Evaluation: 158 Questions **Name: __________________________**

Important Instructions

1. Read all the instructions and both sides of all pages.
2. Manage your time when answering questions on this test.
   Answer the questions you know, first.

Multiple Choice - 158 Questions

This is a practice test containing many practice questions. The real test will contain many questions similar to these. Knowing the concepts behind the questions is necessary; memorizing these answers won't help.

The real test will have approximately one question per minute. The real test may have some questions unlike anything given here.

(Office use only: 62 11 8 69 30 98 132 142 6 12 91 57 49 48 38 26 16 145 64 66 105 81 97 31 106 76 85 56 35 146 112 21 158 39 109 93 149 114 131 87 129 115 134 96 4 157 117 15 67 75 90 59 79 23 28 123 102 32 65 143 52 61 148 84 82 104 54 3 88 10 71 29 141 125 74 89 135 2 42 100 116 53 111 108 25 18 154 150 63 95 80 51 14 68 1 46 130 24 22 13 156 126 55 155 86 103 124 151 44 147 40 138 77 120 140 47 34 33 45 20 72 133 144 9 128 99 122 92 136 101 127 152 119 19 17 58 110 153 5 121 113 37 27 73 60 83 78 50 139 70 107 43 36 41 118 94 137 7)

1. In an empty directory, what is the length of the longest file name created by the following `bash` shell two-command sequence:
   `ok='1 12 123 1234' ; touch '$ok'`
   a. 3 characters
   b. 2 characters
   c. 13 characters
   d. 4 characters
   e. 1 character

2. If `/etc/passwd` is a file name, which of the following pathnames always leads to the same file?
   a. `/../etc/passwd`
   b. `/etc/passwd/../..`
   c. `./etc/passwd`
   d. `/etc/../../passwd`
   e. `/etc/passwd/.`

3. If `x=5` and `y=5`, which `bash` command sequence correctly compares the two numbers as equal and prints `OK`?
   a. `if ( x == y ) ; then echo OK ; fi`
   b. `if [ x = y ] ; then echo OK ; fi`
   c. `if test $x -eq $y ; then echo OK ; fi`
   d. `if test x -eq y ; then echo OK ; fi`
   e. `if [ $x==$y ] ; then echo OK ; fi`

4. What is the output of the following sequence of `bash` commands:
   `x=0 ; y=1 ; touch $x ; test ! -z $x ; echo $?`
   a. the number 1 or 0 followed by another 1 or 0 on a new line
   b. `test: $x: integer expression expected`
   c. `1`
   d. no output
   e. `0`

5. If `x=cow` and `y=dog` then what is the output of the following sequence of `bash` commands: `[ $x = dog -o $y = cow ] ; echo $?`
   a. `1`
   b. `0`
   c. no output
   d. `test: $x: integer expression expected`
   e. the number 0 or 1 followed by another 0 or 1 on a new line

6. What is the output of this sequence of three shell commands:
   `umask 547 ; mkdir newdir ; ls -ld newdir`
   a. `dr--r--rw- 1 me me 0 Feb 20 07:55 newdir`
   b. `dr-xr--rwx 1 me me 0 Feb 20 07:55 newdir`
   c. `d-w--wx--- 1 me me 0 Feb 20 07:55 newdir`
   d. `d-w--w---- 1 me me 0 Feb 20 07:55 newdir`
   e. `d-w--wxrwx 1 me me 0 Feb 20 07:55 newdir`

7. What is the `bash` output of this command sequence:
   `false && echo "Hello     There"`
   a. `HelloThere`
   b. `Hello     There`
   c. no output
   d. `Hello There`
   e. `"Hello     There"`

8. If file `/a` contains 20 lines, and file `/b` contains 30 lines, then how many lines are in file `/c` after this sequence of shell commands:
   `sort /a /b >/c ; cat /a >>/b ; sort /c /b /a >/c`
   a. `50`
   b. `80`
   c. `120`
   d. `70`
   e. no lines (empty file)

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

10. If file **foo** contains nine lines, each of which is the one-digit line number of the line in the file (**1** through **9**), what is the output of this command:
    cat foo foo | cat | tail -4 | head -1
    a. **7**
    b. **8**
    c. **9**
    d. **6**
    e. **5**

11. A shell script named **foo** is executed as follows:
    ./foo a "b c d" e
    Inside the script is the line: **echo "$2"**
    What is the output from this line?
    a. **b**
    b. **"b**
    c. **b c d**
    d. a bash error message: unbound (undefined) variable
    e. **$2**

12. What is the **bash** shell output of this two-command sequence:
    cd /home/alleni && echo "In $(pwd)"
    a. **In 0pwd)**
    b. **In $(pwd)**
    c. **In /home/alleni**
    d. **"In $(pwd)"**
    e. no output

13. Which of the following shell command lines displays all the names in the current directory that are exactly three letters (alphabetic) long (and nothing else)?
    a. **echo ???**
    b. **echo [a-zA-Za-zA-Za-zA-Z]**
    c. **echo [azAZ][azAZ][azAZ]**
    d. **echo [a-zA-Z][a-zA-Z][a-zA-Z]**
    e. **echo [a,zA,Z][a,zA,Z][a,zA,Z]**

14. Which of the following shell command lines displays all the names in the current directory that are exactly three letters (alphabetic) long (and nothing else)?
    a. **echo [azAZ][azAZ][azAZ]**
    b. **echo [0-89][01-9][0-45-9]**
    c. **echo ???**
    d. **echo [a-mn-zA-YZ][ab-zAB-YZ][za-yZA-Y]**
    e. **echo [a,zA,Z][a,zA,Z][a,zA,Z]**

15. Which of the command lines below can generate a non-empty file?
    a. **tail -5 /a/b >/a/b**
    b. **tr abc ABC </a/b >/a/b**
    c. **ls /a/b >/a/b**
    d. **sort -r /a/b >/a/b**
    e. **grep -v /a/b /a/b >/a/b**

16. If my current working directory is **/home**, and my home directory is **/home/me**, which of the of the following commands copies the Unix password file into my home directory under the name **foo**?
    a. **cp me/../../etc/passwd me/foo**
    b. **cp ../etc/passwd ../me/foo**
    c. **cp ./me/../etc/passwd ../home/me/foo**
    d. **cp ../../etc/passwd /me/foo**
    e. **cp ../home/me/../etc/passwd ./me/./foo**

17. Which command sequence below outputs only lines 11-15 of the Unix password file?
    a. **tail -10 /etc/passwd | head -15 /etc/passwd**
    b. **head -15 /etc/passwd | tail -5**
    c. **head -15 /etc/passwd | tail -5 /etc/passwd**
    d. **tail -15 /etc/passwd | head -10**
    e. **head -10 /etc/passwd | tail -15 /etc/passwd**

