Evaluation: 42 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 - 42 Questions - 10 of 20%

(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)

1. Which command sequence correctly searches for **`str`** and then prints **`OK`** if it is found inside the file **`foo`**?
   † **a**. **`if grep <foo str ; then echo OK ; fi`**
   b. **`if [ grep str foo ] ; then echo OK ; fi`**
   c. **`if test str foo ; then echo OK ; fi`**
   d. **`if test str = foo ; then echo OK ; fi`**
   e. **`if [ test str foo ] ; then echo OK ; fi`**

2. Which command sequence correctly compares the numbers and prints **`OK`**?
   † **a**. **`if [ 1 -lt 2 ] ; then echo OK ; fi`**
   b. **`if [ 2 > 1 ] ; then echo OK ; fi`**
   c. **`if [ ! 2 < 1 ] ; then echo OK ; fi`**
   d. **`if ( let 2 > 1 ) ; then echo OK ; fi`**
   e. **`if ( 1 let 2 ) ; then echo OK ; fi`**

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

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

5. In an empty directory, what is the output on your screen of these commands:
   `touch uu .u uv .v uw ; a="*u *v" ; echo "$a"`
   † **a**. **`*u *v`**
   b. **`u* v*`**
   c. **`uu uv`**
   d. **`uu .u uv .v`**
   e. **`$a`**

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 = dog ] ; echo $?`**
   † **a**. **`0`**
   b. **`1`**
   c. the number 1 or 0 followed by another 1 or 0 on a new line
   d. **`test: $a: integer expression expected`**
   e. no output

7. 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. **`1`**
   c. the number 1 or 0 followed by another 1 or 0 on a new line
   d. **`test: $a: integer expression expected`**
   e. no output

8. If **`x=one`** and **`y=two`** then what is the output on your screen of the following sequence of **`bash`** commands: **`if $x = $y ; then echo $a ; fi`**
   † **a**. **`bash: one: command not found`**
   b. **`test: one: integer expression expected`**
   c. **`test: $x: integer expression expected`**
   d. **`one`**
   e. no output

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

10. 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. **`0`**
    c. the number 1 or 0 followed by another 1 or 0 on a new line
    d. **`test: $x: integer expression expected`**
    e. no output

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

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

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

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

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

16. What is the output on your screen of the following sequence of commands:
```
i=00 ; [ $i = 0 ] ; echo $?
```
† **a**. **1**
b. **0**
c. the number 0 or 1 followed by another 0 or 1 on a new line
d. **test: $i: integer expression expected**
e. no output

17. What is the output on your screen of the following sequence of commands:
```
x=ok ; y=ok ; [ x = y ]
```
† **a**. no output
b. **1**
c. **0**
d. **bash: x: command not found**
e. **test: x: integer expression expected**

18. What is the output on your screen of this command sequence:
```
echo pig >one ; echo bat | tail one
```
† **a**. **pig**
b. **bat**
c. **pig** followed by **bat**
d. **bat** followed by **pig**
e. an error message

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

20. 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**. **c d e**
b. **b**
c. **b"**
d. **$2**
e. **a b**

21. What is the output on your screen of this two-command sequence:
```
cd /etc || echo "cd $(pwd)"
```
† **a**. no output
b. **cd /etc**
c. **cd 0pwd)**
d. **cd $(pwd)**
e. **/etc**

22. What is the output on your screen of this two-command sequence:
```
cd /bin/ls && echo "in $(pwd)"
```
† **a**. `bash: cd: /bin/ls: Not a directory`
b. `in /etc`
c. `in 0pwd)`
d. `in $(pwd)`
e. no output

23. What is the output on your screen of this two-command sequence:
```
cd /bin && echo "cd $(pwd)"
```
† **a**. `cd /bin`
b. `cd 0pwd)`
c. `cd $(pwd)`
d. `/bin`
e. no output

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

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

26. 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 2 '$3' 1`
b. `./foo $1 '$2' $3`
c. `./foo '$1' "$3" $2`
d. `./foo $1 $2 $3`
e. `./foo $3 "$2" $1`

27. What is the output on your screen of this sequence of three shell commands:
```
umask 457 ; mkdir dir ; ls -ld dir
```
† **a**. `d-wx-w---- 2 me me 128 Jan 9 9:34 dir`
b. `d-w--w---- 2 me me 128 Jan 9 9:34 dir`
c. `d-wx-w-rwx 2 me me 128 Jan 9 9:34 dir`
d. `dr--r-xrwx 2 me me 128 Jan 9 9:34 dir`
e. `dr-xr-xrwx 2 me me 128 Jan 9 9:34 dir`

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

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

30. What is the output on your screen of this command sequence:
```
true && echo Linux     Rocks $?
```
† **a**. `Linux Rocks 0`
b. `Linux Rocks ?`
c. `Linux     Rocks ?`
d. `Linux     Rocks 1`
e. no output

31. What is the output on your screen of this command sequence:
```
false && echo "linux     rules $?"
```
† **a**. no output
b. `linux rules 1`
c. `linux     rules 0`
d. `linux     rules 1`
e. `linux rules 0`

32. Given the following command line: `read one two three` which user keyboard input line below will assign the text `bb` to the shell variable named `two`?
† **a**. `aa bb cc`
b. `one=aa two=bb three=cc`
c. `aa,bb,cc`
d. `aa:bb:cc`
e. `aa;bb;cc`

33. 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**. **70**
b. **50**
c. **80**
d. **120**
e. no lines (empty file)

34. If directory **/dir** contains these three four-character file names: **.123**, **.124**, **.???**, then what is the output on your screen 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

35. 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/\??**
b. **rm dir/?1**
c. **rm dir/1***
d. **rm dir/*1**
e. **rm dir/??**

36. Which of these commands makes a file owned by me, also executable by me?

† **a**. **chmod u+x ./myfile**
b. **chmod x+u myfile**
c. **chmod x=u ./myfile**
d. **umask 777 myfile**
e. **umask 111 myfile**

37. 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 "... "**

38. 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=$(wc /etc/passwd | awk echo $1)**
d. **foo=$(wc -l /etc/passwd | awk "print $1")**
e. **foo=$(awk -F: /etc/passwd | wc -l)**

39. In an empty directory, what is the output on your screen of this three-command sequence:

```
touch za .z zb .w .1 ; f=".z* .w*" ; echo '$f'
```

† **a**. **$f**
b. **.z* .w***
c. **.z .w**
d. **za .z zb .w**
e. **za zb .w***

40. 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 $3"**
e. **cat "$? $? $?"**

41. What is the output on your screen of this sequence of three shell commands:

```
echo ls >fil ; >fil ls fil ; wc fil
```

† **a**. **1 1 4 fil**
b. **1 1 3 fil**
c. **1 1 2 fil**
d. **0 0 0 fil**
e. no output

42. Select the correct **bash** shell order of command line processing:

† **a**. aliases, redirection, variables, GLOBs
b. aliases, variables, redirection, GLOBs
c. aliases, variables, GLOBs, redirection
d. aliases, GLOBs, variables, redirection
e. redirection, aliases, GLOBs, variables

Answer Key - DAT 2330 – Ian Allen – Fall 2004 - DAT 2330 Unix Test #2 - 20%

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

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

```
Count of a:  42  100%

With 5 choices: 42
  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

Macro .cmd splits: 24
Macro .ans splits: 0
```