

PRINT Name: _____ LAB Section:

Test Version: 537 One-Answer Multiple Choice 173 Questions – 15 of 15%

- ⇒ Read **all** the words of these instructions and **both** sides (back and front) of all pages.
- ⇒ Use your full, unabbreviated name on the mark-sense form. Do not abbreviate your name.
- ⇒ Put the three-digit **Test Version** above into **NO. OF QUESTIONS** and **NO. OF STUDENTS**
- ⇒ Fill in the bubbles with pencil only, no pen. Enter your NAME, Test Version, and answers.
- ⇒ Manage your time. Answer questions you know, first. One Answer per question.
- ⇒ The answer to the questions below about reading/doing all these test instructions is: **Jes**

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

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

2. **My three-digit Lab Section number is:**

- a. The Test Version number printed in the top left corner.
 b. My lecture section number **400** or **410**.
 c. My lab room number, e.g. **B384**.
 d. My lecture room number, e.g. **T130**.
 e. The section number of my weekly 2-hour lab period.

3. Which command line allows programs in the current directory to execute without preceding the names with `./`?

- a. `$PATH=/usr/bin:./bin` b. `PATH=./$HOME:/usr/bin`
 c. `PATH=/usr/bin/.$HOME` d. `PATH=/bin:/usr/bin:.`
 e. `$PATH=.$HOME:/usr/bin`

4. If you are in `/etc` and `ls -l` shows a symbolic link `bar -> foo` then dereference the absolute path of `bar` with no symbolic links:

- a. `/foo` b. `/etc/foo` c. `/bar/foo`
 d. `/etc/foo/bar` e. `/etc/bar/foo`

5. What is the link count of file `foo` after these successful commands?

- `rm foo ; touch foo ; ln foo bar`
`cp bar x ; ln x y ; ln bar z ; ln z a`
 a. 3 b. 2 c. 4 d. 1 e. 5

6. What is the output on your screen after these command lines:

- `echo one >x ; ln x y ; echo ten >y`
`echo two >x ; cat y`
 a. **ten** b. **two**
 c. **one** followed by **ten** and **two** d. **one**
 e. no output on screen

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

- `<bat bat -b "-a -r" >bat bat bat`
 a. 7 b. 4 c. 5 d. 6 e. 3

8. The option to `ls` that shows inode (index) numbers is:

- a. `-x` b. `-l` c. `-i` d. `-1` e. `-a`

9. What is usually in the environment variable `$HOME`?

- a. the relative path of your login home directory
 b. the relative path of the system `/home` directory
 c. the relative path of the `ROOT` directory
 d. the absolute path of your login home directory
 e. the absolute path of the system `/home` directory

10. 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. only the `ls` displays because `cat` ignores the pipe
 c. only the `wc` displays because `cat` ignores the pipe
 d. `cat` reads the pipe and the `wc` and displays both together
 e. `sort` displays the `ls` and `cat` displays the `wc`

11. What is the link count of directory `dir` after these successful commands?

- `mkdir dir ; cd dir ; touch foo ; mkdir a b c`
 a. 2 b. 5 c. 4 d. 1 e. 3

12. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this command line: `ln a e ; ln b d ; ln d c ; cat e b >c`

- a. 4 b. 5 c. 2 d. 0 e. 3

13. 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. 5 c. 3 d. 4 e. 2

14. What is the link count of directory `dir` after these successful commands?

- `mkdir dir ; touch foo ; cd dir ; ln ../foo bar`
 a. 1 b. 4 c. 2 d. 3 e. 5

15. What is the output on your screen after these command lines:

- `echo one >x ; ln x y ; echo two >y`
`echo ten >x ; cat y`
 a. **one** followed by **two** and **ten** b. no output on screen
 c. **two** d. **ten**
 e. **one**

16. What is the output on your screen after this command line:

- `echo 1 >x ; ln x y ; echo 2 >>y ; sort x`
 a. 1 followed by 2 b. 2 followed by 1 c. no output
 d. 2 e. 1

