Evaluation: 44 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 - 44 Questions - 9 of 15%

(Office use only: 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)

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

2. Which command line below allows the shell to execute programs in the current directory without preceding the names with **`./`**?
   † **a**. **`PATH=/etc:/dev:.`**
   b. **`PATH=/usr/bin/.:/dev/null`**
   c. **`PATH=./dev:/usr/bin`**
   d. **`PATH=/bin/.:..`**
   e. **`PATH=./dev:./bin`**

3. Which of the following shell command lines removes all the names in the current directory that are exactly three letters (alphabetic) long (and nothing else)?
   † **a**. **`rm [a-zA-Z][a-zA-Z][a-zA-Z]`**
   b. **`rm [a-zA-Z,a-zA-Z,a-zA-Z]`**
   c. **`rm [azAZ][azAZ][azAZ]`**
   d. **`rm [3][3][3]`**
   e. **`rm ???`**

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

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

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

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

8. If the file **`bat`** contained the word **`foo`**, what would be the output of this two command sequence:
   `PATH=/bin/cat:/bin/who:/bin/ls ; cat bat`
   † **a**. **`bash: cat: command not found`**
   b. **`bat`**
   c. **`foo`**
   d. **`cat: bat: No such file or directory`**
   e. no output on screen

9. If the file **`foo`** contained the word **`mom`**, what would be the output of this two command sequence:
   `PATH=/bin/ps:/bin/echo:/bin/ls ; /bin/ls foo`
   † **a**. **`foo`**
   b. **`mom`**
   c. **`/bin/ls: foo: No such file or directory`**
   d. **`bash: /bin/ls: command not found`**
   e. no output on screen

10. If **/bin/foo** is a program that outputs **one** and **/usr/bin/foo** is a program that outputs **two**, what is the output of this command sequence:
```
PATH=/etc:/usr/bin:/usr:/bin:/dev ; foo
```
† **a**. **two**
b. **one**
c. **two** followed by **one**
d. **one** followed by **two**
e. **bash: foo: command not found**

11. If **/bin/foo** is a program that outputs **one** and **/usr/bin/foo** is a program that outputs **two**, what is the output of this command sequence:
```
PATH=/dev:/usr/bin:/usr:/bin:/etc ; /bin/foo
```
† **a**. **one**
b. **two**
c. **two** followed by **one**
d. **one** followed by **two**
e. **bash: /bin/foo: command not found**

12. If **/bin/foo** is a program that outputs **one** and **/usr/bin/foo** is a program that outputs **two**, what is the output of this command sequence:
```
PATH=/bin/ls:/home:/usr/bin/cat:/etc ; foo
```
† **a**. **bash: foo: command not found**
b. **one**
c. **two**
d. **two** followed by **one**
e. **one** followed by **two**

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

14. How many arguments and options are there to the command:
```
wc <infile -wc >wc
```
† **a**. One argument: a single option argument with two option letters.
b. Three arguments: two file names and one option argument with two options.
c. Three arguments, each of which is a pathname argument.
d. Four arguments, only one of which is an option argument with two options.
e. Two arguments: an input file and an option argument with two options.

15. What is the output of this sequence of three shell commands:
```
umask 574 ; mkdir newdir ; ls -ld newdir
```
† **a**. **d-w-----wx 1 me me 0 Oct 1 07:55 newdir**
b. **dr-xrwxr-- 1 me me 0 Oct 1 07:55 newdir**
c. **dr--rw-r-- 1 me me 0 Oct 1 07:55 newdir**
d. **d-w-----w- 1 me me 0 Oct 1 07:55 newdir**
e. **d-w-rwx-wx 1 me me 0 Oct 1 07:55 newdir**

16. What is the output of this sequence of three shell commands:
```
umask 162 ; touch newfile ; ls -l newfile
```
† **a**. **-rw----r-- 1 me me 0 Oct 1 01:12 newfile**
b. **---xrw--w- 1 me me 0 Oct 1 01:12 newfile**
c. **----rw--w- 1 me me 0 Oct 1 01:12 newfile**
d. **-rw---xr-x 1 me me 0 Oct 1 01:12 newfile**
e. **-rw---x-w- 1 me me 0 Oct 1 01:12 newfile**