18. If directory **dir** contains only these five two-character file names: **a?**, **11**, **?1**, **1***, **.1**, then which shell command below will remove *only* the single two-character name **?1** from the directory?
    a. **rm dir/1***
    b. **rm dir/??**
    c. **rm dir/*1**
    d. **rm dir/?1**
    e. **rm dir/\??**

19. Given my directory **dir** and my file **dir/foo** owned by me, which permissions allow me to delete the file **dir/foo** from the directory, but not change the content (data) in the file?
   a. Permissions **100** on directory **dir** and **100** on file **dir/foo**.
   b. Permissions **100** on directory **dir** and **200** on file **dir/foo**.
   c. Permissions **300** on directory **dir** and **500** on file **dir/foo**.
   d. Permissions **300** on directory **dir** and **300** on file **dir/foo**.
   e. Permissions **500** on directory **dir** and **400** on file **dir/foo**.

20. What is the output of the following sequence of **bash** commands:
   a=cow ; b=dog ; touch $a ; test -z $a ; echo $?
   a. the number 1 or 0 followed by another 1 or 0 on a new line
   b. **test: $a: integer expression expected**
   c. **1**
   d. no output
   e. **0**

21. If **/bin/pig** is a program that outputs **hi** and **/usr/bin/pig** is a program that outputs **foo** what is the output of this shell command sequence: **PATH=/etc:/usr/bin:/bin ; pig**
   a. **foo**
   b. **hi**
   c. **bash: pig: command not found**
   d. **foo** followed by **hi**
   e. **hi** followed by **mom**

22. What is the **bash** shell output of this two-command sequence if run in a directory containing 123 files with names that are all the numbers from **1** to **123** inclusive: **bat="*" ; echo "$bat"**
   a. **$bat**
   b. the file names **1** through **123**
   c. the file names **1** through **123**, surrounded by quotes
   d. **"$bat"**
   e. *****

23. If the file **pig** contained the word **foo**, what would be the **bash** shell output of this two command sequence:
   PATH=/etc/passwd:/bin/ls:/bin/cat ; /bin/ls pig
   a. **/bin/ls: pig: No such file or directory**
   b. **pig**
   c. **foo**
   d. **bash: /bin/ls: command not found**
   e. no output

24. If **x=cow** and **y=dog** then what is the output of the following sequence of **bash** commands: **[$x = cow -a $y = cow] ; echo $?**
   a. no output
   b. **test: $x: integer expression expected**
   c. **0**
   d. the number 0 or 1 followed by another 0 or 1 on a new line
   e. **1**

25. If a **bash** shell script named **foo** contains the line:
   if ['$1' = "$2"] ; then echo SAME ; fi
   then which of the following command lines will produce **SAME** as output?
   a. **./foo $1 $1**
   b. **./foo bar 'bar'**
   c. **./foo 'bar' "bar"**
   d. **./foo 1 "$1"**
   e. **./foo bar '$1'**

26. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to change or create new content (data) in the file **dir/bar** but not delete the file?
   a. Permissions **400** on directory **dir** and **400** on file **dir/bar**.
   b. Permissions **100** on directory **dir** and **100** on file **dir/bar**.
   c. Permissions **500** on directory **dir** and **600** on file **dir/bar**.
   d. Permissions **200** on directory **dir** and **200** on file **dir/bar**.
   e. Permissions **600** on directory **dir** and **700** on file **dir/bar**.

27. If **bar** is an executable script containing the line **foo=dog** then what is the **bash** output of this sequence of three commands:
   foo=cat ; ./bar ; echo "the '$foo' ate"
   a. **the $foo ate**
   b. **the 'foo' ate**
   c. **the 'cat' ate**
   d. **the '$foo' ate**
   e. **the 'dog' ate**

28. Which of these statements is true?
   a. If **/q** is an empty directory, **echo /q/.*** produces an error message.
   b. If **/p** is an empty directory, **ls /p/.*** produces an error message.
   c. The **ls dir** command looks up the directory argument **dir** in your **$PATH**.
   d. Either single or double quotes will stop shell glob (wildcard) patterns from expanding.
   e. Typing **./script** and **bash script** always give identical results.

29. If **a=cow** and **b=dog** then what is the output of the following sequence of **bash** commands: **if $a = $b ; then echo $a ; fi**
   a. no output
   b. **bash: cow: command not found**
   c. **cow**
   d. **test: $a: integer expression expected**
   e. **test: cow: integer expression expected**

30. Which of the following **bash PATH** statements makes the most sense?
   a. **PATH=/bin:/usr/bin:/etc/passwd**
   b. **PATH=/bin:/bin/cat:/usr/bin**
   c. **PATH=/bin/ls:/etc:/usr/bin**
   d. **PATH=/bin/sh:/usr/bin:/etc:/bin**
   e. **PATH=/bin:/usr/bin:/etc**

31. Which of the following shell command lines displays only the names in the current directory that are exactly three numeric digits long?
   a. **echo [1−3][1−3][1−3]**
   b. **echo [0−9][0−9][0−9]**
   c. **echo ???**
   d. **echo ’0−9’0−9’0−9’**
   e. **echo ’[0−9]’’[0−9]’’[0−9]’**

32. In an empty directory, what is the shell output of these three commands:
   touch xx .x xy .y xz ; x=’x* y*’ ; echo "$x"
   a. **xx xy xz y***
   b. **x* y***
   c. ***x *y**
   d. **$x**
   e. **xx xy**

33. Select the correct **bash** shell order of command line processing:
   a. aliases, variables, globs, redirection
   b. aliases, globs, variables, redirection
   c. aliases, variables, redirection, globs
   d. redirection, aliases, globs, variables
   e. aliases, redirection, variables, globs

34. Which of the command lines below can generate a non-empty file?
   a. **sort −r /out >/out**
   b. **ls /out >/out**
   c. **tr abc ABC </out >/out**
   d. **grep −v /out /out >/out**
   e. **tail −5 /out >/out**

35. What is the link count of directory **dir** after this set of successful commands?
   mkdir dir ; cd dir ; touch foo ; mkdir a b c
   a. 2
   b. 5
   c. 3
   d. 4
   e. 1

36. How many arguments and options are there to the command:
   sort −r <infile
   a. Two arguments, neither of which is an option.
   b. A file name starting with a dash and an **<infile** switch option argument.
   c. Two arguments, one of which is a single option name and the other is a pathname.
   d. One command line argument containing one option name.
   e. Three arguments, one of which contains an option and one is a pathname.

