Evaluation: 40 Questions **Name:** _____

Important Instructions

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

One Mark Correct Answers are worth one mark each.

Zero Marks Unanswered questions are worth zero marks.

Minus 0.25 Marks Incorrect Answers are worth −0.25 (minus ¼) mark each. Incorrect guesses will be partially subtracted from your score.

(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)

1. If **`bar`** is an executable script containing the line **`foo=dog`** then what is the **`bash`** output of this sequence of three commands:
```
foo=cat ; ./bar ; echo "the '$foo' ate"
```
† **a**. **`the 'cat' ate`**
b. **`the 'dog' ate`**
c. **`the '$foo' ate`**
d. **`the $foo ate`**
e. **`the 'foo' ate`**

2. What is the output of the following sequence of **`bash`** commands:
```
cd /etc && echo "in $(pwd)"
```
† **a**. **`in /etc`**
b. no output
c. **`in 0pwd)`**
d. **`in $(pwd)`**
e. **`bash: cd: /etc: No such file or directory`**

3. In an empty directory, how many files will be created using the following **`bash`** shell two-command sequence:
```
x="one 'two two'two three four" ; touch $x
```
† **a**. 5 files
b. 4 files
c. 1 file
d. 2 files
e. 3 files

4. What is the output of the following sequence of **`bash`** commands:
```
false && echo "foo     bar $?"
```
† **a**. no output
b. **`foo bar 1`**
c. **`foo     bar 0`**
d. **`foo     bar 1`**
e. **`foo bar 0`**

5. 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. **`2`**
c. **`3`**
d. **`4`**
e. **`6`**

6. A shell script named **`foo`** is executed as follows:
```
./foo 1  "2  2  2"  '3  '
```
Inside the script is the line: **`head $@`**
How many arguments are passed to the **`head`** command inside the script?
† **a**. **`5`**
b. **`6`**
c. **`4`**
d. **`3`**
e. **`2`**

7. What is the output of the following sequence of **`bash`** commands:
```
echo hi >wc ; wc wc >hi ; cat hi
```
† **a**. **`1 1 3 wc`**
b. **`0 0 0 wc`**
c. **`1 1 2 wc`**
d. no output
e. **`hi`**

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

9. 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`**

10. 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`**

11. Which of these commands always adds two to the contents of variable **`x`**?
† **a**. **`let x=x+2`**
b. **`set x=x+2`**
c. **`x=x+2`**
d. **`export x=x+2`**
e. **`[ x = $x+2 ]`**

12. What is the output of the following sequence of **`bash`** commands:
`set -- Jake Nicky Scott ; echo "name ’$2’ rocks"`
† **a**. **`name ’Nicky’ rocks`**
b. **`name ’Jake’ rocks`**
c. **`name ’Scott’ rocks`**
d. **`name ’$2’ rocks`**
e. **`name ’’ rocks`**

13. What is the output of the following sequence of **`bash`** commands:
`wc=’one two’ ; test wc = wc`
† **a**. no output
b. **`1 2 8 wc`**
c. **`1`**
d. **`0`**
e. **`test: too many arguments`**

14. If **`a=1`** and **`b=2`** then which of the following **`bash`** command lines outputs only the word **`hi`** (and nothing else)?
† **a**. **`[ a = a ] && echo hi`**
b. **`[ a -ne b ] && echo hi`**
c. **`[!a = b] && echo hi`**
d. **`[a -ne b] || echo hi`**
e. **`[a!=a] || echo hi`**

15. What is the output of the following sequence of **`bash`** commands:
`a=1 ; b=2 ; test $a -ge $b ; echo $?`
† **a**. **`1`**
b. **`0`**
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

16. What is the output of the following sequence of **`bash`** commands:
`a=cow ; b=dog ; test -z $a ; echo $?`
† **a**. **`1`**
b. **`0`**
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

17. If **`a=cow`** and **`b=dog`** then what is the output of the following sequence of **`bash`** commands: **`[ $a = dog -o $b = cow ] ; echo $?`**
† **a**. **`1`**
b. **`0`**
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

18. If **`a=cow`** and **`b=dog`** then what is the output of the following sequence of **`bash`** commands: **`[ $a = cow -a $b = cow ] ; echo $?`**
† **a**. **`1`**
b. **`0`**
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

19. If **`a=cow`** and **`b=dog`** then what is the output of the following sequence of **`bash`** 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

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

21. If a **`bash`** 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 ’$1’`**
b. **`./foo bar ’bar’`**
c. **`./foo ’bar’ "bar"`**
d. **`./foo $1 $1`**
e. **`./foo 1 "$1"`**

22. In a directory containing only the file named **$?** (two characters), what is the output of the following sequence of **bash** commands:
```
test -d /etc/passwd ; let x=$?+$? ; echo $x
```
† **a**. **2**
b. **1**
c. **0**
d. **$?**
e. **$?+$?**

23. Which **bash** command sequence correctly compares the two numbers and prints **OK**?
† **a**. **if [4 -gt 3] ; then echo OK ; fi**
b. **if [4 > 3] ; then echo OK ; fi**
c. **if [! 4 <= 3] ; then echo OK ; fi**
d. **if (let 4 > 3) ; then echo OK ; fi**
e. **if (! 4 < 3) ; then echo OK ; fi**

