Evaluation: 99 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 - 99 Questions

This is a practice test containing many practice questions. The real test will contain some questions similar to these. There are probably many more questions in this practice test than there will be time for in the real test. The real test will have approximately one question per minute.

The real test may have some questions unlike anything given here. Knowing the concepts behind the questions is necessary; memorizing these specific answers won’t help.

The answers to this test are in the Answer Key on the last page(s).

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

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

3. If **`foo`** is a file containing the first column of the output of the **`last`** command, which command line shows the most frequent login?
   a. **`cat sort foo | uniq -c | sort -nr | head -1`**
   b. **`uniq -c foo | sort -nr | head -1`**
   c. **`sort | uniq -c | sort -nr | head -1 foo`**
   d. **`sort foo | uniq -c | sort -nr | head -1`**
   e. **`sort foo > uniq -c ; sort -nr uniq | head -1`**

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

5. Which line below puts the count of the number of lines in the password file into the variable **`foo`**?
   a. **`foo=$( awk -F: /etc/passwd | wc -l )`**
   b. **`foo=$( wc /etc/passwd | awk echo $1 )`**
   c. **`foo=$( cat -c /etc/passwd )`**
   d. **`foo=$( wc -l </etc/passwd )`**
   e. **`foo=$( wc -l /etc/passwd | awk "print $1" )`**

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

7. 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 of this shell command sequence:
   ```
   PATH=/etc:/usr/bin:/bin ; pig
   ```
   a. **`foo`** followed by **`hi`**
   b. **`bash: pig: command not found`**
   c. **`foo`**
   d. **`hi`** followed by **`mom`**
   e. **`hi`**

8. In a directory containing one file named **`dog`**, what appears on your screen after this command line? **`2>/dev/null ls nosuchfile`**
   a. **`nosuchfile`**
   b. **`ls: nosuchfile: No such file or directory`**
   c. **`bash: 2>/dev/null: command not found`**
   d. **`dog`**
   e. no output on screen

9. In an empty directory, how many lines are in file **`foo`** after this command line:
   ```
   ls  nosuchfile  .  ..  2>foo
   ```
   a. **`2`**
   b. **`1`**
   c. empty file (no data)
   d. **`3`**
   e. **`4`**

10. What is the output on your screen 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 ’$cow’
    ```
    a. the file names **`1`** through **`888`**, surrounded by quotes
    b. the file names **`1`** through **`888`**
    c. **`’$cow’`**
    d. **`$cow`**
    e. **`*`**

11. What is the correct syntax to redirect both standard output and standard error into the same output file?
    a. `command 2>1 >out`
    b. `command >out 2>1`
    c. `command 2>out >out`
    d. `command 2>&1 >out`
    e. `command >out 2>&1`

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

13. 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 of this shell command sequence:
    `PATH=/usr:/etc:/bin:/usr/bin ; foo`
    a. `mom` followed by `dad`
    b. `dad`
    c. `mom`
    d. `bash: foo: command not found`
    e. `dad` followed by `mom`

14. In an empty directory, what is the output on your screen of this three-command sequence: `touch aa .a ab .b .c ; x=’.a* .b*’ ; echo ’$x’`
    a. `aa .a ab .b`
    b. `$x`
    c. `.a* .b*`
    d. `’.a* .b*’`
    e. `.a .b`

15. 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 of this shell command sequence:
    `PATH=/bin/foo:/usr/bin/foo:/usr ; foo`
    a. `mom`
    b. `mom` followed by `dad`
    c. `dad`
    d. `bash: foo: command not found`
    e. `dad` followed by `mom`

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

17. Which 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 [ grep chars /etc/passwd ] ; then echo OK ; fi`
    e. `if test chars /etc/passwd ; then echo OK ; fi`

18. Which of the following `PATH` statements makes the most sense?
    a. `PATH=/bin:/bin/cat:/usr/bin`
    b. `PATH=/dev/null:/usr/bin:/etc:/bin`
    c. `PATH=/dev:/bin:/usr/bin:/etc`
    d. `PATH=/bin:/usr/bin:/etc/passwd`
    e. `PATH=/bin/ls:/etc:/usr/bin`

