

PRINT Name: _____ LAB Section:

One-Answer Multiple Choice 237 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 **D**
193. Answer **193** is **B**
194. Answer **194** is **C**
195. Answer **195** is **A**
196. Answer **196** is **A**

Your Test Version is:

E D B C A A

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. **Sim** (Yes - Portuguese)
 - b. **Tak** (Yes - Polish)
 - c. **Jes** (Yes - Esperanto)
 - d. **Igen** (Yes - Hungarian)
 - e. **Taip** (Yes - Lithuanian)
2. My three-digit Lab Section number is:
 - a. The Test Version number printed in the top left corner.
 - b. My lab room number, e.g. **B384**, **T321**, **J218**, **N201**
 - c. The timetable section number of my weekly 2-hour lab period.
 - d. My lecture room number, e.g. **T130**, **T117**
 - e. My lecture section number, e.g. **010** or **020**.
3. 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
```

 - a. 1
 - b. 4
 - c. 2
 - d. 3
 - e. 5
4. What is the output on your screen after this:


```
mkdir dir ; rmdir dir | wc -w
```

 - a. 1
 - b. no output
 - c. 2
 - d. 0
 - e. 3
5. What is the link count of directory **dir** after these successful commands?


```
mkdir dir ; cd dir ; touch foo ; mkdir a b c
```

 - a. 4
 - b. 3
 - c. 2
 - d. 5
 - e. 1

6. What is the usual output on your screen of this:


```
mkdir dir ; cd dir >dir/foo ; cat foo
```

 - a. **cat: foo: No such file or directory**
 - b. **bash: dir/foo: No such file or directory**
 - c. **foo**
 - d. no output
 - e. **dir**
7. 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. 2
 - c. 3
 - d. 4
 - e. 5
8. 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. 6
 - c. 3
 - d. 2
 - e. 4
9. 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. 4
 - b. 1
 - c. 2
 - d. 3
 - e. 0
10. Which command line 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]'**
11. The option to **ls** that shows inode (index) numbers is:
 - a. **-l**
 - b. **-i**
 - c. **-x**
 - d. **-a**
 - e. **-1**
12. How many files are touched?


```
touch 1 "2 3" '4' 5
```

 - a. 5
 - b. 7
 - c. 3
 - d. 4
 - e. 6
13. 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. 3
 - b. 1
 - c. 5
 - d. 4
 - e. 2
14. In an empty directory, what is the output on your screen after this:


```
touch a ; ls | wc -w
```

 - a. 1
 - b. 3
 - c. no output
 - d. 0
 - e. 2

15. If your terminal type is `xterm`, what is the output of this: `echo '$TERM'`
- '\$TERM'
 - `xterm`
 - '`xterm`'
 - `$TERM`
 - no output on screen
16. 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/b2`
 - create a file named `a/b2`
 - create a directory named `a/b/c`
 - create a file named `a/b/c`
17. You enter this `cp a/b c/` and get `cp: a: No such file or directory` because:
- you forgot to specify the destination file name after `c/`
 - pathname `a` exists but is a file, not a directory
 - directory `a` does not exist
 - directory `c` does not exist
 - the command `cp` is not in your search `PATH`
18. 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
```
- `foo` followed by `hi`
  - `hi`
  - `foo`
  - `bash: prg: command not found`
  - `hi` followed by `foo`
19. 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`
- `cat: bat: No such file or directory`
  - `bash: cat: command not found`
  - `bat`
  - `foo`
  - no output on screen
20. 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
```
- `bash: bat: command not found`
 - `foo` followed by `hi`
 - `foo`
 - `hi` followed by `foo`
 - `hi`

21. 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
```
- `one`
  - `one` followed by `two`
  - `two`
  - `two` followed by `one`
  - `bash: xxx: command not found`
22. 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
```
- 3
 - 2
 - 5
 - 4
 - 1
23. What is usually in the environment variable `$HOME`?
- the relative path of the system `/home` directory
 - the absolute path of the system `/home` directory
 - the relative path of your login home directory
 - the absolute path of your login home directory
 - the relative path of the `ROOT` directory
24. Which command line makes a file executable?
- `chmod -x file`
 - `umask u=x file`
 - `chmod u+x file`
 - `umask u+x file`
 - `umask -x file`
25. 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
```
- 3
  - 4
  - 0
  - 1
  - 2
26. Which command line makes pathnames `/usr/local/bin` and `/usr/bin` lead to the same directory?
- `rmdir /usr/local`
  - `ln -s . /usr/local`
  - `mkdir /usr/local`
  - `ln . /usr/local`
  - `touch /usr/local`
27. How many arguments are passed to the command by the shell:
- ```
echo 'It's a bird! It's a plane!'
```
- 4
 - 1
 - 3
 - 2
 - 5
28. What is the link count of directory `d` after these successful commands?
- ```
mkdir d ; mkdir d/a ; mkdir d/a/b ; mkdir d/a/c
```
- 5
  - 3
  - 1
  - 4
  - 2

