -s . /usr/local`
 e. `touch /usr/local`

142. 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. 5 b. 2 c. 1
 d. no output e. 3
143. What is the link count of an empty directory?
 a. 4 b. 0 c. 2 d. 1 e. 3
144. 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-- 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-- 1 me me 100 Jan 1 1:00 d
```


 a. 3 b. 4 c. 0 d. 2 e. 1
145. 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/\??`
146. What is in file **foo** after this: `echo 1 2 >foo 3`
 a. 1 2 b. 1 2 3
 c. 3 d. echo 1 2
 e. nothing (empty file)
147. 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. 3 b. 4 c. 5 d. 6 e. 2
148. What is the output on your screen of this:
`echo bat >pig ; echo one | tail pig`
 a. one b. one followed by bat
 c. bat d. an error message
 e. bat followed by one
149. In an empty directory, how many words are in file **out** after this:
`echo hi >a ; ls >out`
 a. 0 b. 1 c. 3 d. 4 e. 2
150. Which command counts lines containing the string **refused** in only the month **October 2016** in the **denyhosts** log file?
 a. `fgrep -c 2016-10 denyhosts | fgrep refused`
 b. `fgrep '2016-10 refused' denyhosts`
 c. `fgrep refused denyhosts ; fgrep -c 2016-10`
 d. `fgrep 'refused' denyhosts | fgrep -c '2016-10'`
 e. `fgrep refused denyhosts | fgrep -c October 2016`

164. Which of the following is true, given this long directory listing:
`drwxr-x--x 128 me me 32 Jan 1 1:00 dir`
- The number 32 is the count of links (names) this directory has.
 - The number 128 is the count of links (names) this directory has.
 - The number 128 is the inode number of this directory.
 - The number 128 is the size of this directory.
 - The number 32 is the inode number of this directory.
165. 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*`
166. 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
```
- 4
  - 0
  - 3
  - 2
  - 1
167. 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`?
- 456
  - 787
  - 123
  - 454
  - 789
168. 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`
- `foo` followed by `hi`
  - `foo`
  - `hi`
  - `hi` followed by `foo`
  - `bash: pig: command not found`
169. What is the link count of directory `dir` after these successful commands?  
`mkdir dir ; mkdir dir/foo ; touch dir/bar`
- 2
  - 4
  - 3
  - 1
  - 5
170. 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`
- no output on screen
  - ten
  - two
  - one followed by two and ten
  - one