19. What is the output on your screen of the following sequence of commands:
    `a=sky ; touch $a ; test -z $a ; echo $?`
    a. `test: $a: integer expression expected`
    b. `1`
    c. `0`
    d. `sky`
    e. no output

20. What is the output on your screen of the following command sequence if run in a directory containing 9 files with names that are all the numbers from `1` to `9` inclusive: `pat="?" ; echo "$pat"`
    a. `$pat`
    b. `"$pat"`
    c. `?`
    d. the file names `1` through `9`
    e. the file names `1` through `9`, surrounded by quotes

21. If variable `a` might contain nothing (a null value - defined but empty), which command sequence correctly tests for this and prints the date?
    a. `if [ ’’’’ = ’’$a’’ ] ; then date ; fi`
    b. `if test "" = "$a" ; then date ; fi`
    c. `if [ "$a" = * ] ; then date ; fi`
    d. `if test "" -eq $a ; then date ; fi`
    e. `if [ $a = /dev/null ] ; then date ; fi`

22. What is the output on your screen of the following sequence of commands:
    `a=9 ; b=9 ; [ $a -le $b ] ; echo $?`
    a. the number 1 or 0 followed by another 1 or 0 on a new line
    b. `0`
    c. no output
    d. `test: $a: integer expression expected`
    e. `1`

23. In an empty directory, how many lines are in file **`out`** after this bash shell command line: **`ls . .. nosuchfile 2>out`**
    a. **`2`**
    b. no output (empty file)
    c. **`3`**
    d. **`1`**
    e. **`4`**

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

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

26. Which line below passes three *separate* arguments to the **`cat`** command when placed inside a shell script named **`foo`** invoked by the command line:
    ```
    ./foo  one  two  three
    ```
    a. **`cat "$#"`**
    b. **`cat "$? $? $?"`**
    c. **`cat "$*"`**
    d. **`cat "$1 $2 $3"`**
    e. **`cat "$@"`**

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

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

29. Which command line tells you the recursive count of all pathnames under the current directory and all subdirectories?
    a. **`wc "$PATH"`**
    b. **`wc .`**
    c. **`wc *`**
    d. **`ls | wc`**
    e. **`find | wc`**

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

31. If a 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 bar bar`**
    c. **`./foo "$1" ’$2’`**
    d. **`./foo ’$2’ bar`**
    e. **`./foo $2 $2`**

32. 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. **`foo`**
    b. **`bat`**
    c. **`cat: bat: No such file or directory`**
    d. no output on screen
    e. **`bash: cat: command not found`**

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

34. If **`a=1`** and **`b=1`**, which command sequence correctly compares the two numbers as equal and prints **`OK`**?
    a. **`if test a == b ; then echo OK ; fi`**
    b. **`if [ a -eq b ] ; then echo OK ; fi`**
    c. **`if [ $a==$b ] ; then echo OK ; fi`**
    d. **`if [ b = a ] ; then echo OK ; fi`**
    e. **`if test $b -eq $a ; then echo OK ; fi`**

35. If variable **`mt`** might contain nothing (a null value - defined but empty), which command sequence correctly tests for this and prints **`OK`**?

a. `if [ "$mt" = * ] ; then echo OK ; fi`
b. `if [ "$mt" = "" ] ; then echo OK ; fi`
c. `if [ $mt -eq "" ] ; then echo OK ; fi`
d. `if [ ’’$mt’’ = ’’’’ ] ; then echo OK ; fi`
e. `if [ $mt -eq : ] ; then echo OK ; fi`

36. Which command sequence below always outputs just the date only if the first argument is either readable or executable?

a. `if [ -r || -x "$1" ]; then date ; fi`
b. `if [ "$1" -eq -r -o "$1" -eq -x ]; then date ; fi`
c. `if [ -r -o -x "$1" ]; then date ; fi`
d. `if [ -r "$1" -o -x "$1" ]; then date ; fi`
e. `if [ "-r $1" || "-x $1" ]; then date ; fi`

