

**Evaluation: Part I - 44 M/C Questions**

Name: \_\_\_\_\_

**Important Instructions**

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

**Multiple Choice - 44 Questions - 10 of 10%**

(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. How many arguments and options are there to the command:  
`wc -wc /sort`
  - † a. Two command line arguments, one of which contains two bundled options.
  - b. Two arguments, one of which is a single option and the other is a pathname.
  - c. A three-letter file name and a `/sort` switch option argument.
  - d. Two arguments, no options.
  - e. Two options, no arguments.
2. If `/bin/bash` is a file name, which of the following pathnames always leads to the same file?
  - † a. `/bin/../bin/bash`
  - b. `/bin/bin/../bash`
  - c. `/bin/bash/.`
  - d. `./bin/bash`
  - e. `/bin/../bash`
3. If `foo` is a sub-directory that contains only the file `pig`, what happens after this command: `mv foo/pig foo/./dog`
  - † a. the directory `foo` now contains only a file named `dog`
  - b. the directory `foo` is now empty
  - c. there is a second copy of the file `pig` in the file named `dog`
  - d. the command fails because the name `dog` does not exist
  - e. the command fails because the name `foo/./dog` does not exist
4. If I am in my home directory named `/home/idallen` and `empty` is an empty sub-directory, what is true after this command line:  
`touch ./pig ; mv ./empty/../pig ../idallen/cow`
  - † a. the directory `empty/..` now has a file named `cow` in it
  - b. the directory `empty` now contains only a file named `cow`
  - c. there is a second copy of the file `pig` in the file named `cow`
  - d. the command fails because path `./empty/./pig` does not exist
  - e. the command fails because path `../idallen/cow` does not exist

5. If directory `/dir` contains these three four-character file names: `.123`, `.124`, `.???`, then what is the output of the following command line:  
`echo /dir/????`
  - † a. `/dir/????`
  - b. `/dir/.123 /dir/.124 /dir/.???`
  - c. `/dir/.123 /dir/.124`
  - d. `echo: /dir/????: No such file or directory`
  - e. no output on screen
6. 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/?1`
  - c. `rm \dir/1*\`
  - d. `rm dir/*1`
  - e. `rm dir/??`
7. If file `foo` contains ninety-nine lines, each of which is the two-digit line number of the line in the file (`01` through `99`), what is the output of this command:  
`sort -r foo foo | tail -4 | head -1`
  - † a. `02`
  - b. `98`
  - c. `96`
  - d. `96 96`
  - e. `04 04`
8. If file `nine` 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:  
`sort nine nine nine | tail -3 | head -1`
  - † a. `9`
  - b. `8`
  - c. `7`
  - d. `7 7 7`
  - e. `1 1 1`
9. If file `nine` 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 on your screen of this command:  
`cat nine nine | sort -r | head -4 | tail -1`
  - † a. `8`
  - b. `7`
  - c. `6`
  - d. `5`
  - e. `9`

10. If file **nine** 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 on your screen of this command:  
`cat nine nine | tail -5 | head -1`
- † a. 5  
 b. 6  
 c. 7  
 d. 8  
 e. 9
11. If my current directory is **/home**, which of these pathnames is equivalent to the pathname **/home/a/b/c**?  
 † a. `../home/a/b/c`  
 b. `/a/b/c`  
 c. `./home/a/b/c`  
 d. `../a/b/c`  
 e. `../home/b/c`
12. If my current working directory is **/home**, and my home directory is **/home/me**, which of the following commands copies the Unix password file into my home directory under the name **foo**?  
 † a. `cp ../etc/passwd ./me/foo`  
 b. `cp ./me/../etc/passwd ../home/me/foo`  
 c. `cp ../etc/passwd ../me/foo`  
 d. `cp ../home/me/../etc/passwd ./me../foo`  
 e. `cp ../../etc/passwd /me/foo`
13. If you type the command **cat** which of the following key sequences will send an EOF and take you immediately back to the command prompt?  
 † a. `[CTRL-D]`  
 b. `[CTRL-C]`  
 c. `[CTRL-L]`  
 d. `[CTRL-U]`  
 e. `[CTRL-R]`
14. If you type the command `echo 'missing quote` which of the following key sequences will interrupt it and take you immediately back to the command prompt?  
 † a. `[CTRL-C]`  
 b. `[CTRL-D]`  
 c. `[CTRL-L]`  
 d. `[CTRL-U]`  
 e. `[CTRL-R]`

