Quiz: Shell

Total Score: /12

Printed Name:

Rules:

1. You MAY use any printed or handwritten notes.

2. You MAY NOT use a computer or any other electronic device.

3. You MAY NOT discuss this quiz with any other human.

Instructions:

1. Attached is a POSIX-compliant shell script. Some of these commands have output, and some of them do not. There are six lines that end in the command | wc -l, and your job is to write the output from those lines.

2. There are 6 problems. Each correct problem is worth 2 points, each blank problem is worth 0 points, **each incorrectly answered problem is worth -1 point**.

Problem formation rules: There are several practice shell scripts in the files example?.sh. They all follow the same format that your quiz will follow. I will use the following transformations to generate new shell scripts for the quiz based on these practice scripts.

1. Add/remove git add, git commit and git checkout commands

2. Change file/branch names

3. Change redirection and pipe symbols

4. Add/remove single/double quotes/backticks/dollar signs

5. Add/remove the -a flag to ls

6. Add/remove wildcard characters

7. Add/remove variable references

```
1  #!/bin/sh
2  set -e
3
4  # create a temporary directory for all work to happen in
5  temp_dir=$(mktemp -d)
6  cd "$temp_dir"
7  pwd
8
9
10 echo 'problem 1'
11 echo big data > .README.md
12 echo "big data" >> .README.md
13 echo 'big data' > .README.md
14 cat .README.md | wc -l
15
16
17
18 echo 'problem 2'
19 git init > /dev/null 2> /dev/null
20 git add .README.md
21 git commit -m 'first commit' > /dev/null 2> /dev/null
22 ls -a | wc -l
23
24
25
26 echo 'problem 3'
27 git checkout -b new_branch > /dev/null 2> /dev/null
28 echo test >> .new_file
29 touch example
30 git add *
31 git commit -m 'new_branch' > /dev/null 2> /dev/null
32 git checkout master > /dev/null 2> /dev/null
33 ls -a | wc -l
34
35
36
37 echo 'problem 4'
38 mkdir dir
39 for file in a b c d; do echo 'hello world' > dir/$file; done
40 ls -a dir | wc -l
41
42
43
44 echo 'problem 5'
45 var='this is an example'
46 for file in $var; do echo $file; done | wc -l
47
48
49
50 echo 'problem 6'
51 cd dir
52 touch "*"
53 rm *
54 ls -a | wc -l
```