37. If **`x=5`** and **`y=5`**, which command sequence correctly compares the two numbers as equal and prints **`OK`**?

a. `if test x -eq y ; then echo OK ; fi`
b. `if [ $x==$y ] ; then echo OK ; fi`
c. `if [ x = y ] ; then echo OK ; fi`
d. `if ( x == y ) ; then echo OK ; fi`
e. `if test $x -eq $y ; then echo OK ; fi`

38. 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 on your screen from this line?

a. `3 4`
b. `2 3 4`
c. `1 2 3`
d. `$3`
e. `"3`

39. What is the output on your screen of the following sequence of commands:

```
a=cow ; b=dog ; touch $a ; test -z $a ; echo $?
```

a. `1`
b. the number 1 or 0 followed by another 1 or 0 on a new line
c. `0`
d. no output
e. `test: $a: integer expression expected`

40. What is the output on your screen of the following command sequence:

```
x=1 ; y=2 ; test $x -le $y ; echo $?
```

a. `1`
b. `test: $x: integer expression expected`
c. `0`
d. no output
e. the number 0 or 1 followed by another 0 or 1 on a new line

41. Which command sequence correctly compares the two numbers and prints **`OK`**?

a. `if ( let 4 > 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`

42. Which command line shows just the count of lines in the file?

a. `wc file | awk ’{print #1}’`
b. `wc file | awk ’[print $1]’`
c. `wc file | awk ’{print $1}’`
d. `wc file | awk ’{print 1}’`
e. `wc file | awk ’[print #1]’`

43. What is the output on your screen of this two command sequence:

```
PATH=/bin/cat:/bin/sh:/bin/ls ; ls nosuchfile
```

a. `bash: ls: command not found`
b. `bash: /bin/ls: command not found`
c. `ls: nosuchfile: No such file or directory`
d. `bash: /bin/sh: No such file or directory`
e. `ls: /bin/ls: command not found`

44. What is the correct syntax to redirect both standard output and standard error into the same output file?

a. `wc >out 2>1 foo`
b. `wc 2>1 >out foo`
c. `wc >out 2>out foo`
d. `wc >out 2>&1 foo`
e. `wc 2>&1 >out foo`

45. If a 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 bar ’$1’`
e. `./foo 1 "$1"`

46. What is the output on your screen of the following command sequence:

```
a=sky ; touch $a ; test -z $a ; echo $?
```

a. no output
b. `1`
c. `test: $a: integer expression expected`
d. `sky`
e. `0`

47. Which 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 && -d "$1" ]; then date ; fi`
    c. `if [ -d "$1" -a -s "$1" ]; then date ; fi`
    d. `if [ "-s $1" && "-d $1" ]; then date ; fi`
    e. `if [ "$1" -eq -f -a "$1" -eq -d ]; then date ; fi`

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

49. Which command tells you the count of lines in the **bash** manual page?
    a. `whereis bash | wc`
    b. `man bash | wc`
    c. `which bash | wc`
    d. `apropos bash | wc`
    e. `man bash > wc ; cat wc`

50. How many arguments are passed to the command by the shell on this command line:

```
<bar bar -b "-a" ’-r’ >bar bar bar
```

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

51. 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 "$*"`
    b. `sort "$#"`
    c. `sort "$? $? $?"`
    d. `sort "$1 $2 $3"`
    e. `sort "$@"`

52. What is the output on your screen of the following sequence of commands:

```
a=pig ; b=bat ; touch $b ; test -n $b ; echo $?
```

    a. **1**
    b. `test: $b: integer expression expected`
    c. **0**
    d. no output
    e. the number 0 or 1 followed by another 0 or 1 on a new line

53. What is the output on your screen of the following sequence of commands:

```
x=cow ; y=dog ; touch $x ; test -z $x ; echo $?
```

    a. the number 0 or 1 followed by another 0 or 1 on a new line
    b. no output
    c. `test: $x: integer expression expected`
    d. **1**
    e. **0**

54. If **a=1** and **b=1**, which 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 -eq $b ] ; then echo OK ; fi`
    e. `if [ $a==$b ] ; then echo OK ; fi`

