9.5/10

COSC 350 System Software: Mini Test #1

9/17/2021

Name: Jung An

1. (1 pt.) Create a file named **numbs** that contains the integers 1 through 100, one integer per line with shell commends with output redirection. The file will have 100 lines. You need use for loop.

```
~/numbs
for i in $(seq 1 to 100); do
 echo "$i" >> numbs
done
```

2. (1 pt.) When we pass arguments to a script, system save each arguments in positional parameters $0, $1, $2, … What information will save on each of following positional parameters

- $@ : All parameters
- $*: All parameters
- $0: first parameter ✗ Shell script name
- $#: number of parameters

-0.1

3. (0.5 pt.) What are two conditions to make a shell script file executable
   a. location of Bash
   b. modify file to be executable (chmod)

4. (0.5 pt.) There are two types of libraries: static library and shared library. Briefly explain differences between static and shared library.
   - **Static library:** compiled library.
   - **Shared library:** run-time compilation ✗ look at slide

-0.3

5. (2 pt.) Write a shell script named "sum.sh" that reads sequence of integers on the command line. Each integer are separated by a space and prints their sum to the screen.
Use for loop and **Don't use (()) (c syntax) for calculation**.

For example, when you run this script with 10 sequences of integers
./sum.sh 1 2 3 4 5 6 7 8 9 10

output of the script will be

Sum of Arguments is 55.

```
#!/bin/sh
Sum=0

for i in #@
do
  let "Sum=$Sum+$i"
done
echo "Sum of Arguments is $sum."

exit 0
```

6. (0.5 pt.) Given the following variable what will be displayed for each of the following shell commands?

vari="How are you?"

- **echo vari** vari
- **echo "$vari"** How are you?
- **echo '$vari'** How are you? $vari
- **echo \$vari** $ vari
- **echo \$$vari** $How are you?

7. (1.5 pt.) Write a script to calculating factorial of given number by **using while loop**. The script asks an integer value and calculates factorial and display the result.

```
#!/bin/sh

loop="true"

while ["$loop"="true"]
do
    echo "Please input positive integer"
    read number
    case $number in
        *[0-9]*|"") loop="false"
    esac
done
factorial=1
while [$number -ge 0]
do
    if [$number -eq 0]; then
        let "factorial=$factorial*1"
    else
        let "factorial=$factorial*$number"
    fi
    let number--
done
echo $factorial
exit 0
```

8. (1 pt.) Since a directory itself is a file in Linux system, each directory has its name. Write bash script which test each files in current working directory and display subdirectory names.

```
#!/bin/sh

for file in *
do
    if [-d $file]; then
        ls -l \W\file
    fi

done

exit 0
```

9. (1.5 pt.) Write shell script by using nested for loop to print the following patterns on screen based on an integer input n (between 1 and 9) from the keyboard. **(Do not use (()) in for loop)**. Your program display following shape with input 5.

```
1
22
333
4444
55555
```

```
#!/bin/sh

echo "Enter an integer"
read integer

for i in $(seq 1 to $integer)
do
    for j in $(seq 1 to $i)
    do
      echo "$i"
    done
done

exit 0
```

10. (0.5 pt.) Briefly explain the difference between the following two bash commands:

- **ls -l | less** this is pipe command. After ls -l, it performs less
- **ls -l > less** This send results of ls -l to file called less