Evaluation: 54 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 - 54 Questions

This is a practice test containing many practice questions. The real test will contain some questions similar to these. 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.

(Office use only: 51 22 48 3 27 54 26 9 31 16 53 18 8 19 36 33 28 11 52 47 7 30 5 23 17 32 39 20 50 35 43 49 40 37 46 4 15 21 45 2 6 42 1 29 44 24 10 25 13 14 12 41 34 38)

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

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

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

4. What is the correct syntax to redirect both standard output and standard error into the same output file?
   a. **`ls -l >foo 2>foo`**
   b. **`ls -l >foo 2>&1`**
   c. **`ls -l >foo 2>$1`**
   d. **`ls -l 2>$1 >foo`**
   e. **`ls -l 2>&1 >foo`**

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

6. 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. **`-wx`**
   c. **`r-x`**
   d. **`rw-`**
   e. **`--x`**

7. How many arguments and options are there to the command:
   `sort -r <infile`
   a. One command line argument containing one option name.
   b. Three arguments, one of which contains an option and one is a pathname.
   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. A file name starting with a dash and an **`<infile`** switch option argument.

8. 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 ???`**
   b. **`echo [0-9][0-9][0-9]`**
   c. **`echo [1-31-31-3]`**
   d. **`echo [1-3][1-3][1-3]`**
   e. **`echo [0-90-90-9]`**

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

10. Given my directory **`dir`** and my file **`dir/bar`** owned by me, which permissions allow me to access and change the content (data) in the file **`dir/bar`** but not delete the file?
    a. Permissions **`300`** on directory **`dir`** and **`500`** on file **`dir/bar`** .
    b. Permissions **`100`** on directory **`dir`** and **`200`** on file **`dir/bar`** .
    c. Permissions **`600`** on directory **`dir`** and **`200`** on file **`dir/bar`** .
    d. Permissions **`700`** on directory **`dir`** and **`300`** on file **`dir/bar`** .
    e. Permissions **`500`** on directory **`dir`** and **`500`** on file **`dir/bar`** .

11. Which command stops people from using **write** to put lines of text on your screen?
    a. **umask 000**
    b. **chmod 000**
    c. **write n**
    d. **stop 0**
    e. **mesg n**

12. If the file **bat** contained the word **foo**, what would be the **bash** shell output of this two command sequence:

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

    a. **bash: /bin/ls: command not found**
    b. **/bin/ls: bat: No such file or directory**
    c. **foo**
    d. **bat**
    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-Z][a-zA-Z][a-zA-Z]**
    c. **echo [azAZ][azAZ][azAZ]**
    d. **echo [a-zA-Za-zA-Za-zA-Z]**
    e. **echo [a,zA,Z][a,zA,Z][a,zA,Z]**

14. If the file **foo** contained the word **mom**, what would be the **bash** shell output of this two command sequence:

    ```
    PATH=/bin/ps:/bin/echo:/bin/ls ; /bin/ls foo
    ```

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

15. 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** doesn't allow the same file name twice
    b. the contents of file **foo** 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

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. **5**
    c. **3**
    d. **7**
    e. **6**

17. What is the output of this sequence of three shell commands:

    ```
    umask 547 ; mkdir newdir ; ls -ld newdir
    ```

    a. **d-w--w---- 1 me me 0 Feb 20 07:55 newdir**
    b. **dr--r--rw- 1 me me 0 Feb 20 07:55 newdir**
    c. **d-w--wxrwx 1 me me 0 Feb 20 07:55 newdir**
    d. **dr-xr--rwx 1 me me 0 Feb 20 07:55 newdir**
    e. **d-w--wx--- 1 me me 0 Feb 20 07:55 newdir**

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

19. 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 "... "**

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

21. 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 [a,zA,Z][a,zA,Z][a,zA,Z]**
    b. **echo [azAZ][azAZ][azAZ]**
    c. **echo ???**
    d. **echo [0-89][01-9][0-45-9]**
    e. **echo [a-mn-zA-YZ][ab-zAB-YZ][za-yZA-Y]**

22. What is the output of this sequence of three shell commands:

    ```
    umask 674 ; touch newfile ; ls -l newfile
    ```

    a. **--w--wxr-x 1 me me 0 Feb 20 07:55 newfile**
    b. **--------w- 1 me me 0 Feb 20 07:55 newfile**
    c. **-rw-rwxr-- 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**

23. 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. **$PATH=/usr/bin:./$HOME**
   d. **PATH=/usr/bin:.:$HOME**
   e. **$PATH=$HOME:/usr/bin:.**

24. If **/bin/foo** is a program that outputs **mom** and **/usr/bin/foo** is a program that outputs **dad** what is the output of this shell command sequence:

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

   a. **bash: foo: command not found**
   b. **mom** followed by **dad**
   c. **mom**
   d. **dad** followed by **mom**
   e. **dad**

25. 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. no output
   c. **bash: /bin/ls: command not found**
   d. **foo**
   e. **pig**

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

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

   a. **4**
   b. **6**
   c. **7**
   d. **3**
   e. **5**

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

```
ls 2>/dev/null nosuchfile
```

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

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

29. 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. **:dp**
   b. **ddP**
   c. **DDP**
   d. **DDp**
   e. **ddp**

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

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

   a. **7**
   b. **5**
   c. **6**
   d. **4**
   e. **3**

31. If **cat** is an executable script containing the line:

```
TERM=linux ; export TERM
```

what is the output of the following sequence of commands:

```
TERM=vt100 ; ./cat ; echo "$TERM"
```

   a. **vt100**
   b. **$TERM**
   c. **linux**
   d. **cat**
   e. **TERM**

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

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

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

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

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

35. If your PATH variable contains **/bin:/usr/bin** , what is the output of this **bash** shell command line? **echo ’$PATH’**
   a. **echo: $PATH: No such file or directory**
   b. **/bin:/usr/bin**
   c. **’$PATH’**
   d. **’/bin:/usr/bin’**
   e. **$PATH**

36. 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:.**

37. Given my directory **dir** and my file **dir/foo** owned by me, which permissions allow me to access and change the content (data) in the file **dir/foo** but not delete the file?
   a. Permissions **400** on directory **dir** and **400** on file **dir/foo** .
   b. Permissions **500** on directory **dir** and **600** on file **dir/foo** .
   c. Permissions **300** on directory **dir** and **200** on file **dir/foo** .
   d. Permissions **600** on directory **dir** and **700** on file **dir/foo** .
   e. Permissions **100** on directory **dir** and **100** on file **dir/foo** .