17. If the file **bat** contained the word **foo**, what would be the output on your screen of this two command sequence:
`PATH=/bin/cat:/bin/who:/bin/ls ; cat bat`
 a. **bash: cat: command not found**
 b. no output on screen
 c. **bat**
 d. **foo**
 e. **cat: bat: No such file or directory**
18. How many arguments are passed to the command by the shell:
`echo " 1 2 " three ' 4 ' five"6"`
 a. 5 b. 4 c. 9 d. 1 e. 3
19. If `/bin/bat` is a program that outputs **foo** and `/usr/bin/bat` is a program that outputs **hi** what would be the output on your screen of this two command sequence: `PATH=/usr:/usr/bin:/bin ; bat`
 a. **hi**
 b. **foo** followed by **hi**
 c. **hi** followed by **foo**
 d. **bash: bat: command not found**
 e. **foo**
20. Which command usually goes in your `.bash_profile` file?
 a. `.bash_profile source` b. `cat .bashrc`
 c. `source ~/.bashrc` d. `source ~/.bash_profile`
 e. `.bashrc source`
21. What is the link count of directory **dir** after these successful commands?
`mkdir dir ; cd dir ; touch a b c ; mkdir d e`
 a. 7 b. 5 c. 4 d. 3 e. 2
22. In an empty directory, what is the output on your screen after this command line:
`echo hi >a ; mv a b ; ln b c ; ls >wc -l`
 a. **a** b. 0 c. no output
 d. 1 e. 2
23. What is the output of this command line if run in an empty directory:
`touch A a ; echo * >"*" ; ls`
 a. *** >*** b. **A a >A a** c. *** A a**
 d. **A a >*** e. No output
24. 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/?`
25. How many arguments are passed to the command by the shell:
`echo " 1 '2 3' 4 "5 6 ' 7 "8 ' >out`
 a. 2 b. 5 c. 4 d. 6 e. 3

26. 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/a*"` c. `rm /a/a?`
 d. `rm /a/*` e. `rm /a*`
27. 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?`
28. If I have a directory named `/1/2`, which action would increase its *link count* by exactly one?
 a. create a directory named `/1/22`
 b. create one file named `/1/22`
 c. create a directory named `/1/2/3`
 d. create a directory named `/1/2`
 e. create one file named `/1/2/3`
29. 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. 1 c. 4 d. 2 e. 3
30. How many arguments are passed to the command by the shell:
`<foo foo " a 'b c' d " e f ' g " h " ' >foo`
 a. 6 b. 2 c. 4 d. 5 e. 3
31. Rewrite as a simplified absolute path:
`/home/me/../../you/../../../../etc/../../home/me/../../you/../../me/../../foo`
 a. `/home/you/foo` b. `/etc/foo`
 c. `/home/me/foo` d. `/foo`
 e. `/home/foo`
32. What is the link count of directory **foo** after these successful commands?
`mkdir foo ; cd foo ; touch a b c`
 a. 4 b. 5 c. 3 d. 1 e. 2
33. 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. 5 b. 3 c. 1 d. 4 e. 2
34. 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. 5 b. 1 c. 3 d. 4 e. 2
35. How many arguments are passed to the command by the shell:
`<wc wc " 1 '2 3' 4 " 5 6 ' 7 " 8 " ' >wc 9`
 a. 4 b. 5 c. 6 d. 3 e. 2

50. If `/bin/xxx` is a program that outputs `one` and `/usr/bin/xxx` is a program that outputs `two`, what would be the output on your screen of this two command sequence: `PATH=/usr:/usr/bin:/etc:/bin ; xxx`
- `one`
 - `two` followed by `one`
 - `two`
 - `bash: xxx: command not found`
 - `one` followed by `two`
51. If `/bin/foo` is a program that outputs `hi` and `/usr/bin/foo` is a program that outputs `mom` what would be the output on your screen of this two command sequence: `PATH=/etc:/usr/bin:/bin ; foo`
- `mom`
 - `hi` followed by `mom`
 - `hi`
 - `mom` followed by `hi`
 - `bash: foo: command not found`
52. What is the link count of directory `d` after these successful commands?
`mkdir d ; mkdir d/a ; touch d/b`
- 1
 - 5
 - 3
 - 2
 - 4
53. How many arguments are passed to the command by the shell:
`<f z " a 'b c' d " 1 2 ' g " h " ' >z`
- 2
 - 4
 - 3
 - 6
 - 5
54. 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 `c` does not exist
 - directory `a` does not exist
 - the command `cp` is not in your search `PATH`
55. Which command line makes pathnames `/usr/local/bin` and `/usr/bin` lead to the same directory?
- `ln . /usr/local`
 - `ln -s . /usr/local`
 - `rmdir /usr/local`
 - `touch /usr/local`
 - `mkdir /usr/local`
56. In an empty directory, how many files are created by this command line:
`touch 1 "2 3" ' 4 ' 5`
- 7
 - 4
 - 5
 - 6
 - 3