37. Which command sequence below does *not* generate an error message from the last command in the sequence?
   a. **mkdir one one/two ; rmdir one/two**
   b. **date >foo ; cp foo/. bar**
   c. **mkdir foo foo/bar ; rmdir foo**
   d. **cat /etc/passwd > mail idallen@ncf.ca**
   e. **mkdir foo ; ln foo bar**

38. If **a=xxx** and **b=yyy** then what is the output of the following sequence of **bash** commands: **if $a = $b ; then echo $a ; fi**
   a. **xxx**
   b. no output
   c. **test: xxx: integer expression expected**
   d. **bash: xxx: command not found**
   e. **test: $a: integer expression expected**

39. What is the **bash** shell output of this command sequence:
   true && echo Hello   There $?
   a. **Hello There ?**
   b. **Hello   There ?**
   c. **Hello There 0**
   d. no output
   e. **Hello   There 1**

40. What is the output of the following sequence of **bash** commands:
    cd /etc && echo "in $(pwd)"
    a. **bash: cd: /etc: No such file or directory**
    b. **in 0pwd)**
    c. **in $(pwd)**
    d. **in /etc**
    e. no output

41. What is the output of the following sequence of **bash** commands:
    false && echo "linux   rocks $?"
    a. no output
    b. **linux rocks 0**
    c. **linux rocks 1**
    d. **linux   rocks 0**
    e. **linux   rocks 1**

42. If **foo** is a script containing the line **TERM=linux ; export TERM**, what is the output of the following sequence of **bash** commands:
    TERM=vt100 ; ./foo ; echo "$TERM"
    a. **vt100**
    b. **linux**
    c. **$TERM**
    d. **foo**
    e. **TERM**

43. How many arguments are passed to the command by the shell on this command line: **<bar bar -b "-a" '-r' >bar bar bar**
    a. **6**
    b. **2**
    c. **3**
    d. **5**
    e. **4**

44. If the file **foo** contained the word **mom**, what would be the **bash** shell output of this two command sequence:
    PATH=/etc/passwd:/bin/ls:/bin/cat ; /bin/ls foo
    a. **bash: /bin/ls: command not found**
    b. **/bin/ls: foo: No such file or directory**
    c. **foo**
    d. **mom**
    e. no output

45. Which **bash** command sequence correctly searches for the string **foo** and then prints **YES** if it is found inside the group file?
    a. **if test foo /etc/group ; then echo YES ; fi**
    b. **if grep foo /etc/group ; then echo YES ; fi**
    c. **if test foo = /etc/group ; then echo YES ; fi**
    d. **if [test foo /etc/group] ; then echo YES ; fi**
    e. **if [grep foo /etc/group] ; then echo YES ; fi**

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

47. If **foo** is a script containing the line **TERM=new ; export TERM**, what is the output of the following sequence of **bash** commands that use **foo**:
    TERM=bar ; ./foo ; echo $TERM
    a. **new**
    b. **bar**
    c. **TERM**
    d. **foo**
    e. **$TERM**

48. If file **foo** contains nine lines, each of which is the one-digit line number of the line in the file (**1** through **9**), what is the output of this command:
    cat foo foo | sort -r | head -4 | tail -1
    a. **6**
    b. **5**
    c. **9**
    d. **8**
    e. **7**

49. What is the output of the following sequence of **bash** commands:
    x=cow ; y=dog ; touch $x ; test -z $x ; echo $?
    a. **test: $x: integer expression expected**
    b. the number 0 or 1 followed by another 0 or 1 on a new line
    c. **0**
    d. **1**
    e. no output

50. Given my directory **dir** and my file **dir/foo** owned by me, which permissions allow me to change or create new content (data) in the file **dir/foo** but not delete the file?
   a. Permissions **100** on directory **dir** and **100** on file **dir/foo**.
   b. Permissions **400** on directory **dir** and **400** on file **dir/foo**.
   c. Permissions **200** on directory **dir** and **200** on file **dir/foo**.
   d. Permissions **500** on directory **dir** and **600** on file **dir/foo**.
   e. Permissions **600** on directory **dir** and **700** on file **dir/foo**.

51. A shell script named **foo** is executed as follows:
   ./foo 1 2 "3 4" 5
   Inside the script is the line: **echo "$3"**
   What is the output from this line?
   a. **3 4**
   b. **"3**
   c. **1 2 3**
   d. **2 3 4**
   e. **$3**

52. What is the **bash** shell output of this two-command sequence:
   cd /etc/passwd && echo "in $(pwd)"
   a. **in 0pwd)**
   b. **bash: cd: /etc/passwd: Not a directory**
   c. **in $(pwd)**
   d. no output
   e. **in /etc**

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

54. If variable **cow** might contain nothing (a null value - defined but empty), which **bash** command sequence correctly tests for this and prints **OK**?
   a. **if [$cow -eq :] ; then echo OK ; fi**
   b. **if ["" = "$cow"] ; then echo OK ; fi**
   c. **if [''$cow'' = ''''] ; then echo OK ; fi**
   d. **if ["$cow" = *] ; then echo OK ; fi**
   e. **if [$cow -eq ""] ; then echo OK ; fi**

55. If **a=cow** and **b=dog** then what is the output of the following sequence of **bash** commands: **[$a = dog -o $b = cow] ; echo $?**
   a. the number 1 or 0 followed by another 1 or 0 on a new line
   b. **0**
   c. **test: $a: integer expression expected**
   d. **1**
   e. no output

56. Given the following **bash** shell command line:
   read hi my ok
   which user keyboard input line below will assign the text **two** to the shell variable named **my**?
   a. **<one <two <three**
   b. **one two three**
   c. **"one" "two" "three"**
   d. **one,two,three**
   e. **hi=one my=two ok=three**

57. What is the **bash** shell output of this two command sequence:
   PATH=/bin/ls:/bin/cat:/bin/sh ; cat nosuchfile
   a. **ls: /bin/cat: command not found**
   b. **cat: nosuchfile: No such file or directory**
   c. **bash: /bin/sh: No such file or directory**
   d. **bash: /bin/ls: command not found**
   e. **bash: cat: command not found**

58. If **a=cow** and **b=pig** then what is the output of the following sequence of **bash** commands: **[$a = pig -o $b = pig] ; echo $?**
   a. the number 1 or 0 followed by another 1 or 0 on a new line
   b. **test: $a: integer expression expected**
   c. **0**
   d. **1**
   e. no output

59. Given my directory **dir** and my file **dir/bar** owned by me, which permissions allow me to delete the file **dir/bar** from the directory, but not change the content (data) in the file?
   a. Permissions **500** on directory **dir** and **400** on file **dir/bar**.
   b. Permissions **100** on directory **dir** and **100** on file **dir/bar**.
   c. Permissions **300** on directory **dir** and **500** on file **dir/bar**.
   d. Permissions **300** on directory **dir** and **300** on file **dir/bar**.
   e. Permissions **100** on directory **dir** and **200** on file **dir/bar**.