15. In an empty directory, how many arguments are passed to the **cat** command in this **bash** command line? `touch a1 a2 ac ba .a ; cat a*`  
 † a. 3  
 b. 2  
 c. 1  
 d. 4  
 e. none
16. In an empty directory, how many arguments are passed to the **rm** command in this **bash** command line? `touch 2a a a1 ba ca .a ; rm a*`  
 † a. 2  
 b. 3  
 c. 1  
 d. 4  
 e. none
17. In the output of the command `ls -a`, a dot that begins a name signifies what?  
 † a. A name that is hidden.  
 b. The parent directory.  
 c. The current directory.  
 d. A current file.  
 e. A name with an unprintable character.
18. In the output of the command `ls -a`, the one-character name `.` signifies what?  
 † a. The current directory.  
 b. A name that is hidden.  
 c. The parent directory.  
 d. A current file.  
 e. A name with an unprintable character.
19. What appears on your screen after this bash shell command line?  
`echo cow | wc`  
 † a. 1 1 4  
 b. 1 1 3  
 c. 1 3 1  
 d. 3 1 1  
 e. no output on screen
20. Which Unix command sequence deletes a directory and everything inside it?  
 † a. `rm -r dir`  
 b. `erase dir`  
 c. `erase -r dir`  
 d. `rmdir -all dir`  
 e. `deltree dir`

21. What command shows all the lines in file **foo** that contain the string **group**?
- `grep group foo`
  - `grep foo group`
  - `cat foo ; grep group`
  - `cat foo | wc group`
  - `grep group ; foo`
22. What is the output of this shell command line in an empty directory? `cat *`
- an error message from `cat` saying `*` does not exist
  - no output on screen
  - `*`
  - `. ..`
  - `.`
23. What is the output of this shell command line in an empty directory? `echo *`
- `*`
  - an error message from `echo` saying `*` does not exist
  - no output on screen
  - `. ..`
  - `.`
24. What is the output of this shell command line in an empty directory?
- ```
touch x .a .ab .cde .fghi ; echo .??*
```
- `.ab .cde .fghi`
  - `.cde .fghi`
  - `.??*`
  - `. .. .a .ab .cde .fghi`
  - an error message from `echo` saying `.??*` does not exist
25. What is the output of this shell command line in an empty directory?
- ```
touch .1 .2 .3 4 5 6 ; echo .*
```
- `. .. .1 .2 .3`
  - `.1 .2 .3 4 5 6`
  - `4 5 6`
  - `.*`
  - an error message from `echo` saying `.*` does not exist
26. What is the output of this shell command line in an empty directory?
- ```
touch 1 2 3 .a .b .c ; echo .??*
```
- `.??*`
  - `. .. .a .b .c`
  - `. .. 1 2 3 .a .b .c`
  - `.a .b .c`
  - an error message from `echo` saying `.??*` does not exist

27. What is the output of this shell command line in an empty directory?
- ```
touch 1 2 3 .a .ab .abc ; echo [.]*
```
- `[.]*`
  - `. .. .a .ab .abc`
  - `.a .ab .abc`
  - no output
  - an error message from `echo` saying `[.]*` does not exist
28. What is the output of this shell command line in an empty directory?
- ```
touch 1 .1 23 .23 456 ; echo [12]*
```
- `1 23`
  - `1 .1 23 .23`
  - `1 .1 23 .23 456`
  - `[12]*`
  - an error message from `echo` saying `[ab]*` does not exist
29. What is the output of this successful command sequence?
- ```
cd /tmp ; mkdir one ; mkdir two ; pwd
```
- `/tmp`
  - `/tmp/one`
  - `/tmp/two`
  - `/tmp/one/two`
  - `/two`
30. What would you see if you typed this command: `cat /foo`
- The contents of the file **foo** located in the root directory
  - The contents of the file **foo** located in your home directory
  - The contents of the file **foo** located in the parent directory
  - The contents of your subdirectory named **foo**
  - The contents of your directory named **foo**
31. When doing an `ls -a`, the output pathname that is a double dot (`..`) signifies what?
- The parent directory.
  - A file or directory with double links.
  - A hidden file.
  - The current directory.
  - The root directory.
32. Which command below is the best way to find a line containing an asterisk (\*) in the file named **foo**?
- `grep '*' foo`
  - `grep * foo`
  - `grep foo ""`
  - `grep foo [*]`
  - `grep ./* foo`