57. 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 size of this directory.
 - The number 128 is the inode number of this directory.
 - The number 128 is the count of links (names) this directory has.
 - The number 32 is the inode number of this directory.
58. 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`
- `foo` and `bar` are two of three names for this file
 - this output is not possible
 - `foo` and `bar` are names for the same file
 - `foo` and `bar` each have three names (six names total)
 - `foo` and `bar` are names for different files
59. If `/bin/pig` is a program that outputs `hi` and `/usr/bin/pig` is a program that outputs `foo` what would be the output on your screen of this two command sequence: `PATH=/etc:/usr/bin:/bin ; pig`
- `foo`
 - `hi`
 - `bash: pig: command not found`
 - `foo` followed by `hi`
 - `hi` followed by `foo`
60. What is the link count of directory `d` after these successful commands?
`mkdir d ; touch f ; cd d ; ln ../f x`
- 4
 - 5
 - 3
 - 2
 - 1
61. If file `foo` occupies one disk block, how many disk blocks are in use after this sequence of commands:
`cp foo bar ; ln bar one ; cp one two ; ln one pig`
- 5
 - 3
 - 2
 - 4
 - 1
62. How do you execute the program `foo` in the current directory?
- `$HOME/foo`
 - `foo/`
 - `foo/.`
 - `./foo`
 - `/foo`
63. What is in file `c` after this command line:
`echo foo >a ; ln a b ; echo bar >>b ; ln a c ; rm a`
- `foo` followed by `bar`
 - `bar`
 - nothing (empty file)
 - no such file (nonexistent)
 - `foo`
64. What is the link count of file `foo` after these successful commands?
`rm foo ; touch foo ; ln foo bar`
`cp bar a ; ln a b ; ln bar c ; cp c a`
- 3
 - 1
 - 5
 - 4
 - 2

65. 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
```

 - `foo` and `bar` are names for the same file
 - `foo` and `bar` each have three names (six names total)
 - `foo` and `bar` are two of three names for the same file
 - `foo` and `bar` are names for different files
 - this output is not possible
66. Which of the following `PATH` statements makes the most sense?
 - `PATH=/bin/bash:/usr/bin:/bin`
 - `PATH=/bin/ls:/etc/passwd:/usr/bin`
 - `PATH=/bin:/usr/bin`
 - `PATH=/bin:/etc/passwd:/usr/bin`
 - `PATH=/bin:/usr/bin:/etc/passwd`
67. 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:
 - `/bin/bar/foo`
 - `/foo/bar`
 - `/bin/foo/bar`
 - `/bar`
 - `/bin/bar`
68. 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`?
 - 123
 - 789
 - 456
 - 787
 - 454
69. What is the link count of directory `x` after these successful commands?

```
mkdir x ; mkdir x/y ; mkdir x/z ; mkdir x/y/z
```

 - 4
 - 5
 - 3
 - 2
 - 1
70. Which command line shows just the count of words in the file?
 - `wc file | awk '{print $2}'`
 - `wc file | awk '{print #2}'`
 - `wc file | awk '[print #2]'`
 - `wc file | awk '[print $2]'`
 - `wc file | awk '{print 2}'`
71. How many arguments are passed to the command by the shell:

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

 - 6
 - 11
 - 4
 - 7
 - 5
72. 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*`
 - `rm /a/a*`
 - `rm /a/a*`
 - `rm /a/*`
 - `rm /a/a?`
73. What is the link count of directory `d` after these successful commands?