29. 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. 2 b. 0 c. 3 d. 1 e. 4
30. 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. `pig`
 c. `bash: /bin/cat: command not found`
 d. `/bin/cat: pig: No such file or directory`
 e. no output on screen
31. How many arguments are passed to the command by the shell:
- ```
<bar bar -b"-a '-r' >bar" bar >out
```
- a. 4                      b. 6                      c. 5                      d. 2                      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. 4 b. 5 c. 2 d. 1 e. 3
33. What is the link count of directory `d` after these successful commands?
- ```
mkdir d ; mkdir d/a ; touch d/b
```
- a. 3                      b. 4                      c. 2                      d. 5                      e. 1
34. What is the link count of directory `x` after these successful commands?
- ```
mkdir x ; mkdir x/y ; mkdir x/z ; mkdir x/y/z
```
- a. 3 b. 4 c. 5 d. 1 e. 2
35. Which command copies a directory:
- a. `mv -r dir1 dir2` b. `cp dir1 dir2`
 c. `mv -f dir1 dir2` d. `mv -rf dir1 dir2`
 e. `cp -r dir1 dir2`
36. Which command moves a file into the parent directory?
- a. `mv file ..` b. `mv file/..` c. `mv ../file`
 d. `mv file, ..` e. `mv .. file`
37. What is the output on your screen of this unquoted command line:
- ```
mkdir a ; touch b a/b1 a/b2 ; find a -name b*
```
- a. `b`                      b. `a/b1 a/b2`                      c. `b1 b2`  
 d. no output                      e. `b a/b1 a/b2`

38. In an empty directory, what is the output on your screen after this:
- ```
echo one >.bar ; ls .????*
```
- a. `.. .bar`
 b. `one`
 c. `.bar`
 d. `.????*`
 e. an error message from `ls` saying `.????*` does not exist
39. 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'`
40. How many arguments are passed to the command by the shell:
- ```
<pig pig -x " " -z -r" " >pig pig pig
```
- a. 9                      b. 6                      c. 5                      d. 7                      e. 8
41. Which command counts lines with two adjacent asterisk characters (`**`) inside the file?
- a. `find file -name '**'`                      b. `find file -name **`  
 c. `fgrep -c ** file`                      d. `fgrep -c '**' file`  
 e. `find -c ** file`
42. What is the link count of directory `dir` after these successful commands?
- ```
mkdir dir ; cd dir ; touch a b c ; mkdir d e
```
- a. 3 b. 2 c. 7 d. 5 e. 4
43. If directory `dir` contains only these five two-character names: `a?`, `11`, `?1`, `1*`, `.1`, then which command removes *only* the single two-character name `?1` from the directory?
- a. `rm dir/*1` b. `rm dir/??` c. `rm dir/1*`
 d. `rm dir/\??` e. `rm dir/?1`
44. In an empty directory, what is the output on your screen after this:
- ```
touch A a ; echo * >"*" ; ls
```
- a. `A a >*`                      b. `* >*`                      c. No output  
 d. `* A a`                      e. `A a >A a`
45. How many arguments are passed to the command by the shell:
- ```
echo ' one two ' three ' four ' 5'6'
```
- a. 4 b. 5 c. 1 d. 6 e. 9
46. What is the link count of directory `d` after these successful commands?
- ```
mkdir d ; touch f ; cd d ; ln ../f x
```
- a. 5                      b. 2                      c. 4                      d. 1                      e. 3



65. In an empty directory, how many words are in file **c** after this:  
`touch a b 1 b a ; ls >c`  
 a. 3            b. 1            c. 0            d. 2            e. 4
66. Which of these statements is true?  
 a. Only backslashes are strong enough to stop GLOB patterns from expanding.  
 b. Only double quotes are strong enough to stop GLOB patterns from expanding.  
 c. Only single quotes are strong enough to stop GLOB patterns from expanding.  
 d. If **/x** is an empty directory, `sort /x/*` produces an error message.  
 e. If **/y** is an empty directory, `echo /y/*` produces an error message.
67. How many arguments are passed to the command by the shell:  
`echo 'It's a bird! No! It's a plane!'`  
 a. 4            b. 3            c. 1            d. 2            e. 5
68. 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`  
 a. **foo** and **bar** are names for different files  
 b. **foo** and **bar** are two of three names for the same file  
 c. this output is not possible  
 d. **foo** and **bar** are names for the same file  
 e. **foo** and **bar** each have three names (six names total)
69. How many arguments are passed to the command by the shell:  
`<bar bar -b "-a" '-r' >bar bar bar`  
 a. 4            b. 5            c. 7            d. 6            e. 3
70. Rewrite as a simplified absolute path:  
`/usr/./bin/./lib/./././etc/./usr/./lib/./bin/./bar`  
 a. `/etc/bar`            b. `/bar`            c. `/usr/bin/bar`  
 d. `/usr/lib/bar`        e. `/usr/bar`
71. How many arguments are passed to the command by the shell:  
`echo 'And it's not hard, it's just logical.'`  
 a. 5            b. 3            c. 4            d. 6            e. 7
72. 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. 3            b. 1            c. 5            d. 2            e. 4
73. 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"`
74. What is the link count of directory **d** after these successful commands?  
`mkdir d ; cd d ; touch a ; mkdir b c`  
 a. 5            b. 3            c. 6            d. 4            e. 2