55. In an empty directory, how many lines are in file **bar** after this command line:

```
ls  .  nosuchfile  1>bar
```

    a. **1**
    b. **3**
    c. **4**
    d. empty file (no data)
    e. **2**

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

57. 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 of this shell command sequence:

```
PATH=/etc:/usr/bin:/bin ; pig
```

    a. **foo** followed by **hi**
    b. `bash: pig: command not found`
    c. **hi**
    d. **foo**
    e. **hi** followed by **foo**

58. If **a=ant** and **b=bat** then what is the output on your screen of the following command sequence: `[ $a = ant -a $b = ant ] ; echo $?`
    a. **1**
    b. no output
    c. **0**
    d. `test: $a: integer expression expected`
    e. the number 1 or 0 followed by another 1 or 0 on a new line

59. In an empty directory, what appears on your screen after this command line?

```
      ls out 2>/dev/null
```

a. no output
b. `ls: /dev/null: No such file or directory`
c. `ls: out: No such file or directory`
d. `ls: out 2>/dev/null: No such file or directory`
e. `out`

60. Which correct command sequence below always outputs just the date only if the first argument is **both** not empty **and** a directory?

a. `if [ -s && -d "$1" ]; then date ; fi`
b. `if [ "$1" -eq -f -a "$1" -eq -d ]; then date ; fi`
c. `if [ -d "$1" -a -s "$1" ]; then date ; fi`
d. `if [ "-s $1" && "-d $1" ]; then date ; fi`
e. `if [ -s -a -d "$1" ]; then date ; fi`

61. In an empty directory, what appears on your screen after this command line?

```
      ls 1>/dev/null nosuchfile
```

a. no output
b. `ls: 1>/dev/null nosuchfile: No such file or directory`
c. `ls: /dev/null: No such file or directory`
d. `nosuchfile`
e. `ls: nosuchfile: No such file or directory`

62. What is the output on your screen 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. `*`
b. the file names `1` through `765`
c. all the file names that start with an asterisk (’`*`’)
d. an asterisk (’`*`’) and the file names `1` through `765`
e. `$foo`

63. Which of these first lines will cause this executable file to be interpreted using the Bash shell?

a. `#!/bin/bash`
b. `#/bin/bash`
c. `/bin/bash -u`
d. `!/bin/bash`
e. `!#/bin/bash -u`

64. Which command sequence correctly searches for `foo` and then prints the date if it is found inside the file `bar`?

a. `if [ grep foo bar ] ; then date ; fi`
b. `if grep <bar foo ; then date ; fi`
c. `if [ test foo bar ] ; then date ; fi`
d. `if test foo bar ; then date ; fi`
e. `if test foo = bar ; then date ; fi`

65. If `a=cow` and `b=dog` then what is the output on your screen of the following sequence of commands: `[ $a = dog -o $b = dog ] ; echo $?`

a. no output
b. the number 1 or 0 followed by another 1 or 0 on a new line
c. `1`
d. `test: $a: integer expression expected`
e. `0`

66. Which command tells you the full absolute pathname of the `lynx` command?

a. `absolute "$PATH" | grep lynx`
b. `echo "$PATH" | grep lynx`
c. `absolute lynx`
d. `whereis | grep lynx`
e. `whereis lynx`

67. Which command line shows just the type and permissions of file `foo`?

a. `ls -l foo | awk ’ ’ ’\n’`
b. `tr ’ ’ ’\n’ <ls -l foo | head -1`
c. `ls -l foo | tr ’ ’ ’\n’ | head -1`
d. `cat foo | ls -l | awk ’ ’ ’\n’ | head -1`
e. `ls -l foo | awk ’ ’ ’\n’ | head -1`

68. 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 of this command sequence:

```
    PATH=/dev:/usr/bin:/usr:/bin:/etc ; /bin/foo
```

a. `two` followed by `one`
b. `one` followed by `two`
c. `one`
d. `two`
e. `bash: /bin/foo: command not found`