171. 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`
- 2
  - 0
  - 1
  - 3
  - 4
172. How many arguments are passed to the command by the shell:  
`$ <foo foo " a 'b c' d " e f ' g " h " ' >foo`
- 5
  - 2
  - 3
  - 4
  - 6
173. How many arguments are passed to the command by the shell:  
`$ <foo foo " a 'b c' d " e ' f " g " ' >foo`
- 6
  - 3
  - 2
  - 5
  - 4
174. 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/*`
- `cow/*`
  - `cow/.AA cow/.A1 cow/.BB cow/.B.`
  - no output
  - `cow/.AA cow/.A1 cow/.BB`
  - `cow/.B.`
175. How many arguments are passed to the command by the shell:  
`$ echo 'It's a bird! It's a plane!'`
- 4
  - 5
  - 2
  - 3
  - 1
176. 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`
- 0
  - 5
  - 6
  - 7
  - 4
177. In an empty directory, what is the output on your screen after this:  
`echo hi >a ; ls >wc -l`
- 0
  - 1
  - 2
  - no output
  - a
178. What command shows all the lines in file `cow` that contain the string `pig`?
- `fgrep cow pig`
  - `fgrep pig <cow`
  - `fgrep cat cow pig`
  - `cat cow > fgrep pig`
  - `fgrep pig >cow`
179. In an empty directory, what is the output on your screen after this:  
`ls out 2>/dev/null`
- no output
  - `out`
  - `ls: out 2>/dev/null: No such file or directory`
  - `ls: /dev/null: No such file or directory`
  - `ls: out: No such file or directory`
180. How many arguments are passed to the command by the shell:  
`$ echo " 1 2 "three ' 4 ' five"6"`
- 3
  - 1
  - 5
  - 4
  - 9

181. If directory **dir** contains three four-character file names: **.123**, **.124**, **.???**, then what is the output on your screen of this: **echo dir/????**
- dir/.123 dir/.124 dir/.???**
  - dir/????**
  - echo: dir/????: No such file or directory**
  - no output
  - dir/.123 dir/.124**
182. In a directory containing one file named **dog**, what is the output on your screen after this: **2>/dev/null ls nosuchfile**
- no output
  - bash: 2>/dev/null: command not found**
  - nosuchfile**
  - ls: nosuchfile: No such file or directory**
  - dog**
183. 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
```
- 3
 - 4
 - 5
 - 2
 - 1
184. How many lines are in file **out** after this: **echo hi >dog >out >cat**
- 4
 - 1
 - 0
 - 2
 - 3
185. How many arguments are passed to the command by the shell:
- ```
$ <bar bar -b"-a '-r' >bar" bar >out
```
- 5
  - 3
  - 2
  - 6
  - 4
186. What is in file **c** after this:
- ```
echo A >a ; ln a b ; echo B >b ; ln a c ; rm a b
```
- nothing (empty file)
 - A**
 - B**
 - A** followed by **B**
 - no such file (nonexistent)
187. 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/a\?**
 - rm /a?**
 - rm /a/a[*]**
188. Rewrite as a simplified absolute path:
- ```
/usr/./bin/./lib/./../etc/./usr/./lib/./bin/./bar
```
- /bar**
  - /usr/lib/bar**
  - /usr/bar**
  - /etc/bar**
  - /usr/bin/bar**
189. In an empty directory, how many words are in file **c** after this:
- ```
touch a ; mv b a >b ; ls >c
```
- 1
 - 3
 - 4
 - 2
 - 0

190. In an empty directory, what is the output on your screen after this:
- ```
ls 2>/dev/null nosuchfile
```
- nosuchfile**
  - ls: 2>/dev/null nosuchfile: No such file or directory**
  - no output
  - ls: nosuchfile: No such file or directory**
  - ls: /dev/null: No such file or directory**
191. In an empty directory, how many words are in file **out** after this:
- ```
touch 1 2 3 2 1 ; ls >out
```
- 5
 - 4
 - 0
 - 6
 - 3
192. In an empty directory, what is the output on your screen after this:
- ```
touch 1 2 .a .b ; echo .*
```
- .. .a .b**
  - 1 2**
  - .\***
  - an error message from **echo** saying **.\*** does not exist
  - .a .b**
193. 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
```
- 2
 - 0
 - 4
 - 3
 - 1
194. 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
```
- 9
  - 8
  - 8 8
  - 1
  - 2 2
195. What is the output on your screen of this:
- ```
echo pig >one ; echo cow | head -n 2 one
```
- pig** followed by **cow**
 - cow** followed by **pig**
 - pig**
 - cow**
 - an error message
196. Which of the following commands will leave **file1** non-empty?
- cat file1 > file1**
 - sort file1 > file1**
 - tail file1 > file1**
 - head file1 > file1**
 - wc file1 > file1**
197. In an empty directory, how many lines are in file **out** after this:
- ```
ls . .. nosuchfile 2>out
```
- 3
  - 1
  - 2
  - 4
  - 0