```
mkdir d ; mkdir d/a d/b ; touch d/c d/e
```

 - 1
 - 2
 - 5
 - 4
 - 3

74. What is the link count of an empty directory?
 - 4
 - 2
 - 1
 - 0
 - 3
75. How many arguments are passed to the command by the shell:

```
<pig pig -x " " -z -r" " >pig pig pig
```

 - 7
 - 5
 - 8
 - 9
 - 6
76. What is the link count of directory `d` after these successful commands?

```
mkdir d ; cd d ; touch f ; ln f a ; ln f b
```

 - 5
 - 2
 - 4
 - 3
 - 1
77. What displays on your screen given this command:

```
date >date ; pwd >pwd ; head date | tail pwd
```

 - `tail` reads the pipe and the `pwd` and displays both together
 - nothing displays because `tail` ignores the pipe
 - only the `pwd` displays because `tail` ignores the pipe
 - only the `date` displays because `tail` ignores the pipe
 - `head` displays the `date` and `tail` displays the `pwd`
78. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this command line: `ln a d ; ln b e ; cp d e >c`
 - 3
 - 5
 - 4
 - 2
 - 0
79. Which command removes *only* this four-character name containing a special character: `*xyz`
 - `rm '*xyz'`
 - `rm '*xyz''`
 - `rm *xyz`
 - `rm "*xyz"`
 - `rm "*xyz"`
80. Which command removes *only* this four-character name containing a special character: `?xyz`
 - `rm ?xyz`
 - `rm '?xyz'`
 - `rm '?xyz'`
 - `rm '?xyz'`
 - `rm ''?xyz''`
81. 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
```

 - 4
 - 1
 - 2
 - 3
 - 0
82. How many arguments are passed to the command by the shell:

