Assignment 1

Part 1

Check if there is a file called test_1 in the current directory. If not, use touch to create the file. If the file exists and is not a directory, delete the file and create a directory called test_2. If the file exists and is a directory, delete the directory and create a directory called test_3.

Provide the shell commands or create a script called check 1.sh

Solution:

- osboxes@osboxes:~\$ ls
 Desktop Downloads Pictures repeat.sh sizes_1.sh t.sh
 Documents Music Public seq 1.sh Templates Videos
- osboxes@osboxes:~\$ find test_1 find: 'test_1': No such file or directory
- osboxes@osboxes:~\$ touch test 1
- osboxes@osboxes:~\$ find test_1 test_1
- osboxes@osboxes:~\$ rm test 1
- osboxes@osboxes:~\$ find test_1 find: 'test_1': No such file or directory
- osboxes@osboxes:~\$ mkdir test 2
- osboxes@osboxes:~\$ ls
 Desktop Downloads Pictures repeat.sh sizes_1.sh test_2 Videos
 Documents Music Public seq 1.sh Templates t.sh
- osboxes@osboxes:~\$ find test_2 test_2
- osboxes@osboxes:~\$ rm test_2
 rm: cannot remove 'test_2': Is a directory
- osboxes@osboxes:~\$ rmdir test 2
- osboxes@osboxes:~\$ ls
 Desktop Downloads Pictures repeat.sh sizes_1.sh t.sh
 Documents Music Public seq_1.sh Templates Videos
- osboxes@osboxes:~\$ find test_2 find: 'test_2': No such file or directory

- osboxes@osboxes:~\$ mkdir test 3
- osboxes@osboxes:~\$ ls
 Desktop Downloads Pictures repeat.sh sizes_1.sh test_3 Videos
 Documents Music Public seq 1.sh Templates t.sh
- osboxes@osboxes:~\$ find test_3 test_3

Screenshoot:

```
osboxes@osboxes:~$ ls
Desktop Downloads Pictures repeat.sh sizes_1.sh t.sh
Documents Music
                              seq 1.sh Templates
osboxes@osboxes:~$ find test 1
find: 'test 1': No such file or directory
osboxes@osboxes:~S touch test 1
osboxes@osboxes:~$ find test 1
test 1
osboxes@osboxes:~$ rm test 1
osboxes@osboxes:~$ find test 1
find: 'test 1': No such file or directory
osboxes@osboxes:~$ mkdir test 2
osboxes@osboxes:~S ls
Desktop
        Downloads Pictures repeat.sh sizes 1.sh test 2 Videos
                              seq 1.sh Templates t.sh
osboxes@osboxes:~$ find test_2
test 2
osboxes@osboxes:~$ rm test 2
rm: cannot remove 'test 2': Is a directory
osboxes@osboxes:~$ rmdir test 2
osboxes@osboxes:~$ ls
Desktop Downloads Pictures repeat.sh sizes 1.sh t.sh
Documents Music
                              seq_1.sh Templates Videos
osboxes@osboxes:~$ find test 2
find: 'test 2': No such file or directory
osboxes@osboxes:~$ mkdir test 3
osboxes@osboxes:~$ ls
Desktop Downloads Pictures repeat.sh sizes_1.sh test_3 Videos
Documents Music
                              seq_1.sh Templates t.sh
osboxes@osboxes:~$ find test 3
test 3
osboxes@osboxes:~$
```

Part 2

Write a script called sizes_1.sh which outputs one line for each file or directory in the current working directory. Each line must contain:

- 1. The size of the file, or the directory contents, in kilobytes.
- 2. The name of the file or directory.

The lines ought be ordered by their last-modified date, from oldest to youngest.

Solution:

Code for the script:

```
#!/bin/sh
IFS=""
var=$(ls -lh -tr)
echo $var
```

Screenshots of code and output:

```
osboxes@osboxes:~$ ./sizes_1.sh
total 56K
drwxr-xr-x 2 osboxes osboxes 4.0K Apr 24 16:47 Videos
drwxr-xr-x 2 osboxes osboxes 4.0K Apr 24 16:47 Templates
drwxr-xr-x 2 osboxes osboxes 4.0K Apr 24 16:47 Public
drwxr-xr-x 2 osboxes osboxes 4.0K Apr 24 16:47 Music
drwxr-xr-x 2 osboxes osboxes 4.0K Apr 24 16:47 Downloads
drwxr-xr-x 2 osboxes osboxes 4.0K Apr 24 16:47 Documents
drwxr-xr-x 3 osboxes osboxes 4.0K Oct 5 18:33 Desktop
-rwxrwxr-x 1 osboxes osboxes 60 Oct 5 19:54 t.sh
-rwxrwxr-x 1 osboxes osboxes 239 Oct 6 12:19 repeat.sh
-rwxrwxr-x 1 osboxes osboxes 340 Oct 6 12:57 seq_1.sh
-rwxrwxr-x 1 osboxes osboxes 45 Oct 6 13:00 sizes 1.sh
drwxrwxr-x 2 osboxes osboxes 4.0K Oct 7 17:23 test 3
drwxr-xr-x 2 osboxes osboxes 4.0K Oct 7 17:27 Pictures
-rw-rw-r-- 1 osboxes osboxes 1.2K Oct 7 17:35 command
osboxes@osboxes:~$
```

Part 3

Write a shell function called repeat which accepts a command and (optionally) some arguments. The script then executes the command with its arguments (if they exist) once per second.