198. How many arguments are passed to the command by the shell:  
`$ echo " 1 2 " three ' 4 ' five"6"`  
 a. 4            b. 1            c. 9            d. 3            e. 5
199. Which shows the current date?  
 a. `bash >date ; cat date`            b. `date | bash`  
 c. `echo date | bash`            d. `bash date`  
 e. `bash <date`
200. 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\*`
201. 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. `two`  
 b. `one` followed by `two`  
 c. `one`  
 d. `bash: /bin/foo: command not found`  
 e. `two` followed by `one`
202. Which outputs only lines 11-15 of the Unix password file?  
 a. `head -n 15 /etc/passwd | tail -n 5`  
 b. `tail -n 15 /etc/passwd | head -n 10`  
 c. `head -n 15 /etc/passwd | tail -n 5 /etc/passwd`  
 d. `tail -n 10 /etc/passwd | head -n 15 /etc/passwd`  
 e. `head -n 10 /etc/passwd | tail -n 15 /etc/passwd`
203. Which would show the index (inode) number of a file?  
 a. `find -i file`            b. `ls -l file`            c. `ls -i file`  
 d. `cat -l file`            e. `cat -i file`
204. How many arguments are passed to the command by the shell:  
`$ echo ' one two ' three ' four ' 5'6'`  
 a. 5            b. 6            c. 9            d. 1            e. 4
205. 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. 1            b. 2            c. 3            d. 4            e. 0
206. Which command moves a file into the parent directory?  
 a. `mv ../file`            b. `mv .. file`            c. `mv file, ..`  
 d. `mv file/..`            e. `mv file ..`

207. How many lines are in the file `bar` after this:  
`echo hi >x ; echo ho >>x ; cat x >bar`  
 a. 1            b. 2            c. 4            d. 0            e. 6
208. 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. `nosuchfile`  
 c. `a`  
 d. `ls: 2>/dev/null: No such file or directory`  
 e. `ls: nosuchfile: No such file or directory`
209. In a directory containing one file named `dog`, what is the output on your screen after this: `1>/dev/null ls *`  
 a. `*`  
 b. `ls: *: No such file or directory`  
 c. no output  
 d. `dog`  
 e. `bash: 1>/dev/null: command not found`
210. 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. 7            c. 8            d. 5            e. 1
211. 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. `/bar/../dir/foo`  
 c. `/bin/bar/dir/foo`            d. `/bin/dir/foo/bar`  
 e. `/dir/foo`
212. In an empty directory, what is the output on your screen after this:  
`touch 1 2 3 ; cow="*" ; echo "$cow"`  
 a. `"$cow"`            b. `$cow`            c. `"1 2 3"`  
 d. `*`            e. `1 2 3`
213. 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. 3  
 d. 2 followed by 3            e. 3 followed by 2
214. In an empty directory, what is the output on your screen after this:  
`echo one >.bar ; echo .?*`  
 a. `.. .bar`  
 b. `one`  
 c. `.?*`  
 d. `.bar`  
 e. an error message from `echo` saying `.?* does not exist`

215. 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. 2 c. 4 d. 0 e. 3
216. 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. 1 followed by 2            b. no output            c. 2 followed by 1  
d. 2            e. 1
217. In an empty directory, how many arguments are passed to the `cat` command in this: `date >a1 ; touch a2 ba ca ; cat a*`
- a. 3            b. 1            c. 2            d. 4            e. none
218. What is the output on your screen after this:
- ```
mkdir dir ; rmdir dir | wc -w
```
- a. 2 b. 0 c. 3
d. 1 e. no output
219. 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. 9            b. 6            c. 5            d. 8            e. 7
220. 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 absolute path of your login shell  
d. the relative path of the system `/shell` directory  
e. the relative path of your login shell
221. Which command finds your account login userid in the password file?
- a. `find /etc/passwd -name $USER`  
b. `cat $USER /etc/passwd`  
c. `fgrep /etc/passwd $USER`  
d. `fgrep $USER /etc/passwd`  
e. `find $USER /etc/passwd`
222. What is the output on your screen after this:
- ```
echo one >x ; ln x y ; echo two >>y ; sort x
```
- a. one b. no output
c. one followed by two d. two
e. two followed by one

223. What is the output of this in an empty directory:
- ```
touch .a .b .c ; echo .*
```
- a. an error message from `echo` saying `.*` does not exist  
b. `.*`  
c. no output  
d. `. . . .a .b .c`  
e. `.a .b .c`
224. The option to `ls` that shows inode (index) numbers is:
- a. `-l`            b. `-a`            c. `-l`            d. `-i`            e. `-x`
225. What is in file `c` after this:
- ```
echo foo >a ; ln a b ; echo bar >>b ; ln a c ; rm a
```
- a. `bar` b. `foo` followed by `bar`
c. no such file (nonexistent) d. nothing (empty file)