24. If **x=5** and **y=5**, which **bash** command sequence correctly compares the two numbers as equal and prints **OK**?
† **a**. **if test $x -eq $y ; then echo OK ; fi**
b. **if test x -eq y ; then echo OK ; fi**
c. **if [x = y] ; then echo OK ; fi**
d. **if (x == y) ; then echo OK ; fi**
e. **if [$x==$y] ; then echo OK ; fi**

25. Which **bash** command sequence correctly searches for the **string** and then prints **OK** if it is found inside the password file?
† **a**. **if grep string /etc/passwd ; then echo OK ; fi**
b. **if [grep string /etc/passwd] ; then echo OK ; fi**
c. **if test string /etc/passwd ; then echo OK ; fi**
d. **if test string = /etc/passwd ; then echo OK ; fi**
e. **if [test string /etc/passwd] ; then echo OK ; fi**

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

27. Which **bash** command sequence below always outputs just the word **OK** only if the first argument is either a file or a directory?
† **a**. **if [-f "$1" -o -d "$1"]; then echo OK;fi**
b. **if ["-f $1" || "-d $1"]; then echo OK;fi**
c. **if ["$1" -eq -f -o "$1" -eq -d]; then echo OK;fi**
d. **if [-f -o -d "$1"]; then echo OK;fi**
e. **if [-f || -d "$1"]; then echo OK;fi**

28. Which statement is true about the the tokens (words) that lie between the keyword **case** and the keyword **in** in the **bash case** statement syntax?
† **a**. The single token can be any one word or string.
b. The single token must be one integer.
c. The single token must be one integer variable.
d. Multiple tokens are allowed; all must all be integer variables.
e. Multiple tokens are allowed; they can be any number of words or strings.

29. If **colour=blue** then which one of the following **case** patterns will match this statement: **case "$colour" in**
† **a**. **[Bb]l??) echo match ;;**
b. **"blu?") echo match ;;**
c. **(??ue echo match ;;**
d. **[blue] | [BLUE]) echo match ;;**
e. **b | l | u | e) echo match ;;**

30. If **colour=green** then which one of the following **case** patterns will match this statement: **case "$colour" in**
† **a**. ***) echo match ;;**
b. **gre?) echo match ;;**
c. **"gree?") echo match ;;**
d. **(*een echo match ;;**
e. **[green] | [GREEN]) echo match ;;**

31. What is the value of variable **var** at the end of the loop that starts:
```
for var in 2 1 $# $? 4 3 ; do
```
† **a**. **3**
b. **4**
c. **6**
d. **2**
e. the value is undefined

32. If a script named **foo** contains a loop that starts:
```
for var in $* ; do
```
and the script is executed using this command line:
```
./foo 1 ' 2 2 2 ' 3 4 " 5 5 " 6
```
how many times will the loop iterate?
† **a**. 9 iterations
b. 8 iterations
c. 7 iterations
d. 6 iterations
e. 13 iterations

33. Starting with **x=1**, what is the value of **x** after this loop finishes?

```
while [ $x -le 10 ]; do let x="x*2"; done
```

† **a**. **16**
b. **8**
c. **x*2**
d. **20**
e. **12**

34. Which of the following statements is equivalent to this one:

```
while [ $x -lt 99 ] ; do
```

† **a**. **until test $x -ge 99 ; do**
b. **until [$x -gt 99] ; do**
c. **until [$x > 99] ; do**
d. **until test $x >= 98 ; do**
e. **until ! $x <= 98 ; do**

35. What is the output of the following sequence of **bash** commands:

```
set -- "a b" "c d" "e f" ; shift ; echo $1
```

† **a**. **c d**
b. **b**
c. **a**
d. **a b**
e. bash: $1: unbound variable

36. If **foo** is a script containing the line **TERM=new ; export TERM**, what is the output of the following sequence of **bash** commands that use **foo**:

```
TERM=bar ; ./foo ; echo $TERM
```

† **a**. **bar**
b. **new**
c. **foo**
d. **TERM**
e. **$TERM**

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 of the following lines indicates the start of a shell *here* document (e.g. for a menu)?

† **a**. **cat <<- XXX**
b. **cat <- ENDIT**
c. **cat >>- ENDIT**
d. **cat >- XXX**
e. **>>- cat ENDIT**

39. Which of the following outputs does not end in a newline character?

† **a**. **echo -n "foo"**
b. **echo 1&<2 'foo'**
c. **echo 1<$2 foo**
d. **echo -e 'foo\n'**
e. **echo --newline "foo"**

40. 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 | echo $1]**
d. **foo=[cat -l /etc/passwd]**
e. **foo=[grep -c /etc/passwd]**

Answer Key - CST 8129 − Ian Allen − Fall 2002 - CST 8129 Test #2 - Unix - 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

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

Count of letter a: 40 100%

Questions with 5 choices: 40
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

Macro .cmd splits: 19
Macro .ans splits: 0