75. 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\*`
76. 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. 3            b. 5            c. 2            d. 4            e. 1
77. 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. 21            b. 18            c. 2            d. 10            e. 6
78. 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. **one** followed by **two**  
 b. **one**  
 c. **two**  
 d. **two** followed by **one**  
 e. **bash: xxx: command not found**
79. Which command finds your account login userid in the password file?  
 a. `find /etc/passwd -name $USER`  
 b. `cat $USER /etc/passwd`  
 c. `find $USER /etc/passwd`  
 d. `fgrep /etc/passwd $USER`  
 e. `fgrep $USER /etc/passwd`
80. What is the output on your screen after this: `echo hi >out | wc -w`  
 a. 1            b. no output            c. 3  
 d. 2            e. 0
81. 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. 3            b. 0            c. 4            d. 1            e. 2
82. 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`  
 a. 7            b. 10            c. 6            d. 4            e. 12

83. Which command line outputs inode/filename pairs for names in the current directory, sorted by inode number?
- a. `sort -n | ls -ai`                      b. `ls -ai | sort -n`  
 c. `ls -i * > sort -n`                      d. `ls .* | sort -node`  
 e. `ls -node * > sort -n`
84. What displays on your screen given this command:  
`ls >ls ; wc ls >wc ; sort ls | cat wc`
- a. `cat` reads the pipe and the `wc` and displays both together  
 b. `sort` displays the `ls` and `cat` displays the `wc`  
 c. only the `wc` displays because `cat` ignores the pipe  
 d. nothing displays because `cat` ignores the pipe  
 e. only the `ls` displays because `cat` ignores the pipe
85. 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. `bash: /bin/foo: command not found`  
 c. `dad`  
 d. `mom` followed by `dad`  
 e. `mom`
86. 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. 3                      b. 4                      c. 5                      d. 2                      e. 6
87. How many files are touched? `touch "1" 2 3 " " 4 5`
- a. 5                      b. 4                      c. 6                      d. 7                      e. 3
88. 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. 789                      b. 123                      c. 456                      d. 787                      e. 454
89. In an empty directory, how many words are in file `c` after this:  
`touch a ; mv b a >b ; ls >c`
- a. 2                      b. 3                      c. 1                      d. 0                      e. 4
90. In an empty directory, what is the output on your screen after this:  
`touch 1 2 3 ; cow="*" ; echo $cow`
- a. `*`                      b. `"1 2 3"`                      c. `$cow`  
 d. `"*"`                      e. `1 2 3`
91. 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. 2                      c. 1                      d. 3                      e. 0

92. A "dangling symlink" is a symlink to:
- a. a non-existent target                      b. the current directory  
 c. a directory                      d. a special device file  
 e. a parent directory
93. 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. no output                      b. 1                      c. 0  
 d. 2                      e. a
94. Which command line shows the current date?
- a. `echo date | bash`                      b. `date | bash`  
 c. `bash <date`                      d. `bash date`  
 e. `bash >date ; cat date`
95. 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. 0                      b. 5                      c. 2                      d. 4                      e. 6
96. How many arguments are passed to the command by the shell:  
`echo "1 2 " three ' 4 ' five"6"`
- a. 1                      b. 3                      c. 4                      d. 5                      e. 9
97. 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 command:  
`cat /etc/passwd`
- a. `cat: /etc/passwd: command not found`  
 b. `bash: /bin/sh: command not found`  
 c. `bash: /bin/cat: no such file or directory`  
 d. `bash: cat: command not found`  
 e. `cat: bash: no such file or directory`
98. What is in the local variable `$$` ?
- a. the command name of the previous command line  
 b. the process ID of the current shell  
 c. the first argument of the previous command line  
 d. the cpu cost of the current session, in dollars  
 e. `$$` is not a valid variable name
99. 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. 3                      b. 4                      c. 5                      d. 2                      e. 1
100. 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.'`