60. If file **/a** contains 40 lines, and file **/b** contains 60 lines, then how many lines are output by this command:
sort /a /b | cat /a | cat /b
a. **200**
b. **40**
c. **60**
d. **100**
e. **160**

61. What is the link count of file **foo** after this set of successful commands?
rm foo ; touch foo ; ln foo bar
cp bar x ; ln x y ; ln bar z ; ln z a
a. 5
b. 3
c. 1
d. 2
e. 4

62. In an empty directory, what is the length of the longest file name created by the following **bash** shell two-command sequence:
x='1 12 123 1234' ; touch '$x'
a. 13 characters
b. 2 characters
c. 4 characters
d. 3 characters
e. 1 character

63. Which of these first lines will cause this executable file to be interpreted using the Bash shell?
a. **#/bin/bash**
b. **/bin/bash -u**
c. **#!/bin/bash**
d. **!#/bin/bash -u**
e. **!/bin/bash**

64. What is the **bash** shell output of this two-command sequence if run in a directory containing 765 files with names that are all the numbers from **1** to **765** inclusive: **foo="*" ; echo $foo**
a. **$foo**
b. *****
c. an asterisk ('*****') and the file names **1** through **765**
d. all the file names that start with an asterisk ('*****')
e. the file names **1** through **765**

65. What is the **bash** shell output of this two-command sequence if run in a directory containing 123 files with names that are all the numbers from **1** to **123** inclusive: **glob="*" ; echo "$glob"**
a. the file names **1** through **123**
b. the file names **1** through **123**, surrounded by quotes
c. *****
d. **$glob**
e. **"$glob"**

66. If **/bin/foo** is a program that outputs **hi** and **/usr/bin/foo** is a program that outputs **mom** what is the output of this shell command sequence: **PATH=/etc:/usr/bin:/bin ; foo**
a. **bash: foo: command not found**
b. **mom** followed by **hi**
c. **mom**
d. **hi** followed by **mom**
e. **hi**

67. What is the output of this command sequence:
echo pig >one ; echo cow | head -2 one
a. **cow** followed by **pig**
b. **pig**
c. an error message
d. **pig** followed by **cow**
e. **cow**

68. Which **bash** command sequence correctly searches for the **chars** and then prints **OK** if it is found inside the password file?
a. **grep chars </etc/passwd || echo OK**
b. **grep chars </etc/passwd && echo OK**
c. **grep chars /etc/passwd || echo OK**
d. **grep chars >/etc/passwd || echo OK**
e. **grep chars >/etc/passwd && echo OK**

69. What is the output of the following sequence of **bash** commands:
cd /bin && echo "cd $(pwd)"
a. **cd $(pwd)**
b. **cd /bin**
c. **bash: cd: /bin: No such file or directory**
d. **cd 0pwd)**
e. no output

70. If `/etc/passwd` is a file name, which of the following pathnames always leads to the same file?
    a. `/./../etc/./passwd`
    b. `/etc/../../passwd`
    c. `././etc/passwd`
    d. `/etc/passwd/./.`
    e. `/etc/passwd/../..`

71. What is the output of the following sequence of `bash` commands:
    `x=1 ; touch x ; test ! -z $x ; echo $?`
    a. `0`
    b. the number 1 or 0 followed by another 1 or 0 on a new line
    c. `1`
    d. `test: $x: integer expression expected`
    e. no output

72. If `a=cow` and `b=dog` then what is the output of the following sequence of `bash` commands: `[ $a = cow -a $b = cow ] ; echo $?`
    a. `0`
    b. `1`
    c. `test: $a: integer expression expected`
    d. no output
    e. the number 1 or 0 followed by another 1 or 0 on a new line

73. What is the output of the following sequence of `bash` commands:
    `date='October Monday' ; test date = date`
    a. `0`
    b. no output
    c. `test: too many arguments`
    d. `1`
    e. `Mon Oct 27 17:01:38 EST 2003`

74. Which `bash` command sequence below always outputs just the date only if the first argument is either a file or a directory?
    a. `if [ -f "$1" -o -d "$1" ]; then date ; fi`
    b. `if [ "-f $1" || "-d $1" ]; then date ; fi`
    c. `if [ -f || -d "$1" ]; then date ; fi`
    d. `if [ -f -o -d "$1" ]; then date ; fi`
    e. `if [ "$1" -eq -f -o "$1" -eq -d ]; then date ; fi`

75. What is the output of the following sequence of `bash` commands:
    `x=0 ; test $x ; echo $?`
    a. the number 0 or 1 followed by another 0 or 1 on a new line
    b. `test: $x: integer expression expected`
    c. `1`
    d. `0`
    e. no output

76. A shell script named `foo` is executed as follows:
    `./foo 1 "2 3 4" 5`
    Inside the script is the line: `echo "$2"`
    What is the output from this line?
    a. `2 3 4`
    b. `"2`
    c. `2`
    d. `"2 3 4"`
    e. `$2`

77. How many arguments are passed to the command by the shell on this command line: `<pig pig -b "-a -r" >pig pig pig`
    a. `2`
    b. `5`
    c. `3`
    d. `6`
    e. `4`

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

79. If `a=1` and `b=2` then which of the following `bash` command lines outputs only the word `hi` (and nothing else)?
    a. `[ a -ne b ] && echo hi`
    b. `[a -ne b] || echo hi`
    c. `[!a = b] && echo hi`
    d. `[ a = a ] && echo hi`
    e. `[a!=a] || echo hi`

80. What is the output of this sequence of three shell commands:
    `umask 674 ; touch newfile ; ls -l newfile`
    a. `-rw-rwxr-- 1 me me 0 Feb 20 07:55 newfile`
    b. `--w--wxr-x 1 me me 0 Feb 20 07:55 newfile`
    c. `--------w- 1 me me 0 Feb 20 07:55 newfile`
    d. `-rw-rw-r-- 1 me me 0 Feb 20 07:55 newfile`
    e. `---x----wx 1 me me 0 Feb 20 07:55 newfile`

81. If **foo** is a script containing the line **TERM=vt100 ; export TERM**, what is the output of the following sequence of **bash** commands:
    TERM=linux ; ./foo ; echo $TERM
    a. **$TERM**
    b. **foo**
    c. **vt100**
    d. **TERM**
    e. **linux**

82. What is the output of this command sequence:
    echo dog >one ; echo cow | head -2 one
    a. **cow**
    b. **cow** followed by **dog**
    c. **dog**
    d. an error message
    e. **dog** followed by **cow**

83. Which of the following statements is true about this shell command line:
    >bar zoom bar haven
    a. The command **bar** sees only two arguments
    b. The command **bar** sees three arguments.
    c. The command **zoom** sees two arguments.
    d. The command **zoom** sees three arguments.
    e. Error: The command name is missing from the command line.