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

 - 6
 - 3
 - 2
 - 5
 - 4
83. 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
```

 - 3
 - 5
 - 4
 - 2
 - 1
84. What is the link count of directory `dir` after these successful commands?

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

 - 3
 - 4
 - 5
 - 2
 - 1

85. Which command line shows the current date?
 a. `date | bash` b. `echo date | bash`
 c. `bash >date ; cat date` d. `bash date`
 e. `bash <date`
86. What is the output on your screen after this command line:
`echo hi >a ; cp a b | wc -c`
 a. no output b. 1 c. 3
 d. 2 e. 0
87. Which of these statements is true?
 a. Only backslashes are strong enough to stop GLOB patterns from expanding.
 b. If `/y` is an empty directory, `echo /y/*` produces an error message.
 c. Only single quotes are strong enough to stop GLOB patterns from expanding.
 d. Only double quotes are strong enough to stop GLOB patterns from expanding.
 e. If `/x` is an empty directory, `sort /x/*` produces an error message.
88. If `/bin/prg` is a program that outputs `hi` and `/usr/bin/prg` is a program that outputs `foo` what would be the output on your screen of this two command sequence: `PATH=/etc:/usr/bin:/bin ; prg`
 a. `foo` followed by `hi`
 b. `foo`
 c. `bash: prg: command not found`
 d. `hi`
 e. `hi` followed by `foo`
89. If `/bin/foo` is a program that outputs `one` and `/usr/bin/foo` is a program that outputs `two`, what would be the output on your screen of this two command sequence: `PATH=/etc:/usr/bin:/usr:/bin:/dev ; foo`
 a. `two`
 b. `two` followed by `one`
 c. `one` followed by `two`
 d. `bash: foo: command not found`
 e. `one`
90. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `d` (not in `c`) after this command line:
`ln a d ; ln d c ; ln c e ; cat a a b b c c d d e e >c`
 a. 21 b. 10 c. 2 d. 18 e. 6
91. What is the link count of directory `z` after these successful commands?
`mkdir z ; cd z ; touch a b ; mkdir c d e`
 a. 6 b. 3 c. 7 d. 5 e. 4
92. 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. 2 b. 4 c. 6 d. 3 e. 5

93. What is the link count of file `foo` after these successful commands?
`rm foo ; touch foo ; ln foo bar`
`cp bar x ; ln x y ; ln bar z`
 a. 4 b. 5 c. 1 d. 2 e. 3
94. 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. 2 d. 5 e. 1
95. Which command line would show the index (inode) number of a file?
 a. `cat -i file` b. `ls -l file` c. `find -i file`
 d. `cat -l file` e. `ls -i file`
96. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this command line: `ln a d ; ln d c ; cat a b >c`
 a. 5 b. 0 c. 4 d. 2 e. 3
97. How many arguments are passed to the command by the shell:
`<foo foo " a 'b c' d " e ' f " g " ' >foo`
 a. 2 b. 6 c. 4 d. 5 e. 3
98. Rewrite as a simplified absolute path:
`/usr/./bin/./lib/./../etc/./usr/./lib/./bin/./bar`
 a. `/usr/lib/bar` b. `/bar` c. `/etc/bar`
 d. `/usr/bar` e. `/usr/bin/bar`
99. In an empty directory, what is the output on your screen of this command line:
`echo hi >foo ; cp foo bar | wc -l`
 a. 0 b. 3 c. 1
 d. no output e. 2
100. The output of the `whoami` command is:
 a. a list of users logged in to the system
 b. your HOME directory
 c. the current directory
 d. your userid
 e. a list of accounts in the password file
101. If your terminal type is `xterm`, what is the output of this command line?
`echo '$TERM'`
 a. `'xterm'` b. no output on screen
 c. `xterm` d. `$TERM`
 e. `'$TERM'`
102. How many arguments are passed to the command by the shell:
`echo ' one two ' three ' four ' 5'6'`
 a. 5 b. 4 c. 1 d. 9 e. 6

103. 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. 2 b. 5 c. 1 d. 4 e. 3
104. What is in file **out** after this command line:
`echo me >a ; ln a b ; echo hi >b ; ln a out ; rm a b`
 a. nothing (empty file) b. **me** followed by **hi**
 c. no such file (nonexistent) d. **me**
 e. **hi**
105. 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. 4 b. 3 c. 6 d. 5 e. 2
106. What is usually in the environment variable **\$PATH**?
 a. the absolute path of the system **/path** directory
 b. the absolute path of your login shell
 c. a colon-separated list of your **passwd** file fields
 d. the absolute path of your login home directory
 e. a colon-separated list of directories containing command names