101. How many arguments are passed to the command by the shell:  
`echo " 1 '2 3' 4 "5 6 ' 7 "8 ' >out`  
 a. 2            b. 4            c. 3            d. 6            e. 5
102. What is true about this output from `ls -il foo bar`  
`15 -r-x-----x 2 me me 3 Jan 1 1:00 foo`  
`15 -r-x-----x 2 me me 3 Jan 1 1:00 bar`  
 a. `foo` and `bar` each have three names (six names total)  
 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` are names for different files  
 e. this output is not possible
103. Which command line 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=.$HOME:/usr/bin`  
 c. `$PATH=/usr/bin:./bin`            d. `PATH=/bin:/usr/bin:.`  
 e. `PATH=.$HOME:/usr/bin`
104. 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. 2            b. 3            c. 4            d. 5            e. 1
105. 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. 8            b. 0            c. 7            d. 5            e. 12
106. How many arguments are passed to the command by the shell:  
`<cow cow "-x" -y '-z' >cow cow`  
 a. 5            b. 6            c. 3            d. 2            e. 4
107. 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. 5            b. 2            c. 1            d. 3            e. 4
108. 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. 4            d. 0            e. 3
109. What is the link count of directory `dir` after these successful commands?  
`mkdir dir ; touch foo ; cd dir ; ln ../foo bar`  
 a. 2            b. 4            c. 1            d. 3            e. 5
110. 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. 0            b. 2            c. 3            d. 1            e. 4

111. Which command line 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'`
112. 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 followed by 2  
 d. 3            e. 2 followed by 3
113. 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. `one` followed by `two`  
 b. `one`  
 c. `two`  
 d. `two` followed by `one`  
 e. `bash: foo: command not found`
114. 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 count of links (names) this directory has.  
 b. The number 32 is the count of links (names) this directory has.  
 c. The number 128 is the size of this directory.  
 d. The number 32 is the inode number of this directory.  
 e. The number 128 is the inode number of this directory.
115. In an empty directory, what is the output on your screen after this:  
`echo one >.bar ; echo .*`  
 a. `one`  
 b. `.bar`  
 c. `. .. .bar`  
 d. `.*`  
 e. an error message from `echo` saying `.*` does not exist
116. 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. `one` followed by `ten` and `two`            b. `ten`  
 c. no output on screen            d. `one`  
 e. `two`
117. 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. `nosuchfile`  
 d. `hi`  
 e. `ls: cannot access nosuchfile`

118. Rewrite as a simplified absolute path:  
`../../../../var/./a/../../../../var/b/../../../../etc/./bar/./foo`  
 a. `/etc/bar/foo`      b. `/var/b/foo`      c. `/var/foo`  
 d. `/var/a/foo`      e. `/etc/foo`
119. What is in file `foo` after this:  
`echo hi >a ; ln a b ; echo me >b ; ln a foo ; rm a b`  
 a. no such file (nonexistent)      b. `me`  
 c. `hi`      d. nothing (empty file)  
 e. `hi` followed by `me`
120. 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. 1      d. 2      e. 5
121. 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. 2      b. 3      c. 4      d. 5      e. 1
122. If your `PATH` variable contains `/bin:/usr/bin`, what is the output of this:  
`echo $PATH`  
 a. `/bin:/usr/bin`  
 b. `/bin:/usr/bin`  
 c. `$PATH`  
 d. `echo: $PATH: No such file or directory`  
 e. `$PATH`
123. 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. 1      b. 4      c. 2      d. 3      e. 0
124. Rewrite as a simplified absolute path:  
`/home/me/../../../../etc/./home/me/./you/./me/./foo`  
 a. `/etc/foo`      b. `/home/me/foo`  
 c. `/home/foo`      d. `/home/you/foo`  
 e. `/foo`

125. 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. 3      b. 4      c. 0      d. 1      e. 2
126. How many arguments are passed to the command by the shell:  
`echo " 1 2 "three ' 4 ' five"6"`  
 a. 5      b. 9      c. 1      d. 3      e. 4
127. How many files are touched? `touch 1 "2 3" ' ' 4 5`  
 a. 3      b. 6      c. 5      d. 4      e. 7
128. 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. 3      b. 5      c. 4      d. 1      e. 2
129. 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. 5      d. 4      e. 2
130. What is the link count of directory `dir` after these successful commands?  
`mkdir dir ; cd dir ; touch one ; mkdir two`  
 a. 3      b. 4      c. 1      d. 5      e. 2
131. Dereference the following symlink `xyz` into its equivalent absolute path:  
`ln -s ../../a/./b/./bar /tmp/a/b/xyz`  
 a. `/tmp/a/bar`      b. `/tmp/a/b/bar`      c. `/tmp/bar`  
 d. `/tmp/b/xyz`      e. `/tmp/b/bar`
132. 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`
133. Which option to `ls` displays the directory itself and not its contents?  
 a. `-a`      b. `-d`      c. `-i`      d. `-l`      e. `-R`
134. 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. 2      b. 3      c. 4      d. 5      e. 1