84. What is the link count of directory **dir** after this set of successful commands? **mkdir dir ; cd dir ; touch one ; mkdir two**
    a. 5
    b. 4
    c. 3
    d. 1
    e. 2

85. If variable **bar** might contain nothing (a null value - defined but empty), which **bash** command sequence correctly tests for this and prints **YO**?
    a. **if ["$bar" = ""] ; then echo YO ; fi**
    b. **if [''$bar'' = ''''] ; then echo YO ; fi**
    c. **if [$bar -eq :] ; then echo YO ; fi**
    d. **if ["$bar" = *] ; then echo YO ; fi**
    e. **if [$bar -eq ""] ; then echo YO ; fi**

86. In an empty directory, what is the shell output of these three commands:
    touch .1 .2 .3 11 12 ; b='.1* .2*' ; echo '$b'
    a. **'.1* .2*'**
    b. **.1 .2**
    c. **.1* .2***
    d. **$b**
    e. **11 .1 12 .2**

87. Which of the following VI/VIM key sequences will move the entire line on which the cursor resides to after the line that follows it (i.e. it would move line 5 to be line 6 and line 6 would become line 5)?
    a. **ddP**
    b. **DDP**
    c. **ddp**
    d. **DDp**
    e. **:dp**

88. Which line below is most likely to be the beginning of an error message?
    a. **echo 2>&1 "... "**
    b. **echo 2>$1 "... "**
    c. **echo 2<$1 "... "**
    d. **echo 1>&2 "... "**
    e. **echo 1<&2 "... "**

89. In an empty directory, what is the length of the longest file name created by the following **bash** shell two-command sequence:
    var='1 12 123 1234 12345' ; touch '$var'
    a. 3 characters
    b. 1 character
    c. 13 characters
    d. 4 characters
    e. 2 characters

90. How many arguments and options are there to the command:
    cal -yj 1752
    a. Two command line arguments, one of which contains two options.
    b. Two arguments, one of which is a single option and the other is a pathname.
    c. Three arguments, one of which contains options and one is a pathname.
    d. Two arguments, neither of which is an option.
    e. A single numeric option and a three-letter file name.

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

92. What is the link count of file **foo** after this set of successful commands?
   rm foo ; touch foo ; ln foo bar
   cp bar x ; ln x y ; ln bar z
   a. 4
   b. 5
   c. 3
   d. 2
   e. 1

93. If file **foo** contains nine lines, each of which is the one-digit line number of the line in the file (**1** through **9**), what is the output of this command:
   cat foo foo | sort -r | head -5 | tail -1
   a. **5**
   b. **9**
   c. **6**
   d. **8**
   e. **7**

94. What is the output of the following sequence of **bash** commands:
   a=cow ; touch $a ; test -z $a ; echo $?
   a. no output
   b. **0**
   c. the number 1 or 0 followed by another 1 or 0 on a new line
   d. **test: $a: integer expression expected**
   e. **1**

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

96. What is the output of the following sequence of **bash** commands:
   echo wc >wc ; wc wc >wc ; head wc
   a. **0 0 0 wc**
   b. **1 1 3 wc**
   c. **wc**
   d. **1 1 2 wc**
   e. no output

97. What is the **bash** shell output of this command sequence:
   true && echo space   junk $?
   a. **space   junk ?**
   b. no output
   c. **space   junk 1**
   d. **space junk ?**
   e. **space junk 0**

98. If variable **mt** might contain nothing (a null value - defined but empty), which **bash** command sequence correctly tests for this and prints **OK**?
   a. **if [$mt -eq ""] ; then echo OK ; fi**
   b. **if ["$mt" = *] ; then echo OK ; fi**
   c. **if [$mt -eq :] ; then echo OK ; fi**
   d. **if ["$mt" = ""] ; then echo OK ; fi**
   e. **if [''$mt'' = ''''] ; then echo OK ; fi**

99. If variable **foo** might contain nothing (a null value - defined but empty), which **bash** command sequence correctly tests for this and prints **OK**?
   a. **if [$foo -eq ""] ; then echo OK ; fi**
   b. **if [''$foo'' = ''''] ; then echo OK ; fi**
   c. **if ["$foo" = *] ; then echo OK ; fi**
   d. **if [$foo -eq :] ; then echo OK ; fi**
   e. **if ["$foo" = ""] ; then echo OK ; fi**

100. If file **foo** contains nine lines, each of which is the one-digit line number of the line in the file (**1** through **9**), what is the output of this command:
   cat foo foo | cat | tail -5 | head -1
   a. **9**
   b. **6**
   c. **5**
   d. **7**
   e. **8**

101. What is the output of this sequence of three shell commands:
   echo x >abc ; ls >abc abc ; wc abc
   a. **1 1 4 abc**
   b. **0 0 0 abc**
   c. no output
   d. **1 1 2 abc**
   e. **1 1 3 abc**

102. Which **bash** command sequence below always outputs just the date only if the first argument is both not empty and a directory?
   a. **if [-s -a -d "$1"]; then date ; fi**
   b. **if ["-s $1" && "-d $1"]; then date ; fi**
   c. **if [-s && -d "$1"]; then date ; fi**
   d. **if ["$1" -eq -f -a "$1" -eq -d]; then date ; fi**
   e. **if [-d "$1" -a -s "$1"]; then date ; fi**

103. Which of these statements is true?
   a. Only single quotes are strong enough to stop shell glob (wildcard) patterns from expanding.
   b. If **/x** is an empty directory, **sort /x/*** produces an error message.
   c. Typing **./script** and **bash script** always give identical results.
   d. The **cat food** command looks up the file name argument **food** in your **$PATH**.
   e. If **/y** is an empty directory, **echo /y/*** produces an error message.