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

18. If **one** were a file of text containing 10 different lines, what would be the output of this command line: **cp one two ; diff one two**
† **a**. no output on screen
b. an error message because **diff** only allows one file name
c. an error message because **diff** doesn't allow different file names
d. several lines, which are the lines that are different between the two files
e. the contents of one of the files would be displayed

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

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

21. If **foo** is an executable script containing the line:
```
PATH=/etc ; export PATH
```
what is the output of the following sequence of commands:
```
PATH=/bin ; ./foo ; echo "$PATH"
```
† **a**. **/bin**
b. **/etc**
c. **./foo**
d. **/bin:/etc**
e. **$PATH**

22. If **dog** is an executable script containing the line: **umask 0002**
what is the output of the following sequence of commands:
```
umask 0077 ; ./dog ; umask
```
† **a**. **0077**
b. **0002**
c. **0079**
d. **0075**
e. no output on screen

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

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

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

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

27. Which of these commands makes a file owned by me, also readable by me?
† **a**. **chmod u+r ./myfile**
b. **chmod r+u myfile**
c. **chmod r=u ./myfile**
d. **umask 400 myfile**
e. **umask 300 ./myfile**

28. 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**

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

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

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

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

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

34. 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]'**

35. Which command line shows the file in **/bin** with the largest checksum?
† **a**. **sum /bin/* | sort -nr | head -1**
b. **sum /bin | sort -nr | head -1**
c. **cat /bin/* | sum | sort -nr | head -1**
d. **cat /bin | sum | sort -nr | head -1**
e. **ls /bin/* | sum | sort -nr | head -1**

36. Which command line shows just the type and permissions of file **foo**?
† **a**. **ls -l foo | tr ' ' '\n' | head -1**
b. **ls -l foo | awk ' ' '\n'**
c. **ls -l foo | awk ' ' '\n' | head -1**
d. **tr ' ' '\n' <ls -l foo | head -1**
e. **cat foo | ls -l | awk ' ' '\n' | head -1**

37. Which command line shows the current date?
† **a**. **echo date | bash**
b. **date | bash**
c. **bash date**
d. **bash <date**
e. **bash >date ; cat date**

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

39. Which command tells you the full absolute pathname of the **lynx** command?
† **a**. **whereis lynx**
b. **absolute lynx**
c. **whereis | grep lynx**
d. **echo "$PATH" | grep lynx**
e. **absolute "$PATH" | grep lynx**

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

41. Which command counts the number of Unix permission groups you are in?
† **a**. **groups | wc**
b. **umask | wc**
c. **id | wc**
d. **echo groups | wc**
e. **wc groups**

42. Which command sends a file to a remote machine **foo.ca**?
† **a**. **scp one foo.ca:two**
b. **scp one >foo.ca:two**
c. **cp one foo.ca:two**
d. **mv one foo.ca:two**
e. **cat one >foo.ca:two**

43. 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**. **sort foo | uniq -c | sort -nr | head -1**
b. **cat sort foo | uniq -c | sort -nr | head -1**
c. **uniq -c foo | sort -nr | head -1**
d. **sort | uniq -c | sort -nr | head -1 foo**
e. **sort foo > uniq -c ; sort -nr uniq | head -1**

44. Which command line locates scripts in the **`/bin`** directory?

† **a**. **`file /bin/* | grep script`**
b. **`file /bin | grep script`**
c. **`cat /bin/* | file | grep script`**
d. **`cat /bin | file | grep script`**
e. **`ls /bin/* | file | grep script`**

This page intentionally left blank.

Answer Key - DAT 2330 – Ian Allen – Fall 2004 - DAT 2330 Unix Test #2 - 15%

Office use only: 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

1. a
2. a
3. a
4. a
5. a
6. a
7. a
8. a
9. a
10. a
11. a
12. a
13. a
14. a
15. a
16. a
17. a
18. a
19. a
20. a
21. a
22. a
23. a
24. a
25. a
26. a
27. a
28. a
29. a
30. a
31. a
32. a
33. a
34. a
35. a
36. a
37. a
38. a
39. a
40. a
41. a
42. a
43. a
44. a

```
Count of a:  44  100%

With 5 choices: 44
  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

Macro .cmd splits: 15
Macro .ans splits: 0
```