135. In an empty directory, what is the output on your screen after this:  
`echo hi >a ; ls | wc -w`  
 a. a                      b. 0                      c. 2  
 d. 1                      e. no output
136. If `mt` is an empty sub-directory, what is true after this:  
`touch bar ; mkdir foo ; mv mt/./bar mt/foo`  
 a. the directory `mt` now contains a file named `foo`  
 b. the directory `mt` is still empty  
 c. the command fails because `mt/foo` is not a directory  
 d. the directory `foo` now contains a file named `bar`  
 e. the directory `mt` now contains a file named `bar`
137. How many arguments are passed to the command by the shell:  
`<foo foo -x " " -z -r" " >foo 'foo foo'`  
 a. 6                      b. 5                      c. 9                      d. 7                      e. 8
138. What is the link count of directory `foo` after these successful commands?  
`mkdir foo ; cd foo ; touch a b c`  
 a. 2                      b. 3                      c. 4                      d. 5                      e. 1
139. What is the output of this in an empty directory:  
`date >.date ; users >.users ; echo .?*`  
 a. .\*  
 b. an error message from `echo` saying `.?*` does not exist  
 c. .. `.date` `.users`  
 d. `.date`  
 e. `.date` `.users`
140. In an empty directory, what is the output on your screen after this:  
`touch 1 2 3 ; cow="*" ; echo "$cow"`  
 a. \*                      b. 1 2 3                      c. "1 2 3"  
 d. \$cow                      e. "\$cow"
141. 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. 4                      c. 2                      d. 1                      e. 5
142. What command will recursively find all pathnames named `foo` in `/bin`?  
 a. `find /bin -name 'foo'`  
 b. `find foo -name '/bin'`  
 c. `grep 'foo' /bin`  
 d. `grep /bin -basename 'foo'`  
 e. `ls -R 'foo' /bin`
143. 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

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-- 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. 4                      b. 3                      c. 0                      d. 1                      e. 2
145. Which command shows names under directory `oldnotes` containing `RTFM` anywhere in the name?  
 a. `ls oldnotes RTFM *`                      b. `ls oldnotes *RTFM*`  
 c. `ls oldnotes/*RTFM*`                      d. `ls oldnotes/RTFM*`  
 e. `ls oldnotes*RTFM*`
146. Which one of these names is usually a shell environment variable?  
 a. `FOOBAR`                      b. `fooBar`                      c. `foobar`  
 d. `FooBar`                      e. `FooBar`
147. If directory `/a` contains these seven two-character names: `aa`, `ab`, `ac`, `ad`, `a?`, `a*`, `a.`, then which command 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/a*`
148. 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. 4                      c. 2                      d. 0                      e. 3
149. 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. no output on screen                      b. `one`  
 c. `ten`                      d. `one` followed by `two` and `ten`  
 e. `two`
150. 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`  
 a. `/bin/ls: bat: No such file or directory`  
 b. `bat`  
 c. no output on screen  
 d. `foo`  
 e. `bash: /bin/ls: command not found`

151. In an empty directory, what is the output on your screen after this:  
`echo one >.bar ; echo .*?`  
 a. one  
 b. .\*?  
 c. an error message from `echo` saying `.*?` does not exist  
 d. .bar  
 e. .. .bar
152. 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 directory `mt` now contains only a file named `me`  
 c. the command fails because the name `mt/./bar/me` does not exist  
 d. the `mkdir` fails because `bar` already exists  
 e. there is a second copy of the file `bar` in the file named `me`
153. To change to the parent directory, do this:  
 a. `cd ..`  
 b. `cd .`  
 c. `pwd ..`  
 d. `pwd`  
 e. `cd`
154. If directory `/a` contains these seven two-character names: `aa`, `ab`, `ac`, `ad`, `a?`, `a*`, `a.`, then which command removes *only* the single two-character name `a*` from the directory?  
 a. `rm /a/*`  
 b. `rm /a*`  
 c. `rm /a/a?`  
 d. `rm /a/a\*`  
 e. `rm /a/a*`
155. 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. 4  
 d. 3  
 e. 2
156. What is usually in the environment variable `$SHELL`?  
 a. the relative path of the system `/shell` directory  
 b. the relative path of the `/home/shell` directory  
 c. the absolute path of the system `/shell` directory  
 d. the absolute path of your login shell  
 e. the relative path of your login shell
157. 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. 2  
 d. 1  
 e. 5