33. Which command below removes *only* this five-character file name containing a special character (and no others): **\*test**
- † a. **rm "\*\*test"**
  - b. **rm -r ./ \*test**
  - c. **rm ../ \*test**
  - d. **rm / \*test**
  - e. **rm -r \*test**
34. Which command line below does not show any lines from inside the file **out**?
- † a. **slocate out**
  - b. **head -9 out**
  - c. **tail -3 out**
  - d. **sort -n out**
  - e. **cat out**
35. Which command line displays the contents of the Unix **passwd** file one page at a time?
- † a. **less /etc/passwd**
  - b. **less | /etc/passwd**
  - c. **cat less | /etc/passwd**
  - d. **/etc/passwd | less**
  - e. **cat /etc/passwd | vim**
36. Which command line lists all possible utilities available for editing files?
- † a. **man -k edit**
  - b. **man edit**
  - c. **finger edit**
  - d. **which edit**
  - e. **whereis edit**
37. Which command line takes the list of users currently logged in to the machine and displays just the one line containing the userid that sorts first in the alphabet?
- † a. **w | sort -r | tail -1**
  - b. **sort | w | head -1**
  - c. **sort who | tail -1**
  - d. **head -1 sort who**
  - e. **head -1 who | sort**
38. Which command pipeline outputs the count of the number of userids containing the string **allen** that are currently logged in?
- † a. **w | grep allen | wc**
  - b. **w | wc allen | grep**
  - c. **who | grep | allen wc**
  - d. **who | wc allen**
  - e. **who | wc | grep allen**

39. Which command pipeline outputs the count of the number of pathnames (including all subdirectories) that lie under the current directory?
- † a. **find . | wc**
  - b. **file . | wc**
  - c. **ls / | wc**
  - d. **ls . | wc**
  - e. **dir / | wc**
40. Which command sequence below outputs only lines 5-10 of the file named **foo**?
- † a. **head -10 foo | tail -6**
  - b. **head -15 foo | tail -5**
  - c. **head -5 foo | tail -10**
  - d. **tail -10 foo | head -6**
  - e. **tail -15 foo | head -5**
41. Which of the following shell command lines displays **all** the names in the current directory that are exactly three digits long (and no others)?
- † a. **echo [0-9][0-9][0-9]**
  - b. **echo [1-3][1-3][1-3]**
  - c. **echo [3][3][3]**
  - d. **echo [?][?][?]**
  - e. **echo ???**
42. Which of the following shell command lines displays only the names in the current directory that are exactly three alphabetic letters long?
- † a. **echo [a-zA-Z][a-zA-Z][a-zA-Z]**
  - b. **echo [0-3][0-3][0-3]**
  - c. **echo [?][?][?]**
  - d. **echo ???**
  - e. **echo \*\*\***
43. Which of these statements is true?
- † a. If **/x** is an empty directory, **sort /x/\*** produces an error message.
  - b. If **/y** is an empty directory, **echo /y/\*** produces an error message.
  - c. Only single quotes are strong enough to stop shell glob (wildcard) patterns from expanding.
  - d. Only double quotes are strong enough to stop shell glob (wildcard) patterns from expanding.
  - e. Only backslashes are strong enough to stop shell glob (wildcard) patterns from expanding.
44. Which of these statements is true?
- † a. To indicate End-of-File (no more input) to a program, type **[CONTROL]-[D]**.
  - b. To erase an entire line of typing, type **[ALT]-[DELETE]**.
  - c. To interrupt a Unix process from the keyboard, type **[CONTROL]-[D]**.
  - d. Command **apropos** is an exact synonym for command **man**.
  - e. The **file** command creates a new, empty file in the current directory

**Answer Key - DAT 2330 – Ian Allen – Winter 2006 - DAT 2330 Unix Test  
#1 - 10%**

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

Macro .cmd split with indent: 14