104. Which line below passes three *separate* arguments to the **sort** command when placed inside a shell script named **foo** invoked by the command line:
   ./foo 111 222 333
   a. **sort "$1 $2 $3"**
   b. **sort "$*"**
   c. **sort "$#"**
   d. **sort "$? $? $?"**
   e. **sort "$@"**

105. If **bar** is an executable script containing the line **cow=pig** then what is the **bash** output of this sequence of three commands:
   cow=cat ; ./bar ; echo "the '$cow' ate"
   a. **the 'cow' ate**
   b. **the 'cat' ate**
   c. **the $cow ate**
   d. **the '$cow' ate**
   e. **the 'pig' ate**

106. What is the **bash** shell output of this two command sequence:
   PATH=/bin/ls:/bin/head:/bin/sh ; head nosuchfile
   a. **bash: head: command not found**
   b. **ls: /bin/head: command not found**
   c. **bash: /bin/sh: No such file or directory**
   d. **head: nosuchfile: No such file or directory**
   e. **bash: /bin/ls: command not found**

107. Given the following **bash** shell command line: **read a b c**, which user keyboard input line below will assign the text **b** to the shell variable named **b**?
   a. **a:b:c**
   b. **a,b,c**
   c. **a b c**
   d. **a;b;c**
   e. **a=a b=b c=c**

108. Which line below puts the count of the number of lines in the password file into the variable **foo**?
   a. **foo=[cat -l /etc/passwd]**
   b. **foo=$(wc -l </etc/passwd)**
   c. **foo=[wc /etc/passwd | echo $1]**
   d. **foo=[grep -c /etc/passwd]**
   e. **foo=$(cat -c /etc/passwd)**

109. What is the output of the following sequence of **bash** commands:
   echo wc >wc ; wc wc >wc ; cat wc
   a. no output
   b. **0 0 0 wc**
   c. **1 1 3 wc**
   d. **wc**
   e. **1 1 2 wc**

110. Which of these commands makes a file owned by me, also executable by me?
   a. **chmod x=u ./myfile**
   b. **chmod x+u myfile**
   c. **chmod u+x ./myfile**
   d. **umask 111 myfile**
   e. **umask 777 myfile**

111. If **dog=12** and **cat=99** then which of the following **bash** command lines outputs only the word **hi** (and nothing else)?
   a. **[!dog = cat] && echo hi**
   b. **[dog!=dog] || echo hi**
   c. **[dog -ne cat] && echo hi**
   d. **[dog -ne cat] || echo hi**
   e. **[dog = dog] && echo hi**

112. Given this long listing:
   drwxr-xr-x 448 me me 296 Dec 4 9:12 /dir
   How many subdirectories lie immediately under **/dir**?
   a. 296
   b. 294
   c. 448
   d. 446
   e. there is not enough information shown to answer the question

113. What is in the file named **file** after this command sequence:
   echo a >x ; echo b >>x ; mv x y >file
   a. **b**
   b. **a**
   c. **a** followed by **b**
   d. no such file (nonexistent file)
   e. nothing - **file** is empty - no data

114. If **foo** were a file of text containing 50 different lines, what would be the output of this exact command line: **diff foo foo**
   a. an error message because **diff** only allows one file name
   b. an error message because **diff** doesn't allow the same file name twice
   c. no output
   d. the contents of file **foo** would be displayed
   e. several lines, which are the lines that are different between the two files

115. In an empty directory, what is the **bash** shell output of this three-command sequence:
   touch aa .a ab .b .c ; x='.a* .b*' ; echo '$x'
   a. **aa .a ab .b**
   b. **.a .b**
   c. **$x**
   d. **'.a* .b*'**
   e. **.a* .b***

116. What is the output of the following sequence of **bash** commands:
   echo hi >wc ; wc wc >hi ; cat hi
   a. **hi**
   b. **0 0 0 wc**
   c. **1 1 2 wc**
   d. **1 1 3 wc**
   e. no output

117. If **a=cow** and **b=dog** then what is the output of the following sequence of **bash** commands linked by **&&**:
   [$a = dog -o $b = dog] && echo $?
   a. **1**
   b. the number 1 or 0 followed by another 1 or 0 on a new line
   c. **test: $a: integer expression expected**
   d. no output
   e. **0**

118. If **a=pig** and **b=dog** then what is the output of the following sequence of **bash** commands: **[$a = pig -a $b = pig] ; echo $?**
   a. **test: $a: integer expression expected**
   b. **0**
   c. **1**
   d. the number 1 or 0 followed by another 1 or 0 on a new line
   e. no output

119. What is the output of the following sequence of **bash** commands:
   echo wc >wc ; wc wc >wc ; sort wc
   a. **wc**
   b. no output
   c. **1 1 3 wc**
   d. **0 0 0 wc**
   e. **1 1 2 wc**

120. In an empty directory, what is the shell output of these three commands:
   touch xx .x xy .y xz ; a='x* y*' ; echo "$a"
   a. **x* y***
   b. **xx xy xz y***
   c. **$a**
   d. ***x *y**
   e. **xx xy**

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

122. What is the **bash** shell output of this command sequence:
false && echo "linux   rocks $?"
a. no output
b. **linux   rocks 0**
c. **linux   rocks 1**
d. **linux rocks 1**
e. **linux rocks 0**

123. If directory **/dir** contains these three four-character file names: **.123**, **.124**, **.???**, then what is the output of the following **bash** shell command line: **echo /dir/????**
a. **echo: /dir/????: No such file or directory**
b. no output
c. **/dir/.123 /dir/.124**
d. **/dir/????**
e. **/dir/.123 /dir/.124 /dir/.???**

124. If **cow=5** and **dog=5**, which **bash** command sequence correctly compares the two numbers as equal and prints **OK**?
a. **if test cow -eq dog ; then echo OK ; fi**
b. **if test $cow -eq $dog ; then echo OK ; fi**
c. **if [cow = dog] ; then echo OK ; fi**
d. **if (cow == dog) ; then echo OK ; fi**
e. **if [$cow==$dog] ; then echo OK ; fi**

125. What is the output of the following sequence of **bash** commands:
false && echo "foo   bar $?"
a. **foo   bar 1**
b. **foo bar 0**
c. **foo   bar 0**
d. no output
e. **foo bar 1**

126. What is the output of this sequence of three shell commands:
umask 457 ; mkdir newdir ; ls -ld newdir
a. **d-wx-w---- 2 me me 512 Oct 1 1:12 newdir**
b. **dr-xr-xrwx 2 me me 512 Oct 1 1:12 newdir**
c. **d-wx-w-rwx 2 me me 512 Oct 1 1:12 newdir**
d. **dr--r-xrwx 2 me me 512 Oct 1 1:12 newdir**
e. **d-w--w---- 2 me me 512 Oct 1 1:12 newdir**

127. Given the following **bash** shell command line:
read xx yy zz
which user keyboard input line below will assign the text **22** to the shell variable named **yy**?
a. **11;22;33**
b. **xx=11 yy=22 zz=33**
c. **11,22,33**
d. **11:22:33**
e. **11 22 33**

128. How many arguments and options are there to the command:
wc -l <infile
a. Three arguments, one of which contains an option and one is a pathname.
b. A file name starting with a dash and an **<infile** switch option argument.
c. Two arguments, neither of which is an option.
d. Two arguments, one of which is a single option name and the other is a pathname.
e. One command line argument containing one option name.