158. 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. no output  
 b. 2  
 c. 1 followed by 2  
 d. 1  
 e. 2 followed by 1
159. In an empty directory, how many words are in file `a` after this:  
`echo It's redirected >b isn't it\? ; ls >a`  
 a. 3  
 b. 4  
 c. 2  
 d. 0  
 e. 1
160. What is in file `c` after this:  
`echo B >b ; ln b a ; echo A >a ; ln a c ; rm a b`  
 a. no such file (nonexistent)  
 b. `A` followed by `B`  
 c. nothing (empty file)  
 d. `A`  
 e. `B`
161. What is usually in the environment variable `$PATH`?  
 a. the absolute path of your login home directory  
 b. a colon-separated list of directories containing command names  
 c. the absolute path of the system `/path` directory  
 d. a colon-separated list of your `passwd` file fields  
 e. the absolute path of your login shell
162. In an empty directory, what is the output on your screen after this:  
`touch 1 2 3 ; cow="*" ; echo '$cow'`  
 a. '\$cow'  
 b. 1 2 3  
 c. \*  
 d. \$cow  
 e. '1 2 3'
163. Which command usually goes in your `.bash_profile` file?  
 a. `source ./bashrc`  
 b. `cat .bashrc`  
 c. `.bashrc source`  
 d. `.bash_profile source`  
 e. `source ./bash_profile`
164. What is the output on your screen after this:  
`echo 1 >x ; ln x y ; echo 2 >>y ; sort x`  
 a. 2  
 b. no output  
 c. 2 followed by 1  
 d. 1  
 e. 1 followed by 2
165. 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`  
 a. 5  
 b. 4  
 c. 2  
 d. 1  
 e. 3
166. If directory `/a` contains these seven two-character names: `aa`, `ab`, `ac`, `ad`, `a*`, `a?`, `??`, then which command removes *only* the single two-character name `a?` from the directory?  
 a. `rm "/a?"`  
 b. `rm /a/a?`  
 c. `rm /a\?`  
 d. `rm /a/?\?`  
 e. `rm '/a/a?'`

167. 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. `mom`  
 c. `mom` followed by `hi`  
 d. `bash: foo: command not found`  
 e. `hi`
168. 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. 2                    b. 0                    c. 4                    d. 3                    e. 5
169. 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. 4                    b. 5                    c. 0                    d. 2                    e. 3
170. 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. 3                    c. 4                    d. 5                    e. 2
171. What is the usual output on your screen of this:  
`mkdir dir ; cd dir >foo ; cat foo`  
 a. `cat: foo: No such file or directory`  
 b. `foo`  
 c. `bash: cd: dir: No such file or directory`  
 d. `dir`  
 e. no output
172. 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. `bash: pig: command not found`  
 d. `foo` followed by `hi`  
 e. `hi`
173. 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`  
 a. `dad`  
 b. `bash: foo: command not found`  
 c. `mom` followed by `dad`  
 d. `mom`  
 e. `dad` followed by `mom`

174. How many arguments are passed to the command by the shell:  
`<bat bat -b "-a -r" >bat bat bat`  
 a. 5                    b. 4                    c. 6                    d. 3                    e. 7
175. 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. 3                    b. 5                    c. 6                    d. 9                    e. 2
176. 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. `/bar/foo`  
 d. `/foo`                    e. `/etc/bar/foo`
177. Which command shows *only* names that match `rtfm`, case-insensitive?  
 a. `echo *rtfmRTFM*`                    b. `echo [rR][tT][fF][mM]`  
 c. `echo [rR,tT,fF,mM]`                    d. `echo [rRtTfFmM]`  
 e. `echo *[rRtTfFmM]*`
178. 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 `/etc/group` file for next log in  
 c. put the alias into the `/bin/bash` file for next log in  
 d. create the alias and then type `save` to save it to all shells  
 e. define the alias in my file `$HOME/.bashrc`
179. 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. 3                    b. 0                    c. 1                    d. 4                    e. 2
180. 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 names for the same file  
 b. `foo` and `bar` each have two names (four names total)  
 c. this output is not possible  
 d. `foo` and `bar` are two of three names for this file  
 e. `foo` and `bar` each have three names (six names total)
181. How many arguments are passed to the command by the shell:  
`<foo foo " a 'b c' d " e f ' g " h " ' >foo`  
 a. 2                    b. 3                    c. 6                    d. 4                    e. 5
182. 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 directory `mt`  
 b. the directory `mt` is now empty  
 c. the directory `mt` now contains only a file named `me`  
 d. there is a second copy of the file `bar` in file `/tmp/me`  
 e. the command fails because the name `mt/../me` does not exist