38. If **/bin/bat** is a program that outputs **foo** and **/usr/bin/bat** is a program that outputs **hi** what is the output of this shell command sequence:

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

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

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

40. What is the correct syntax to redirect both standard output and standard error into the same output file?
   a. **wc 2>&1 >out foo**
   b. **wc >out 2>1 foo**
   c. **wc >out 2>&1 foo**
   d. **wc >out 2>out foo**
   e. **wc 2>1 >out foo**

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

42. If **bar** is an executable script containing the line:

```
TERM=vt100 ; export TERM
```

what is the output of the following sequence of commands:

```
TERM=linux ; ./bar ; echo "$TERM"
```

   a. **linux**
   b. **./bar**
   c. **vt100**
   d. **TERM**
   e. **$TERM**

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

44. What is the output of this sequence of three shell commands:

```
umask 457 ; mkdir newdir ; ls -ld newdir
```

a. `d-w--w---- 2 me me 512 Oct 1 1:12 newdir`
b. `d-wx-w-rwx 2 me me 512 Oct 1 1:12 newdir`
c. `dr--r-xrwx 2 me me 512 Oct 1 1:12 newdir`
d. `dr-xr-xrwx 2 me me 512 Oct 1 1:12 newdir`
e. `d-wx-w---- 2 me me 512 Oct 1 1:12 newdir`

45. If **`dog`** is an executable script containing the line: **`umask 0777`** what is the output of the following sequence of commands:

```
umask 0022 ; ./dog ; umask
```

a. `0777`
b. `0022`
c. `0755`
d. nothing; no output
e. `0799`

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

47. Which of the following shell command lines displays only the names in the current directory that are exactly three alphabetic letters long?

a. `echo [?][?][?]`
b. `echo ???`
c. `echo [a-zA-Z][a-zA-Z][a-zA-Z]`
d. `echo ***`
e. `echo [0-3][0-3][0-3]`

48. What is the **`bash`** shell output of this two command sequence:

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

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

49. 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 **`300`** on directory **`dir`** and **`500`** on file **`dir/foo`** .
b. Permissions **`100`** on directory **`dir`** and **`100`** on file **`dir/foo`** .
c. Permissions **`300`** on directory **`dir`** and **`300`** on file **`dir/foo`** .
d. Permissions **`500`** on directory **`dir`** and **`400`** on file **`dir/foo`** .
e. Permissions **`100`** on directory **`dir`** and **`200`** on file **`dir/foo`** .

50. 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 **`300`** on directory **`dir`** and **`200`** on file **`dir/bar`** .
b. Permissions **`100`** on directory **`dir`** and **`300`** on file **`dir/bar`** .
c. Permissions **`300`** on directory **`dir`** and **`400`** on file **`dir/bar`** .
d. Permissions **`500`** on directory **`dir`** and **`500`** on file **`dir/bar`** .
e. Permissions **`100`** on directory **`dir`** and **`500`** on file **`dir/bar`** .

51. Which of these statements is true?

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

52. If **`foo`** is an executable script containing the line:

```
PATH=/bin ; export PATH
```

what is the output of the following sequence of commands:

```
PATH=/etc ; ./foo ; echo "$PATH"
```

a. `/bin`
b. `/etc:/bin`
c. `/etc`
d. `$PATH`
e. `foo`

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

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

a. `8`
b. `7`
c. `6`
d. `5`
e. `9`

54. In an empty directory, what appears on your screen after this **`bash`** command line?

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

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

Answer Key - DAT 2330 − Ian Allen − Fall 2004 - DAT 2330 Practice Test - 0%

Office use only: 51 22 48 3 27 54 26 9 31 16 53 18 8 19 36 33 28 11 52 47 7 30 5 23 17 32 39 20 50 35 43 49 40 37 46 4 15 21 45 2 6 42 1 29 44 24 10 25 13 14 12 41 34 38

```
  1. d
  2. b
  3. c
  4. b
    5. a
    6. c
    7. a
    8. b
  9. c
 10. b
 11. e
 12. d
   13. b
   14. b
   15. e
   16. a
 17. e
 18. b
 19. e
 20. c
   21. e
   22. b
   23. d
   24. a
 25. e
 26. e
 27. a
 28. c
   29. e
   30. e
   31. a
   32. b
 33. b
 34. e
 35. e
 36. d
   37. b
   38. a
   39. b
   40. c
 41. b
 42. a
 43. d
 44. e
   45. b
   46. c
   47. c
   48. b
 49. a
 50. c
 51. c
 52. c
   53. c
   54. e
```

```
Count of a:      9  1 7%
Count of b:      15  28%
Count of c:      12  22%
Count of d:      5  9 %
Count of e:      13  24%

With 5 choices: 54
  1 2 3 4 5 6 7 8 9 1   0 1 1 1 2
  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

Macro .cmd splits: 31
Macro .ans splits: 0
```