129. How many arguments are passed to the command by the shell on this command line: **<cow cow "-x" -y '-z' >cow cow**
a. **2**
b. **4**
c. **5**
d. **6**
e. **3**

130. What is the output of this sequence of three shell commands:
umask 762 ; touch newfile ; ls -l newfile
a. **--------wx 1 me me 0 Oct 1 1:12 newfile**
b. **-rwxrw--w- 1 me me 0 Oct 1 1:12 newfile**
c. **------xr-x 1 me me 0 Oct 1 1:12 newfile**
d. **-------r-- 1 me me 0 Oct 1 1:12 newfile**
e. **-rw-rw--w- 1 me me 0 Oct 1 1:12 newfile**

131. Which **bash** command sequence below always outputs just the date only if the first argument is either readable or executable?
   a. **if ["$1" -eq -r -o "$1" -eq -x]; then date ; fi**
   b. **if [-r "$1" -o -x "$1"]; then date ; fi**
   c. **if [-r -o -x "$1"]; then date ; fi**
   d. **if ["-r $1" || "-x $1"]; then date ; fi**
   e. **if [-r || -x "$1"]; then date ; fi**

132. Which command line displays the contents of the Unix **passwd** file one page at a time?
   a. **/etc/passwd >less**
   b. **less | /etc/passwd**
   c. **less </etc/passwd**
   d. **/etc/passwd | less**
   e. **cat /etc/passwd >less**

133. How can you ask the **bash** shell to complete commands or file names for you?
   a. Type **[ALT]-[F2]** the shell will present a menu of commands.
   b. Type **[CONTROL]-[ALT]-[DEL]** and the shell will present a menu of commands.
   c. You can type the first part of the command or file name and press the **TAB** key.
   d. Type **[CONTROL]-[D]** and the shell will present a menu of commands.
   e. You can type the first part of the command or file name and press the **ALT** key.

134. In an empty directory, what is the shell output of these three commands:
   touch .1 .2 .3 11 12 ; a=’.1* .2*’ ; echo ’$a’
   a. **.1* .2***
   b. **’.1* .2*’**
   c. **.1 .2**
   d. **$a**
   e. **11 .1 12 .2**

135. Which command sequence below outputs only lines 10-15 of the Unix password file?
   a. **head -10 /etc/passwd | tail -5 /etc/passwd**
   b. **tail -15 /etc/passwd | head -5**
   c. **head -15 /etc/passwd | tail -5 /etc/passwd**
   d. **head -15 /etc/passwd | tail -6**
   e. **tail -10 /etc/passwd | head -15 /etc/passwd**

136. What is the **bash** shell output of this two-command sequence:
   cd /home/alleni || echo "In $(pwd)"
   a. no output
   b. **In /home/alleni**
   c. **In $(pwd)**
   d. **In 0pwd)**
   e. **"In $(pwd)"**

137. Which of the following statements is true about this shell command line:
   >foo file bar haven
   a. The command **file** sees three arguments.
   b. The command **file** sees two arguments.
   c. The command **foo** sees three arguments.
   d. The command **foo** sees only two arguments
   e. Error: The command name is missing from the command line.

138. Which Unix command sequence deletes a directory and everything inside it?
   a. **rmdir -all dir**
   b. **deltree -all dir**
   c. **rm -all dir**
   d. **rm -r dir**
   e. **rmdir -r dir**

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

140. If **happy** were a file of text containing 50 different lines, what would be the output of this exact command line: **diff happy happy**
   a. an error message because **diff** doesn’t allow the same file name twice
   b. the contents of file **happy** would be displayed
   c. an error message because **diff** only allows one file name
   d. several lines, which are the lines that are different between the two files
   e. no output

141. If **x=pig** and **y=dog** then what is the output of the following sequence of **bash** commands: **if $x = $y ; then echo $y ; fi**
   a. no output
   b. **dog**
   c. **test: pig: integer expression expected**
   d. **bash: pig: command not found**
   e. **test: $x: integer expression expected**

142. Which **bash** command line below allows programs in the current directory to execute without preceding the names with **./**?
a. **$PATH=/usr/bin:./$HOME**
b. **$PATH=.:$HOME:/usr/bin**
c. **PATH=./$HOME:/usr/bin**
d. **PATH=/usr/bin/.:$HOME**
e. **PATH=/usr/bin:$HOME:.**

143. If my current working directory is **/home**, and my home directory is **/home/xx**, which of the of the following commands copies the Unix password file into my home directory under the name **foo**?
a. **cp xx/../etc/passwd ../home/xx/foo**
b. **cp xx/../../etc/passwd xx/foo**
c. **cp ../../etc/passwd /xx/foo**
d. **cp ../etc/passwd ../xx/foo**
e. **cp ../home/xx/../etc/passwd ./xx/./foo**

144. What is the output of the following sequence of **bash** commands:
x=1 ; y=2 ; test $x -le $y ; echo $?
a. **test: $x: integer expression expected**
b. no output
c. **1**
d. **0**
e. the number 0 or 1 followed by another 0 or 1 on a new line

145. What is the **bash** shell output of this two-command sequence:
cd /bin && echo "echo $(pwd)"
a. **/bin**
b. **echo /bin**
c. **echo 0pwd)**
d. **echo $(pwd)**
e. no output

146. What is the **bash** shell output of this two-command sequence if run in a directory containing 888 files with names that are all the numbers from **1** to **888** inclusive: **cow="*" ; echo '$cow'**
a. *****
b. the file names **1** through **888**, surrounded by quotes
c. **$cow**
d. the file names **1** through **888**
e. **'$cow'**

147. What is in the file named **file** after this command sequence:
echo a >c ; echo b >>c ; mv c d >file
a. **a** followed by **b**
b. nothing - **file** is empty - no data
c. **b**
d. no such file (nonexistent file)
e. **a**

148. Which of the following shell command lines displays the names in the current directory that are exactly three numeric digits long (and nothing else)?
a. **echo [0-90-90-9]**
b. **echo [1-3][1-3][1-3]**
c. **echo [1-31-31-3]**
d. **echo [0-9][0-9][0-9]**
e. **echo ???**

149. What is the output of the following sequence of **bash** commands:
wc='one two' ; test wc = wc
a. **test: too many arguments**
b. no output
c. **1**
d. **1 2 8 wc**
e. **0**