183. 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`
- `foo`
  - `xx` followed by `foo`
  - `bash: pig: command not found`
  - `xx`
  - `foo` followed by `xx`
184. 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:
- `/bar`
  - `/bin/foo/bar`
  - `/bin/bar`
  - `/bin/bar/foo`
  - `/foo/bar`
185. What is the link count of directory `z` after these successful commands?  
`mkdir z ; cd z ; touch a b ; mkdir c d e`
- 4
  - 3
  - 5
  - 6
  - 7
186. What displays on your screen given this command:  
`date >date ; pwd >pwd ; head date | tail pwd`
- `head` displays the `date` and `tail` displays the `pwd`
  - `tail` reads the pipe and the `pwd` and displays both together
  - only the `pwd` displays because `tail` ignores the pipe
  - only the `date` displays because `tail` ignores the pipe
  - nothing displays because `tail` ignores the pipe
187. In an empty directory, what is the output on your screen after this:  
`echo hi >foo ; cp foo bar | wc -w`
- no output
  - 2
  - 0
  - 1
  - 3
188. Which environment variable contains your HOME directory?
- `$/HOME`
  - `$home`
  - `/home/abcd0001`
  - `/home`
  - `$HOME`
189. If directory `/a` contains these seven two-character names: `aa`, `ab`, `ac`, `ad`, `a?`, `a*`, `a.`, then which command removes *only* the single two-character name `a*` from the directory?
- `rm "/a/a*"`
  - `rm /a/*`
  - `rm /a*`
  - `rm /a/a?`
  - `rm /a/a*`
190. How many arguments are passed to the command by the shell:  
`<wc wc " 1 '2 3' 4 " 5 6 ' 7 " 8 " ' >wc 9`
- 5
  - 6
  - 3
  - 4
  - 2

191. Which command counts lines containing the string `refused` in only the month `October 2016` in the `denyhosts` log file?
- `fgrep -c 2016-10 denyhosts | fgrep refused`
  - `fgrep refused denyhosts | fgrep -c October 2016`
  - `fgrep refused denyhosts ; fgrep -c 2016-10`
  - `fgrep 'refused' denyhosts | fgrep -c '2016-10'`
  - `fgrep '2016-10 refused' denyhosts`
192. 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`
- 4
  - 5
  - 2
  - 1
  - 3
193. In an empty directory, what is in file `out` after this:  
`ls nosuchfile | wc -w >out`
- nothing (empty file)
  - 1
  - `out`
  - 0
  - `nosuchfile`
194. Which command appends directory `/bin` to your search path?
- `$PATH=PATH:/bin`
  - `PATH=PATH+/bin`
  - `PATH=$PATH:/bin`
  - `PATH=PATH:/bin`
  - `$PATH=$PATH:/bin`
195. 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`
- `one` followed by `two`
  - `two`
  - `bash: foo: command not found`
  - `one`
  - `two` followed by `one`
196. 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`
197. How many arguments are passed to the command by the shell:  
`<foo foo " a 'b c' d " e ' f " g " ' >foo h`
- 4
  - 5
  - 2
  - 3
  - 6
198. What is the link count of directory `z` after these successful commands?  
`mkdir z ; cd z ; touch a ; ln a b ; ln a c`
- 2
  - 4
  - 3
  - 1
  - 5

199. 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 foo ../dir2`                      d. `cp .. dir2 foo`  
 e. `cp foo dir2 ..`
200. How many arguments are passed to the command by the shell:  
`<cow cow "-x "-y '-z' >cow cow`
- a. 5                      b. 6                      c. 7                      d. 4                      e. 3
201. If I have a directory named **/1/2**, which action would increase its *link count* by exactly one?
- a. create a directory named **/1/2**  
 b. create one file named **/1/22**  
 c. create a directory named **/1/22**  
 d. create one file named **/1/2/3**  
 e. create a directory named **/1/2/3**
202. How many arguments are passed to the command by the shell:  
`<foo foo " a 'b c' d " e ' f " g " ' >foo`
- a. 3                      b. 4                      c. 2                      d. 5                      e. 6
203. 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/bar**                      b. **/etc/bar/foo**                      c. **/foo**  
 d. **/bar/foo**                      e. **/etc/foo**
204. 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. 5                      b. 4                      c. 6                      d. 2                      e. 3
205. Which file is a DOS/Windows file?
- a. ASCII text, with LF line terminators  
 b. ASCII text, with CR line terminators  
 c. ASCII text, with CRLF line terminators  
 d. ASCII text, with no line terminators  
 e. ASCII text
206. How do you execute the program **foo** in the current directory?
- a. `$HOME/foo`                      b. `foo/.`                      c. `./foo`  
 d. `/foo`                      e. `foo/`
207. How many files are touched? `touch "1 " 2 3" " ' ' 4 5`
- a. 4                      b. 7                      c. 3                      d. 6                      e. 5