e. `foo`
226. How many words are in the file `x` after this:
- ```
echo 1 2 >x ; echo 3 >x ; echo 4 >>x
```
- a. 3            b. 4            c. 1            d. 0            e. 2
227. 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. 4 c. 1 d. 2 e. 3
228. 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. 3 c. 5
d. 0 e. 2
229. What does *quoting* mean on a shell command line?
- a. using more than one pathname argument to a command, e.g. `rm a b c`
b. setting the `PS1` variable to be your shell prompt
c. typing a "control" character using the `[CTRL]` key
d. using a leading tilde ("`~`") on a pathname to mean your `HOME` directory
e. turning off the special meaning of shell meta-characters
230. 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. 4            b. 5            c. 1            d. 2            e. 3

231. Which allows programs in the current directory to execute without preceding the names with ./? (P.S. Security Risk! Don't do this!)
- `PATH=/bin:/usr/bin:.`
  - `$PATH=/usr/bin:./bin`
  - `$PATH=.:$HOME:/usr/bin`
  - `PATH=/usr/bin/.:$HOME`
  - `PATH=./$HOME:/usr/bin`
232. You enter this `cp a/b c/` and get `cp: a: No such file or directory` because:
- directory `a` does not exist
  - the command `cp` is not in your search `PATH`
  - pathname `a` exists but is a file, not a directory
  - you forgot to specify the destination file name after `c/`
  - directory `c` does not exist
233. 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
```
- `one` followed by `two`
 - no output
 - `two` followed by `one`
 - `two`
 - `one`
234. In an empty directory, what is the output on your screen after this:
- ```
echo one >.bar ; ls .????*
```
- an error message from `ls` saying `.????*` does not exist
  - `one`
  - `.bar`
  - `.????*`
  - `.. .bar`
235. If I have a directory named `a/b`, which action would increase its *link count* by exactly one?
- create a hard link to directory `b` named `b2`
  - create a directory named `a/b/c`
  - create a file named `a/b2`
  - create a directory named `a/b2`
  - create a file named `a/b/c`
236. Which file is a DOS/Windows file?
- ASCII text
  - ASCII text, with CR line terminators
  - ASCII text, with CRLF line terminators
  - ASCII text, with no line terminators
  - ASCII text, with LF line terminators
237. 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`
- 0
  - 5
  - 10
  - 6
  - 7

238. In an empty directory, what is the output on your screen after this:
- ```
echo hi >a ; sort * 1>/dev/null
```
- `sort: 1>/dev/null: No such file or directory`
 - `a`
 - `sort: *: No such file or directory`
 - no output
 - `hi`
239. How many arguments are passed to the command by the shell:
- ```
$ echo 'It's a bird! No! It's a plane!'
```
- 1
  - 4
  - 3
  - 2
  - 5
240. The correct syntax to assign to a shell variable is:
- `V = "foo bar"`
  - `"V=foo bar"`
  - `V=foo bar`
  - `V="foo bar"`
  - `V = foo bar`
241. How many files are touched? `touch "1" 2 3 " " 4 5`
- 5
  - 7
  - 3
  - 4
  - 6
242. What is the output on your screen of this:
- ```
echo hi >hi ; head hi >hi ; wc hi
```
- 1 1 3 hi
 - 1 1 2 hi
 - 0 0 0 hi
 - no output
 - 2 2 4 hi
243. What is the output of this in an empty directory:
- ```
touch 1 2 3 .a .ab .abc ; echo [.]*
```
- `[.]*`
  - no output
  - `. . .a .ab .abc`
  - an error message from `echo` saying `[.]*` does not exist
  - `.a .ab .abc`
244. What is in file `foo` after this:
- ```
echo hi >a ; ln a b ; echo me >b ; ln a foo ; rm a b
```
- `hi`
 - no such file (nonexistent)
 - nothing (empty file)