69. 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 of this shell command sequence:

```
    PATH=/home:/bin:/dev:/usr/bin ; pig
```

a. `foo`
b. `xx` followed by `foo`
c. `foo` followed by `xx`
d. `xx`
e. `bash: pig: command not found`

70. If your `PATH` variable contains `/bin:/usr/bin`, what is the output on your screen of this command line: `echo ’$PATH’`

a. `/bin:/usr/bin`
b. `’/bin:/usr/bin’`
c. `$PATH`
d. `echo: $PATH: No such file or directory`
e. `’$PATH’`

71. In an empty directory, what is the output on your screen of these commands:

```
touch uu .u uv .v uw ; a="*u *v" ; echo "$a"
```

   a. `uu uv`
   b. `u* v*`
   c. `uu .u uv .v`
   d. `*u *v`
   e. `$a`

72. In a directory containing one file named `dog`, what appears on your screen after this command line? `1>/dev/null ls *`
   a. `bash: 1>/dev/null: command not found`
   b. no output on screen
   c. `ls: *: No such file or directory`
   d. `dog`
   e. `*`

73. Which command sequence correctly searches for the `chars` and then prints `OK` if it is found inside the password file?
   a. `if [ grep 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 test chars /etc/passwd ; then echo OK ; fi`

74. What is the output on your screen of the following sequence of commands:

```
x=pig ; y=bat ; touch $x ; [ -z $x ] ; echo $?
```

   a. `0`
   b. `test: $x: integer expression expected`
   c. `1`
   d. no output
   e. the number 0 or 1 followed by another 0 or 1 on a new line

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

76. A shell script named `bar` is executed as follows:

```
./bar "a b" "c d e" f
```

   Inside the script is the line: `echo "$2"`
   What is the output on your screen from this line?
   a. `$2`
   b. `a b`
   c. `b"`
   d. `c d e`
   e. `b`

77. If a shell script named `foo` contains the line:

```
if [ ’$3’ = "$2" ] ; then echo SAME ; fi
```

   then which of the following command lines will always produce `SAME` as output?
   a. `./foo $1 ’$2’ $3`
   b. `./foo 2 ’$3’ 1`
   c. `./foo $3 "$2" $1`
   d. `./foo $1 $2 $3`
   e. `./foo ’$1’ "$3" $2`

78. What is the output on your screen of the following command sequence:

```
a=1 ; b=2 ; test $a -ge $b ; echo $?
```

   a. `0`
   b. `1`
   c. the number 1 or 0 followed by another 1 or 0 on a new line
   d. no output
   e. `test: $a: integer expression expected`

79. What is the output on your screen of the following command sequence:

```
f=1 ; touch f ; test ! -z $f ; echo $?
```

   a. the number 1 or 0 followed by another 1 or 0 on a new line
   b. no output
   c. `0`
   d. `1`
   e. `test: $f: integer expression expected`

80. What is the output on your screen of the following sequence of commands:

```
x=0 ; y=1 ; touch $x ; test ! -z $x ; echo $?
```

   a. `1`
   b. no output
   c. `0`
   d. the number 1 or 0 followed by another 1 or 0 on a new line
   e. `test: $x: integer expression expected`

81. If `/bin/bat` is a program that outputs `hi` and `/usr/bin/bat` is a program that outputs `foo` what is the output on your screen of this shell command sequence:

```
PATH=/etc:/usr/bin:/bin ; bat
```

   a. `hi`
   b. `hi` followed by `mom`
   c. `bash: bat: command not found`
   d. `foo`
   e. `foo` followed by `hi`

82. What is the output on your screen of the following sequence of commands:

```
x=cow ; y=dog ; test -z $x ; echo $?
```

   a. `test: $x: integer expression expected`
   b. no output
   c. `1`
   d. the number 0 or 1 followed by another 0 or 1 on a new line
   e. `0`

83. Which command line shows the file in `/bin` with the largest checksum?

a. `cat /bin | sum | sort -nr | head -1`
b. `sum /bin/* | sort -nr | head -1`
c. `cat /bin/* | sum | sort -nr | head -1`
d. `sum /bin | sort -nr | head -1`
e. `ls /bin/* | sum | sort -nr | head -1`