208. 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. **foo** and **bar** are names for the same file
 b. **foo** and **bar** are names for different files
 c. **foo** and **bar** each have three names (six names total)
 d. this output is not possible
 e. **foo** and **bar** are two of three names for the same file
209. How many arguments are passed to the command by the shell:
`echo 'It's "1 2" isn't it? I can't decide.`
- a. 5 b. 6 c. 3 d. 4 e. 2
210. In an empty directory, what is the output on your screen after this:
`touch A a ; echo * ">*`
- a. **A a >A a** b. **A a >*** c. *** >***
 d. **A a** e. No output
211. What is the output on your screen after this:
`echo one >x ; ln x y ; echo two >>y ; sort x`
- a. **one** b. **two** followed by **one**
 c. **one** followed by **two** d. no output
 e. **two**
212. 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. **\$cow**
213. How many files are touched? `touch 1 "2 3 ' 4 '" 5`
- a. 3 b. 1 c. 2 d. 4 e. 5
214. Which of the following **PATH** statements makes the most sense?
- a. `PATH=/bin:/usr/bin`
 b. `PATH=/bin:/usr/bin:/etc/passwd`
 c. `PATH=/bin/bash:/usr/bin:/bin`
 d. `PATH=/bin:/etc/passwd:/usr/bin`
 e. `PATH=/bin/ls:/etc/passwd:/usr/bin`
215. How many arguments are passed to the command by the shell:
`echo " one '2 three' 4 "five 6 ' 7 "8 ' >out`
- a. 3 b. 4 c. 6 d. 5 e. 2
216. 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**
 c. **two** d. **one** followed by **two**
 e. no output

217. What is the link count of an empty directory?
a. 3 *b.* 4 *c.* 0 *d.* 1 *e.* 2
218. 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. 1 *b.* 0 *c.* 4 *d.* 3 *e.* 2
219. In an empty directory, what is the output on your screen after this:

```
echo one >.bar ; echo .????*
```

a. .. .bar
b. .bar
c. an error message from `echo` saying `.????*` does not exist
d. .????*
e. one
220. What command will recursively show disk usage in directories?
a. `find` *b.* `du` *c.* `ls` *d.* `df` *e.* `tree`
221. 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.* `/dir/bar`
c. `/foo/dir/bar` *d.* `/bin/dir/bar`
e. `/bin/dir/bar/foo`
222. 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. 3 *b.* 2 *c.* 1 *d.* 4 *e.* 5
223. 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. `bash: bat: command not found`
c. `bar`
d. `bar` followed by `foo`
e. `foo`
224. How many files are touched? `touch '1' '2 3' '4' '5'`
a. 1 *b.* 5 *c.* 2 *d.* 3 *e.* 4
225. What is in file `c` after this:

```
echo A >a ; ln a b ; echo B >b ; ln a c ; rm a b
```

a. nothing (empty file) *b.* `A`
c. `B` *d.* `A` followed by `B`
e. no such file (nonexistent)

226. How many arguments are passed to the command by the shell:

```
echo "cow "y " bat 'man x' " pig'a "hop' a b
```

a. 5 *b.* 6 *c.* 11 *d.* 4 *e.* 7
227. If `mt` is an empty sub-directory, what is true after this:

```
touch foo ; mkdir bar ; mv foo bar/mt
```

a. the directory `mt` is still empty
b. the command fails because `bar/mt` is not a directory
c. the directory `bar` now contains a file named `foo`
d. the directory `mt` now contains a directory named `bar`
e. the directory `mt` now contains a file named `foo`
228. 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. `dad` followed by `mom`
c. `mom`
d. `bash: foo: command not found`
e. `dad`
229. 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 32 is the size of this directory.
b. The number 32 is the count of links (names) this directory has.
c. The number 128 is the inode number of this directory.
d. The number 128 is the size of this directory.
e. The number 32 is the inode number of this directory.
230. 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: cat: command not found`
b. no output on screen
c. `bash: /bin/cat: command not found`
d. `bash: ls: command not found`
e. `/bin/cat: foo: No such file or directory`
231. 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/bar` *b.* `/bin/dir/foo`
c. `/bin/bar/dir/foo` *d.* `/bar/../dir/foo`
e. `/dir/foo`
232. Which command recursively finds all things with names beginning with `foo`?
a. `fgrep -name foo?` *b.* `find -name foo?`
c. `find -name 'foo*'` *d.* `ls foo*`
e. `ls -name foo*`

233. What is in file **out** after this:

```
echo me >a ; ln a b ; echo hi >b ; ln a out ; rm a b
```

- a. **me** followed by **hi**
- b. no such file (nonexistent)
- c. **hi**
- d. **me**
- e. nothing (empty file)

234. 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
```

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

235. What does *quoting* mean on a shell command line?

- a. turning off the special meaning of shell meta-characters
- b. typing a "control" character using the [CTRL] key
- c. using a leading tilde ("~") on a pathname to mean your **HOME** directory
- d. using more than one pathname argument to a command, e.g. **rm a b c**
- e. setting the **PS1** variable to be your shell prompt

236. 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 two of three names for the same file
- e. **foo** and **bar** are names for different files

237. **Did you read all the words of the test instructions on page one?**

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

This page intentionally left blank.