150. What is the output of the following sequence of **bash** commands:
x=cow ; y=dog ; test -z $x ; echo $?
a. **test: $x: integer expression expected**
b. the number 0 or 1 followed by another 0 or 1 on a new line
c. **0**
d. no output
e. **1**

151. If a **bash** shell script named **foo** contains the line:
if ["$1" = '$2'] ; then echo SAME ; fi
then which of the following command lines will produce **SAME** as output?
a. **./foo bar bar**
b. **./foo '$2' bar**
c. **./foo "bar" 'bar'**
d. **./foo "$1" '$2'**
e. **./foo $2 $2**

152. If **x=8** and **y=9** then which of the following **bash** command lines outputs only the word **foobar** (and nothing else)?
   a. **[x!=x] || echo foobar**
   b. **[x = x] && echo foobar**
   c. **[x -ne y] || echo foobar**
   d. **[x -ne y] && echo foobar**
   e. **[!x = y] && echo foobar**

153. Which **bash** command line below allows programs in the current directory to execute without preceding the names with **./**?
   a. **$PATH=/bin:./$HOME**
   b. **$PATH=.:$HOME:/bin**
   c. **PATH = ./$HOME:/bin**
   d. **PATH = /bin:$HOME:.**
   e. **PATH=/bin:$HOME:.**

154. If **pig=12** and **cat=99** then which of the following **bash** command lines outputs only the word **hi** (and nothing else)?
   a. **[pig!=pig] || echo hi**
   b. **[pig = pig] && echo hi**
   c. **[pig -eq 12] || echo hi**
   d. **[!pig = cat] && echo hi**
   e. **[pig -ne cat] && echo hi**

155. How can you ask the **bash** (Linux) shell to complete commands or file names for you?
   a. You can type the first part of the command or file name and press the **TAB** key.
   b. You can type the first part of the command or file name and press the **ALT** key.
   c. Type **[CONTROL]-[D]** and the shell will present a menu of commands.
   d. Type **[CONTROL]-[ALT]-[DEL]** and the shell will present a menu of commands.
   e. Type **[ALT]-[F2]** the shell will present a menu of commands.

156. How many arguments and options are there to the command:
   ls -lid /p
   a. Two command line arguments, one of which contains three options.
   b. Two arguments: A file name starting with a dash and a **/p** switch option argument.
   c. Two arguments, one of which is a single option name and the other is a pathname.
   d. Two arguments, neither of which is an option.
   e. Three arguments, one of which contains options and one is a pathname.

157. Given this long listing:
   drwxr-xr-x 2048 bin bin 4096 Jan 2 14:22 /dir
   How many subdirectories lie immediately under **/dir**?
   a. 4094
   b. 2048
   c. 4096
   d. there is not enough information shown to answer the question
   e. 2046

158. If **a=9** and **b=9**, which **bash** command sequence correctly compares the two numbers as equal and prints **OK**?
   a. **if [a = b] ; then echo OK ; fi**
   b. **if [$a==$b] ; then echo OK ; fi**
   c. **if test a -eq b ; then echo OK ; fi**
   d. **if (a == b) ; then echo OK ; fi**
   e. **if [$a -eq $b] ; then echo OK ; fi**

Answer Key - DAT 2330 – Ian Allen – Winter 2004 - DAT 2330 Practice Test - 0%

Office use only: 62 11 8 69 30 98 132 142 6 12 91 57 49 48 38 26 16 145 64 66 105 81 97 31 106 76 85 56 35 146 112 21 158 39 109 93 149 114 131 87 129 115 134 96 4 157 117 15 67 75 90 59 79 23 28 123 102 32 65 143 52 61 148 84 82 104 54 3 88 10 71 29 141 125 74 89 135 2 42 100 116 53 111 108 25 18 154 150 63 95 80 51 14 68 1 46 130 24 22 13 156 126 55 155 86 103 124 151 44 147 40 138 77 120 140 47 34 33 45 20 72 133 144 9 128 99 122 92 136 101 127 152 119 19 17 58 110 153 5 121 113 37 27 73 60 83 78 50 139 70 107 43 36 41 118 94 137 7

1. a
2. a
3. c
4. e
5. a
6. c
7. c
8. d
9. b
10. d
11. c
12. c
13. d
14. d
15. c
16. a
17. b
18. e
19. c
20. c
21. a
22. e
23. b
24. e
25. e
26. c
27. c
28. d
29. b
30. e
31. b
32. b
33. e
34. b
35. b
36. d
37. a
38. d
39. c
40. d
41. a
42. a
43. d
44. c
45. b
46. d
47. b
48. d
49. d
50. d
51. a
52. b
53. b
54. b
55. d
56. b
57. e
58. c
59. c
60. c
61. e
62. b
63. c
64. e
65. c
66. c
67. b
68. b
69. b
70. a
71. a
72. b
73. b
74. a
75. d
76. a
77. e
78. d
79. d
80. c

81. e
82. c
83. c
84. c
85. a
86. d
87. c
88. d
89. d
90. a
91. d
92. c
93. e
94. e
95. c
96. a
97. e
98. d
99. e
100. c
101. a
102. e
103. b
104. e
105. b
106. a
107. c
108. b
109. b
110. c
111. e
112. d
113. e
114. c
115. c
116. d
117. e
118. c
119. d
120. a
121. e
122. a
123. d
124. b
125. d
126. a
127. e
128. e
129. b
130. d
131. b
132. c
133. c
134. d
135. d
136. a
137. b
138. d
139. c
140. e
141. d
142. e
143. b
144. d
145. b
146. c
147. b
148. d
149. b
150. e
151. b
152. b
153. e
154. b
155. a
156. a
157. e
158. e

```
Count of a:  24  15%
Count of b:  35  22%
Count of c:  35  22%
Count of d:  34  22%
Count of e:  30  19%

With 5 choices: 158
  1 2 3 4 5 6 7 8 9 10 11
  12 13 14 15 16 17 18 19
  20 21 22 23 24 25 26 27
  28 29 30 31 32 33 34 35
  36 37 38 39 40 41 42 43
  44 45 46 47 48 49 50 51
  52 53 54 55 56 57 58 59
  60 61 62 63 64 65 66 67
  68 69 70 71 72 73 74 75
  76 77 78 79 80 81 82 83
```

```
  84 85 86 87 88 89 90 91
  92 93 94 95 96 97 98 99
  100 101 102 103 104 105
  106 107 108 109 110 111
  112 113 114 115 116 117
  118 119 120 121 122 123
  124 125 126 127 128 129
  130 131 132 133 134 135
  136 137 138 139 140 141
  142 143 144 145 146 147
  148 149 150 151 152 153
  154 155 156 157 158

Macro .cmd splits: 77
Macro .ans splits: 0
```