107. What is the output on your screen after this command line:
`echo one >x ; ln x y ; echo two >>y ; sort x`
 a. **two** b. **two** followed by **one**
 c. **one** d. **one** followed by **two**
 e. no output
108. In an empty directory, what is in file **count** after this command line:
`ls ??? | wc -w >count`
 a. 0 b. 1 1 2
 c. 1 d. nothing (empty file)
 e. 1 1 1
109. 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. 4 b. 2 c. 1 d. 5 e. 3
110. What is the output on your screen after this command line:
`echo hi >out | wc -l`
 a. 1 b. no output c. 2
 d. 3 e. 0
111. What is in file **foo** after this command line:
`echo hi >a ; ln a b ; echo me >b ; ln a foo ; rm a b`
 a. nothing (empty file) b. no such file (nonexistent)
 c. **hi** followed by **me** d. **me**
 e. **hi**

112. If **/bin/foo** is a program that outputs **one** and **/usr/bin/foo** is a program that outputs **two**, what would be the output on your screen of this two command sequence: `PATH=/bin/ls:/home:/usr/bin/cat:/etc ; foo`
 a. **one** followed by **two**
 b. **two** followed by **one**
 c. **bash: foo: command not found**
 d. **one**
 e. **two**
113. 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. 1 followed by 2 c. 2
 d. 2 followed by 1 e. 1
114. File **a** contains 2 lines. File **b** contains 3 lines. How many lines are in file **e** after this command line:
`ln a d ; cp a f ; ln d c ; ln c e ; cat a b d f >e`
 a. 3 b. 6 c. 5 d. 2 e. 9
115. If file **one** occupies one disk block, how many disk blocks are in use after this sequence of commands:
`cp one foo ; ln foo two ; ln two bar ; ln one cow`
 a. 1 b. 3 c. 4 d. 2 e. 5
116. What is the link count of directory **d** after these successful commands?
`mkdir d ; cd d ; touch a ; mkdir b c`
 a. 6 b. 5 c. 4 d. 2 e. 3
117. What is the link count of directory **dir** after these successful commands?
`mkdir dir ; mkdir dir/foo ; touch dir/bar`
 a. 5 b. 4 c. 3 d. 1 e. 2
118. 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 count of links (names) this directory has.
 b. The number 32 is the inode number of this directory.
 c. The number 128 is the inode number of this directory.
 d. The number 32 is the size of this directory.
 e. The number 128 is the size of this directory.
119. If file **foo** occupies one disk block, how many disk blocks are in use after this sequence of commands:
`cp foo bar ; ln bar one ; cp one two ; cp one xxx`
 a. 3 b. 5 c. 1 d. 4 e. 2

120. If `/bin/foo` is a program that outputs `dad` and `/usr/bin/foo` is a program that outputs `mom` what would be the output on your screen of this two command sequence: `PATH=/usr:/etc:/bin:/usr/bin ; foo`
- `dad`
 - `mom` followed by `dad`
 - `mom`
 - `bash: foo: command not found`
 - `dad` followed by `mom`
121. In an empty directory, what is the output on your screen after this command line:
`touch a ; ls | wc -l`
- no output
 - 2
 - 0
 - 1
 - 3
122. If `/bin/xxx` is a program that outputs `one` and `/usr/bin/xxx` is a program that outputs `two`, what would be the output on your screen of this two command sequence: `PATH=/bin/xxx:/usr/bin/xxx:/etc/passwd ; xxx`
- `one` followed by `two`
 - `two` followed by `one`
 - `two`
 - `bash: xxx: command not found`
 - `one`
123. 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:
- `/dir/bar`
 - `/foo/dir/bar`
 - `/bin/dir/bar`
 - `/bin/dir/bar/foo`
 - `/bin/foo/dir/bar`
124. What is the link count of file `foo` after these successful commands?
`rm foo ; touch foo ; ln foo bar ; ln bar x`
`cp bar a ; ln a b ; ln x c ; cp c d`
- 1
 - 5
 - 3
 - 2
 - 4
125. In an empty directory, what is in file `out` after this command line:
`ls nosuchfile | wc -l >out`
- 0
 - 1
 - nothing (empty file)
 - `nosuchfile`
 - `out`
126. What is in file `c` after this command line:
`echo B >b ; ln b a ; echo A >a ; ln a c ; rm a b`
- nothing (empty file)
 - B
 - no such file (nonexistent)
 - A
 - A followed by B

127. If `/bin/foo` is a program that outputs `mom` and `/usr/bin/foo` is a program that outputs `dad` what would be the output on your screen of this two command sequence: `PATH=/bin/foo:/usr/bin/foo:/usr ; foo`
- `mom`
 - `dad` followed by `mom`
 - `bash: foo: command not found`
 - `mom` followed by `dad`
 - `dad`