 - `me`
 - `hi` followed by `me`
245. Which shows only lines 6-10 of file `foo`?
- `head -10 foo | tail -5`
 - `tail -15 foo | head -5`
 - `tail -10 foo | head -6`
 - `head -6 foo | tail -10`
 - `head -10 foo | tail -6`
246. Which outputs inode/filename pairs for names in the current directory, sorted by inode number?
- `ls /* | sort -node`
 - `sort -n | ls -ai`
 - `ls -i * > sort -n`
 - `ls -ai | sort -n`
 - `ls -node * > sort -n`

247. In an empty directory, how many arguments are passed to the **wc** command in this:
`date >ol ; touch a1 b2 out >ol ; wc o*`
 a. 5 b. 2 c. 4 d. 1 e. 3
248. Which one of these names is usually a shell environment variable?
 a. **FooBar** b. **fooBar** c. **FooBar**
 d. **FOOBAR** e. **foobar**
249. 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 the same file
 c. **foo** and **bar** each have three names (six names total)
 d. **foo** and **bar** are names for different files
 e. **foo** and **bar** are two of three names for the same file
250. Which command copies a directory:
 a. `mv -r dir1 dir2` b. `mv -f dir1 dir2`
 c. `cp dir1 dir2` d. `cp -r dir1 dir2`
 e. `mv -rf dir1 dir2`
251. 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. **cat: bat: No such file or directory**
 b. **bat**
 c. **foo**
 d. no output on screen
 e. **bash: cat: command not found**
252. What is the link count of directory **foo** after these successful commands?
`mkdir foo ; cd foo ; touch a b c`
 a. 5 b. 2 c. 3 d. 1 e. 4
253. How many arguments are passed to the command by the shell:
`$ <cow cow "-x" "-y" "-z" >cow cow`
 a. 4 b. 6 c. 5 d. 7 e. 3
254. What is the output on your screen after this: `echo hi >out | wc -w`
 a. 1 b. 3 c. no output
 d. 2 e. 0
255. In an empty directory, what is the output on your screen after this:
`touch a ; ls >wc -l`
 a. 1 b. 2 c. 3
 d. 0 e. no output

256. 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. an error message because `*/.` does not exist
 c. 15 directory names
 d. 25 pathnames
 e. `*/.`
257. How many arguments are passed to the command by the shell:
`$ <cow cow "-x" "-y" "-z" >cow cow`
 a. 2 b. 3 c. 5 d. 4 e. 6
258. 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. no output b. 1 c. 4
 d. 8 e. 6
259. What is the link count of directory **dir** after these successful commands?
`mkdir dir ; cd dir ; touch a b c ; mkdir d e`
 a. 2 b. 3 c. 7 d. 5 e. 4
260. How many arguments are passed to the command by the shell:
`$ echo 'It's "1 2" isn't it? I can't decide.'`
 a. 4 b. 2 c. 5 d. 6 e. 3
261. 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`
 a. 4 b. 2 c. 3 d. 5 e. 1
262. 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. **foo**
 b. **foo** followed by **hi**
 c. **hi**
 d. **hi** followed by **foo**
 e. **bash: prg: command not found**
263. What is the link count of directory **dir** after these successful commands?
`mkdir dir ; touch foo ; cd dir ; ln ../foo bar`
 a. 3 b. 5 c. 4 d. 2 e. 1
264. 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`
 a. 5 b. 2 c. 6 d. 4 e. 3
265. How many files are touched? `touch '1' '2 3' '4' ' ' 5`
 a. 2 b. 1 c. 4 d. 5 e. 3