The command sleep can be used to sleep for some number of seconds.

Solution:

Code for the script:

```
sleep 1

x=\$((x+1))

done

echo "Total Sleep $x"

}

repeat "$@"
```

Screenshots of code and Output:

```
#!/bin/sh
#arg1=$1
#arg2=$2
#IFS=""
echo "Please the number of repeation of a command<mark>"</mark>
read var
repeat()
X=0
for i in $(seq 1 $var);
do
        echo $($@)
        echo "Command no. $i\n"
        sleep 1
        x=$((x+1))
done
echo "Total Sleep $x"
repeat "$@"
"repeat.sh" 20L, 239B
                                                                 5,50
                                                                                All
```

```
osboxes@osboxes:~$ ./repeat.sh echo "Hello"
Please the number of repeation of a command
5
Hello
Command no. 1
Hello
Command no. 2
Hello
Command no. 3
Hello
Command no. 4
Hello
Command no. 5
Total Sleep 5
osboxes@osboxes:~$
```

Part 4

The command cal prints out a calendar for the current month. If you type cal 11 2020, you will get a calendar for November 2020. Create a file called seq_1.sh that takes in a number n representing a month.

When the file is run it will print out the 2020 calendar showing the months from n until the end of the year, i.e. December. If no number is supplied, it will print out the calendar for the even months of the year 2020, i.e. Feb, Apr, Jun and so on.

Solution:

Code for the scipt:

```
#!/bin/sh
n=$1
if [-z $n];
then
for i in $(seq 1 12);
```

```
do
             if [\$((i\% 2)) - eq 0];
             then
                   divide=$(cal $i 2020)
                   echo "$divide"
                   sleep 1s
             fi
      done
elif [ $n -eq 0 ];
then
      echo "Please enter a valid vlaue"
else
      for i in $(seq $n 12);
      do
             #divide=$(cal $i 2020)
             echo "$(cal $i 2020)"
             #echo "$divide"
             sleep 1s
      done
fi
```

Screenshots od code and outputs:

```
#!/bin/sh
n=$1
then
        for i in $(seq 1 12);
        do
                 if [ $((i % 2)) -eq 0 ];
                 then
                         divide=$(cal $i 2020)
                         echo "$divide'
                         sleep 1s
        done
elif [ $n -eq 0 ];
then
        echo "Please enter a valid vlaue"
else
        for i in $(seq $n 12);
                 #divide=$(cal $i 2020)
                 echo "$(cal $i 2020)"
                 #echo "$divide"
                 sleep 1s
        done
```

• Output when there is some vlaue of n:

```
osboxes@osboxes:~$ ./seq_1.sh 11
   November 2020
Su Mo Tu We Th Fr Sa
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
   December 2020
Su Mo Tu We Th Fr Sa
      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
osboxes@osboxes:~S
```

```
sboxes@osboxes:~$ ./seq_1.sh 5
      May 2020
u Mo Tu We Th Fr Sa
                         1
                             2
                       8
    4 5 6 7
                            9
0 11 12 13 14 15 16
7 18 19 20 21 22 23
4 25 26 27 28 29 30
      June 2020
u Mo Tu We Th Fr Sa
1 2 3 4 5 6
7 8 9 10 11 12 13
4 15 16 17 18 19 20
1 22 23 24 25 26 27
8 29 30
July 2020
u Mo Tu We Th Fr Sa
              1 2 3 4
             8 9 10 11
2 13 14 15 16 17 18
9 20 21 22 23 24 25
6 27 28 29 30 31
    August 2020
u Mo Tu We Th Fr Sa
   3 4 5
                   6
                             8
9 10 11 12 13 14 15
6 17 18 19 20 21 22
3 24 25 26 27 28 29
```

```
September 2020
Su Mo Tu We Th Fr
                               Sa
3 4 5
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30
      October 2020
Su Mo Tu We Th Fr Sa
                      1
                                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
     November
                     2020
Su Mo Tu We
                     Th Fr Sa
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
     December 2020
Su Mo Tu We Th Fr Sa
            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
osboxes@osboxes:~$
```

Output when there is no argument to the script

```
osboxes@osboxes:~$ ./seq_1.sh
    February 2020
Su Mo Tu We Th Fr Sa
                            1
 2
      3
         4
              5
                   б
                       7
                            8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
       April 2020
Su Mo Tu We Th Fr Sa
                   2
                       3
               1
                           4
 5
          7
              8
                   9 10 11
      б
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30
       June 2020
Su Mo Tu We Th Fr Sa
                        5
          2
               3
                   4
                            б
      8
          9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30
19 20 21 22 23 24 25
26 27 28 29 30
     June 2020
Su Mo Tu We Th Fr Sa
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
    August 2020
Su Mo Tu We Th Fr Sa
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
    October 2020
Su Mo Tu We Th Fr Sa
   1 2 3
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
   December 2020
Su Mo Tu We Th Fr Sa
1 2 3 4 5
   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
```

• Output when 0 is passed as an arrgument

```
osboxes@osboxes:~$ ./seq_1.sh 0
Please enter a valid vlaue
osboxes@osboxes:~$
```

Submitted by:

Name: Vikrant Singh Student no: 21262315