128. What is the output on your screen after this two-command sequence if run in a directory containing 8 files with names that are all the numbers from 1 to 8 inclusive: `cow="" ; echo '$cow'`
- `'$cow'`
 - the file names 1 through 8, surrounded by quotes
 - the file names 1 through 8
 - *
 - `$cow`
129. If your `PATH` contained only the file names `/bin/sh`, `/bin/cat`, and `/bin/ls`, then what would be the output on your screen of this command:
`cat /etc/passwd`
- `bash: cat: command not found`
 - `bash: /bin/sh: command not found`
 - `bash: /bin/cat: no such file or directory`
 - `cat: bash: no such file or directory`
 - `cat: /etc/passwd: command not found`
130. 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`
- 3
 - 4
 - 5
 - 2
 - 6
131. How many arguments are passed to the command by the shell:
`<bar bar -b "-a" '-r' >bar bar bar`
- 5
 - 6
 - 3
 - 7
 - 4
132. If the file `bat` contained the word `foo`, what would be the output on your screen of this two command sequence:
`PATH=/etc/passwd:/bin/ls:/bin/cat ; /bin/ls bat`
- `bat`
 - `/bin/ls: bat: No such file or directory`
 - `bash: /bin/ls: command not found`
 - `foo`
 - no output on screen
133. What is the resulting link count of empty directory `dir` after these successful commands? `cd dir ; touch foo ; ln foo one ; ln foo two`
- 1
 - 2
 - 4
 - 5
 - 3

134. 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
```
- this output is not possible
  - `foo` and `bar` each have three names (six names total)
  - `foo` and `bar` are two of three names for this file
  - `foo` and `bar` are names for the same file
  - `foo` and `bar` each have two names (four names total)
135. 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
```
- 3
 - 1
 - 5
 - 2
 - 4
136. How many arguments are passed to the command by the shell:
- ```
<foo foo -x " " -z -r" " >foo 'foo foo'
```
- 9
  - 6
  - 5
  - 8
  - 7
137. Which option to `ls` displays the directory itself and not its contents?
- `-a`
  - `-l`
  - `-i`
  - `-R`
  - `-d`
138. How many arguments are passed to the command by the shell:
- ```
echo 'It's "1 2" isn't it? I can't decide.'
```
- 2
 - 4
 - 5
 - 3
 - 6
139. File `a` contains 2 lines. File `b` contains 3 lines. How many lines are in file `c` after this command line: `ln a d ; ln d e ; ln b f >c`
- 0
 - 4
 - 2
 - 5
 - 3
140. 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:
- `/bin/dir/foo`
 - `/bin/bar/dir/foo`
 - `/bar/../dir/foo`
 - `/dir/foo`
 - `/bin/dir/foo/bar`
141. If `foo` were a readable empty file, what would be the output on your screen of this two command sequence:
- ```
PATH=/etc/passwd:/bin/ls:/bin/cat ; /bin/cat foo
```
- `/bin/cat: foo: No such file or directory`
  - `bash: cat: command not found`
  - `bash: ls: command not found`
  - no output on screen
  - `bash: /bin/cat: command not found`
142. What is the link count of directory `z` after these successful commands?
- ```
mkdir z ; mkdir z/a ; touch z/b z/c z/d
```
- 3
 - 1
 - 5
 - 2
 - 4

143. What is in the local variable `$$` ?
- `$$` is not a valid variable name
 - the first argument of the previous command line
 - the process ID of the current shell
 - the command name of the previous command line
 - the cpu cost of the current session, in dollars
144. If `/bin/pig` is a program that outputs `xx` and `/usr/bin/pig` is a program that outputs `foo` what would be the output on your screen of this two command sequence: `PATH=/home:/bin:/dev:/usr/bin ; pig`
- `foo`
 - `xx` followed by `foo`
 - `xx`
 - `foo` followed by `xx`
 - `bash: pig: command not found`
145. What is in file `c` after this command line:
- ```
echo A >a ; ln a b ; echo B >b ; ln a c ; rm a b
```
- `A`
  - nothing (empty file)
  - no such file (nonexistent)
  - `B`
  - `A` followed by `B`
146. If the file `pig` contained the word `bar`, what would be the output on your screen of this two command sequence:
- ```
PATH=/etc/passwd:/bin/ls:/bin/who ; /bin/cat pig
```
- `/bin/cat: pig: No such file or directory`
 - `bash: /bin/cat: command not found`
 - `bar`
 - `pig`
 - no output on screen
147. A "dangling symlink" is a symlink:
- to the current directory
 - to a parent directory
 - to a special device file
 - to a directory
 - to a non-existent target
148. If you want a user-defined alias in all your `bash` shells, what do you do?
- put the alias into the `grub.conf` file for next log in