84. 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 of this command sequence:

```
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`

85. Which line below is most likely to be the beginning of an error message?

a. `echo 1<&2 "... "`
b. `echo 2>$1 "... "`
c. `echo 1>&2 "... "`
d. `echo 2<$1 "... "`
e. `echo 2>&1 "... "`

86. What is the output on your screen of the following sequence of commands:

```
x=1 ; touch x ; test ! -z $x ; echo $?
```

a. the number 1 or 0 followed by another 1 or 0 on a new line
b. `1`
c. no output
d. `test: $x: integer expression expected`
e. `0`

87. Which command sequence correctly compares the numbers and prints `OK`?

a. `if ( 1 let 2 ) ; then echo OK ; fi`
b. `if [ 1 -lt 2 ] ; then echo OK ; fi`
c. `if ( let 2 > 1 ) ; then echo OK ; fi`
d. `if [ ! 2 < 1 ] ; then echo OK ; fi`
e. `if [ 2 > 1 ] ; then echo OK ; fi`

88. Which command sends a file to a remote machine `foo.ca`?

a. `cat one >foo.ca:two`
b. `cp one foo.ca:two`
c. `mv one foo.ca:two`
d. `scp one >foo.ca:two`
e. `scp one foo.ca:two`

89. What is the correct syntax to redirect both standard output and standard error into the same output file?

a. `sum 2>1 >out foo`
b. `sum foo 1>out 2>1`
c. `sum >out foo 2>&1`
d. `sum 2>&1 foo >out`
e. `sum 1>out 2>out foo`

90. What is the output on your screen 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. `*`
b. the file names `1` through `123`, surrounded by quotes
c. `$bat`
d. `"$bat"`
e. the file names `1` through `123`

91. If the file `bat` contained the word `foo`, what would be the output on your screen of this two command sequence:

```
PATH=/bin/ls:/bin/who:/etc/passwd ; /bin/ls bat
```

a. `/bin/ls: bat: No such file or directory`
b. `foo`
c. `bat`
d. no output
e. `bash: /bin/ls: command not found`

92. 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 of this command sequence:

```
PATH=/etc:/usr/bin:/usr:/bin:/dev ; foo
```

a. `two` followed by `one`
b. `one`
c. `one` followed by `two`
d. `two`
e. `bash: foo: command not found`

93. What is the output on your screen of the following sequence of commands:

```
x=1 ; y=2 ; test $x -le $y ; echo $?
```

a. `0`
b. the number 0 or 1 followed by another 0 or 1 on a new line
c. `test: $x: integer expression expected`
d. `1`
e. no output

94. 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
```

a. **foo**
b. no output
c. **bat**
d. **bash: /bin/ls: command not found**
e. **/bin/ls: bat: No such file or directory**

95. What is the output on your screen of the following sequence of commands:

```
    x=cow ; y=dog ; touch $y ; test -n $y ; echo $?
```

a. no output
b. **0**
c. **1**
d. **test: $y: integer expression expected**
e. the number 0 or 1 followed by another 0 or 1 on a new line

96. In an empty directory, what is the length of the longest file name created by the following two-command sequence:

```
    xx="1234 123 12 1" ; touch ’$xx’
```

a. 1 character
b. 13 characters
c. 4 characters
d. 3 characters
e. 2 characters

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

98. If **a=ant** and **b=bat** then what is the output on your screen of the following command sequence: **[$a = bat -o $b = bat] ; echo $?**
a. **0**
b. no output
c. the number 1 or 0 followed by another 1 or 0 on a new line
d. **1**
e. **test: $a: integer expression expected**

99. 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**

This page intentionally left blank.

Answer Key - NET 2003 − Ian Allen − Winter 2007 - NET 2003 Practice Test

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

```
  97. a
  98. a
  99. a

Count of a:  19  19%
Count of b:  22  22%
Count of c:  23  23%
Count of d:  20  20%
Count of e:  15  15%

With 5 choices: 99

Macro .cmd split with good indent: 49
```