266. In an empty directory, what is the output on your screen after this:
`touch 1 2 3 ; cow="*" ; echo '$cow'`
 a. '\$cow' b. * c. 1 2 3
 d. '1 2 3' e. \$cow
267. If file **foo** occupies one disk block, how many disk blocks are in use after this:
`cp foo bar ; ln bar one ; cp one two ; cp one xxx`
 a. 3 b. 2 c. 4 d. 5 e. 1
268. What is the link count of file **f** after these successful commands?
`rm f ; touch f ; cp f x
 ln -s f a ; ln x y ; ln a z ; ln x b`
 a. 5 b. 4 c. 2 d. 1 e. 3
269. Given an existing file of yours named **cat**, what is the output on your screen of this: `echo xx >cat ; head cat >cat ; wc cat`
 a. 1 1 3 cat b. no output c. 2 2 4 cat
 d. 0 0 0 cat e. 1 1 2 cat
270. How many arguments are passed to the command by the shell:
`$ echo "cow "y " bat 'man x' " pig'a "hop' a b`
 a. 11 b. 4 c. 7 d. 6 e. 5
271. 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. **bar**
 b. **bash: /bin/cat: command not found**
 c. no output on screen
 d. **pig**
 e. **/bin/cat: pig: No such file or directory**
272. 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. 8 c. 0 d. 12 e. 5
273. 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** followed by **one**
 b. **bash: foo: command not found**
 c. **one**
 d. **one** followed by **two**
 e. **two**
274. 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. 2 c. 3 d. 4 e. 5

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=/bin/xxx:/usr/bin/xxx:/etc/passwd ; xxx`
 a. **bash: xxx: command not found**
 b. **two** followed by **one**
 c. **one**
 d. **two**
 e. **one** followed by **two**
276. 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. 9 b. 7 c. 5 d. 8 e. 6
277. 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. 0 b. 4 c. 3 d. 5 e. 2
278. 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. 2 b. 3 c. 5 d. 9 e. 6
279. 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. 8 b. 12 c. 0 d. 7 e. 10
280. Which command below is the best way to find a line containing a question mark (?) in the file **/etc/passwd**?
 a. `find '?' /etc/passwd`
 b. `search '?' /etc/passwd`
 c. `fgrep /etc/passwd '?'`
 d. `fgrep './?' /etc/passwd`
 e. `fgrep '?' /etc/passwd`
281. Which command usually goes in your **.bash_profile** file?
 a. `source ~/.bash_profile` b. `source ~/.bashrc`
 c. `.bashrc source` d. `cat .bashrc`
 e. `.bash_profile source`
282. What is the output of this in an empty directory:
`date >.date ; users >.users ; echo .?*`
 a. `.date .users`
 b. `.. .date .users`
 c. `.?*`
 d. an error message from **echo** saying `.?*` does not exist
 e. `.date`

283. In an empty directory, what is the output on your screen after this:
`echo one >.bar ; echo .??*`
 a. `.??*`
 b. `one`
 c. an error message from `echo` saying `.??*` does not exist
 d. `.. .bar`
 e. `.bar`
284. In an empty directory, what is the output on your screen after this:
`echo hi >a ; mv a b ; ln b c ; ls >wc -l`
 a. 2
 b. no output
 c. 1
 d. a
 e. 0
285. How many arguments are passed to the command by the shell:
`$ <pig pig -x " " -z -r" " >pig pig pig`
 a. 6
 b. 5
 c. 9
 d. 8
 e. 7
286. What command will recursively find all pathnames named `foo` in `/bin`?
 a. `find /bin -name 'foo'`
 b. `fgrep 'foo' /bin`
 c. `ls -R 'foo' /bin`
 d. `fgrep /bin -basename 'foo'`
 e. `find foo -name '/bin'`
287. What is the link count of directory `d` after these successful commands?
`mkdir d ; touch f ; cd d ; ln ../f x`
 a. 1
 b. 2
 c. 5
 d. 3
 e. 4
288. 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. 1
 b. 7
 c. 2
 d. 3
 e. 5
289. 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. `/foo`
 c. `/etc/foo/bar`
 d. `/etc/bar/foo`
 e. `/bar/foo`
290. Which option to `ls` displays the directory itself and not its contents?
 a. `-a`
 b. `-l`
 c. `-R`
 d. `-i`
 e. `-d`
291. 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. 1
 b. 0
 c. 2
 d. 3
 e. 4