 - create the alias and then type "save" to save it to all shells
 - put the alias into the `/etc/group` file for next log in
 - put the alias into the `/etc/passwd` file for next log in
 - define the alias in my file `$HOME/.bashrc`
149. 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:
- `/etc/foo/bar`
 - `/etc/foo`
 - `/etc/bar/foo`
 - `/bar/foo`
 - `/foo`

150. 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. 1 b. 4 c. 0 d. 3 e. 2
151. How many arguments are passed to the command by the shell:
`echo " one '2 three' 4 "five 6 ' 7 "8 ' >out`
 a. 5 b. 6 c. 4 d. 3 e. 2
152. How many arguments are passed to the command by the shell:
`<bar bar -b"-a '-r' >bar" bar >out`
 a. 4 b. 3 c. 2 d. 6 e. 5
153. What is the output on your screen after this command line:
`mkdir foo ; rmdir foo | wc -c`
 a. 3 b. 4 c. no output
 d. 0 e. 1
154. What command will recursively show disk usage in directories?
 a. `du` b. `ls` c. `find` d. `tree` e. `df`
155. How many arguments are passed to the command by the shell:
`echo 'And it's not hard, it's just logical.'`
 a. 5 b. 6 c. 7 d. 3 e. 4
156. What is the link count of file **foo** after these successful commands?
`rm foo ; touch foo ; ln foo bar`
`cp bar x ; ln x y ; ln y z`
 a. 3 b. 2 c. 0 d. 4 e. 1
157. If your **PATH** variable contains `/bin:/usr/bin`, what is the output of this command line: `echo '$PATH'`
 a. `/bin:/usr/bin`
 b. `'$PATH'`
 c. `'/bin:/usr/bin'`
 d. `echo: $PATH: No such file or directory`
 e. `$PATH`
158. 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. 4 b. 5 c. 3 d. 2 e. 6
159. If `/bin/foo` is a program that outputs **mom** and `/usr/bin/foo` is a program that outputs **dad**, what would be the output on your screen of this two command sequence: `PATH=/dev:/usr/bin:/usr:/bin:/etc ; /bin/foo`
 a. **dad**
 b. **mom** followed by **dad**
 c. **dad** followed by **mom**
 d. **mom**
 e. `bash: /bin/foo: command not found`

160. To change to the parent directory, do this:
 a. `cd ..` b. `pwd` c. `cd .`
 d. `pwd ..` e. `cd`
161. How many arguments are passed to the command by the shell:
`<cow cow "-x "-y '-z' >cow cow`
 a. 7 b. 6 c. 3 d. 5 e. 4
162. Which command shows the name of the current computer:
 a. `find` b. `history` c. `hostname`
 d. `whoami` e. `comname`
163. Which one of these names is usually a shell environment variable?
 a. `FooBar` b. `foobar` c. `fooBar`
 d. `fooBAR` e. `FOOBAR`
164. What is the output on your screen after this two command sequence:
`PATH=/bin/cat:/bin/sh:/bin/ls ; ls nosuchfile`
 a. `bash: ls: command not found`
 b. `ls: /bin/ls: command not found`
 c. `ls: nosuchfile: No such file or directory`
 d. `bash: /bin/ls: command not found`
 e. `bash: /bin/sh: No such file or directory`
165. Rewrite as a simplified absolute path:
`/../../../../var/./a/../../../../var/b/../../../../etc/./bar/./foo`
 a. `/etc/bar/foo` b. `/var/foo` c. `/var/b/foo`
 d. `/etc/foo` e. `/var/a/foo`
166. What is the output of this command line if run in an empty directory:
`touch A a ; echo * ">*"`
 a. `* >*` b. `A a >A a` c. `A a >*`
 d. No output e. `A a`
167. Which command line outputs inode/filename pairs for names in the current directory, sorted by inode number?
 a. `ls -node * > sort -n` b. `ls ./ * | sort -node`
 c. `ls -i * > sort -n` d. `sort -n | ls -ai`
 e. `ls -ai | sort -n`
168. What is the link count of file **f** after these successful commands?
`rm f ; touch f ; ln f b ; cp f g`
`cp b a ; ln a d ; ln b c ; cp c g`
 a. 2 b. 1 c. 4 d. 3 e. 5
169. 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`

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

```
echo 'It's a bird! It's a plane!'
```

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

171. If I have a directory named **a/b**, which action would increase its *link count* by exactly one?

- a. create a file named **a/b/c**
b. create a directory named **a/b/c**
c. create a hard link to directory **b** named **b2**
d. create a directory named **a/b2**
e. create a file named **a/b2**

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

```
<cow cow "-x" -y '-z' >cow cow
```

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

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

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

This page intentionally left blank.