292. Which command shows the name of the current computer:
 a. `hostname`
 b. `history`
 c. `comname`
 d. `find`
 e. `whois`
293. What is the link count of directory `z` after these successful commands?
`mkdir z ; cd z ; touch a b ; mkdir c d e`
 a. 4
 b. 3
 c. 6
 d. 5
 e. 7
294. What is the output of this in an empty directory:
`touch .a .b .c ; echo .??*`
 a. no output
 b. an error message from `echo` saying `.??*` does not exist
 c. `.??*`
 d. `.. .a .b .c`
 e. `.a .b .c`
295. How many lines are in file `out` after this: `date >wc >cat >out`
 a. 0
 b. 2
 c. 1 6 29
 d. 1
 e. 0 0 0
296. 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. `bash: pig: command not found`
 d. `foo` followed by `xx`
 e. `xx`
297. What is the link count of directory `d` after these successful commands?
`mkdir d ; cd d ; touch a ; mkdir b c`
 a. 2
 b. 5
 c. 6
 d. 3
 e. 4
298. How many files are touched? `touch "1 " 2 3" " ' ' 4 5`
 a. 7
 b. 5
 c. 6
 d. 4
 e. 3
299. 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. 2
 b. 5
 c. 3
 d. 4
 e. 6
300. 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. 5
 b. 6
 c. 0
 d. 10
 e. 8

334. How many arguments are passed to the command by the shell:
`$ <bar bar -b "-a" '-r' >bar bar bar`
 a. 6 b. 7 c. 5 d. 3 e. 4
335. How many arguments are passed to the command by the shell:
`$ <f z " a 'b c' d " 1 2 ' g " h " ' >z`
 a. 5 b. 4 c. 3 d. 2 e. 6
336. 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/foo` b. `/bin/bar` c. `/bin/foo/bar`
 d. `/foo/bar` e. `/bar`
337. 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. `foo` and `bar` are two of three names for this file
 b. `foo` and `bar` are names for the same file
 c. this output is not possible
 d. `foo` and `bar` each have two names (four names total)
 e. `foo` and `bar` each have three names (six names total)
338. In an empty directory, what is in file `foo` after this:
`echo hi >foo ; ls nosuchfile | cat >foo`
 a. `foo`
 b. nothing (empty file)
 c. `hi`
 d. `ls: cannot access nosuchfile`
 e. `nosuchfile`
339. 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. 5 b. 1 c. 4 d. 2 e. 3
340. 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`
 b. `mom` followed by `hi`
 c. `mom`
 d. `bash: foo: command not found`
 e. `hi` followed by `mom`
341. In an empty directory, how many words are in file `foo` after this:
`date >.bar >.out ; ls >foo`
 a. 1 b. 4 c. 2 d. 0 e. 3

342. If `mt` is an empty sub-directory, what is true after this:
`touch bar ; mkdir foo ; mv mt/../../bar mt/foo`
 a. the command fails because `mt/foo` is not a directory
 b. the directory `foo` now contains a file named `bar`
 c. the directory `mt` is still empty
 d. the directory `mt` now contains a file named `foo`
 e. the directory `mt` now contains a file named `bar`
343. 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. `pig`
 d. `bat`
 e. `bat` followed by `pig`
344. Which shows the file in `/bin` with the largest checksum?
 a. `cat /bin | sum | sort -nr | head -n 1`
 b. `ls /bin/* | sum | sort -nr | head -n 1`
 c. `sum /bin | sort -nr | head -n 1`
 d. `cat /bin/* | sum | sort -nr | head -n 1`
 e. `sum /bin/* | sort -nr | head -n 1`
345. In an empty directory, how many lines are in file `foo` after this:
`ls nosuchfile . .. 2>foo`
 a. 0 b. 3 c. 2 d. 4 e. 1
346. What is the usual output on your screen of this:
`mkdir dir ; cd dir >foo ; cat foo`
 a. no output
 b. `dir`
 c. `foo`
 d. `bash: cd: dir: No such file or directory`
 e. `cat: foo: No such file or directory`
347. Did you read all the words of the test instructions on page one?
 a. **Sim** (Yes - Portuguese) b. **Taip** (Yes - Lithuanian)
 c. **Tak** (Yes - Polish) d. **Igen** (Yes - Hungarian)
 e. **Jes** (Yes - Esperanto)