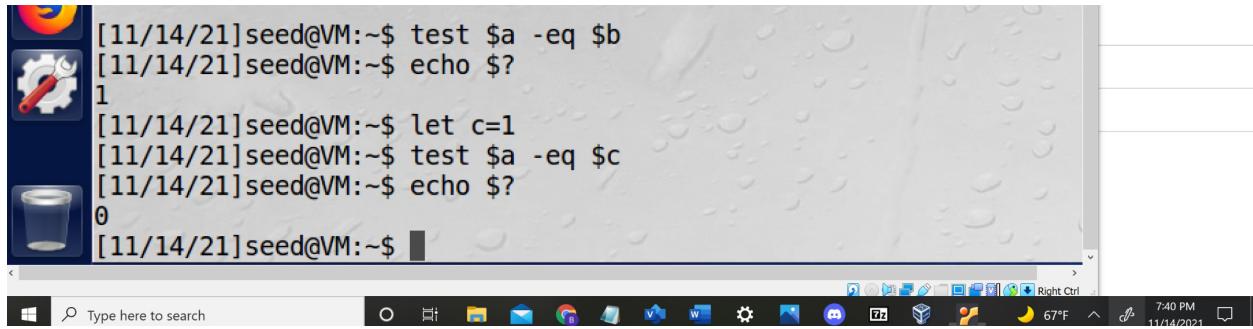


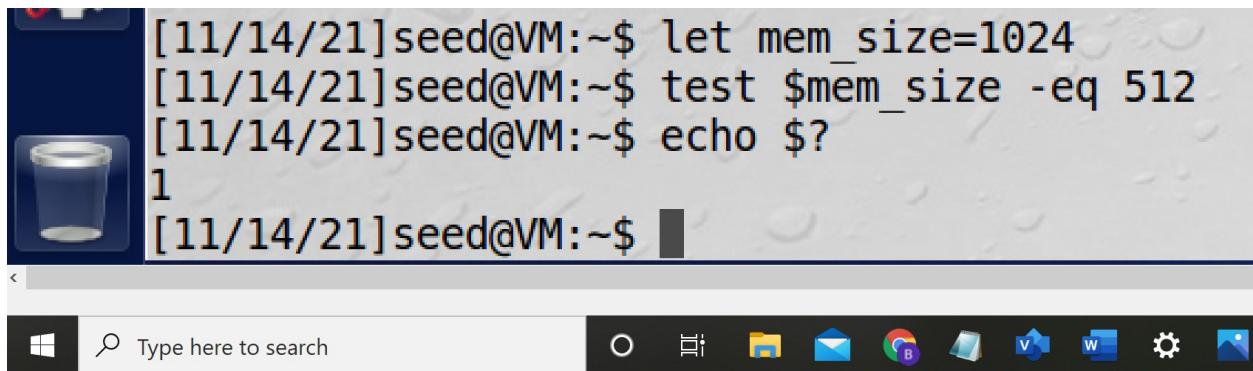
1. What are the exit statuses of the test commands discussed in this chapter and what do they mean?

The exit statuses are 0 if the test is a success it is true and 1 if the test is unsuccessful and not true.



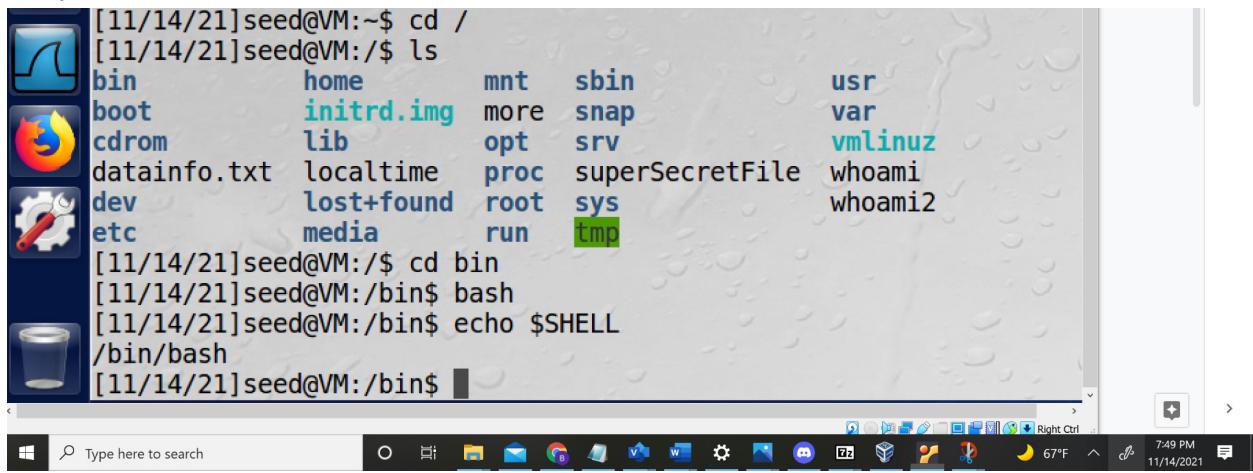
```
[11/14/21]seed@VM:~$ test $a -eq $b
[11/14/21]seed@VM:~$ echo $?
1
[11/14/21]seed@VM:~$ let c=1
[11/14/21]seed@VM:~$ test $a -eq $c
[11/14/21]seed@VM:~$ echo $?
0
[11/14/21]seed@VM:~$
```

2. Create a variable called mem_size and set its contents to 1024. Next use the test command to determine if the contents of mem_size are less than or equal to 512



```
[11/14/21]seed@VM:~$ let mem_size=1024
[11/14/21]seed@VM:~$ test ${mem_size} -eq 512
[11/14/21]seed@VM:~$ echo $?
1
[11/14/21]seed@VM:~$
```

3. Set your shell from the command line to be the Bash shell. Then use the echo command to verify the contents of the shell variable. What is now contained in the shell variable?

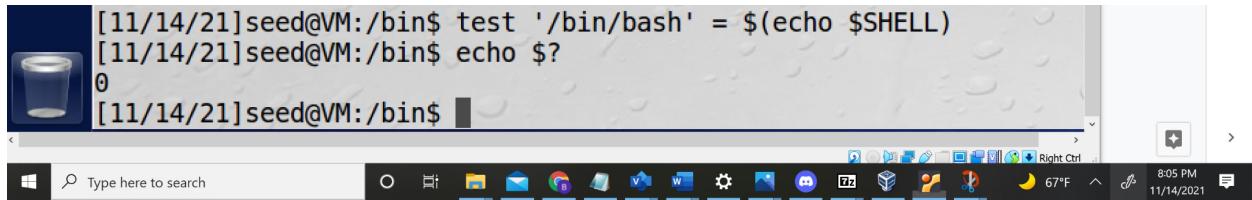


```
[11/14/21]seed@VM:~$ cd /
[11/14/21]seed@VM:/$ ls
bin          home      mnt      sbin        usr
boot         initrd.img more     snap        var
cdrom        lib       opt      srv        vmlinuz
datainfo.txt localtime proc    superSecretFile whoami
dev          lost+found root    sys        whoami2
etc          media     run     tmp
[11/14/21]seed@VM:/$ cd bin
[11/14/21]seed@VM:/bin$ bash
[11/14/21]seed@VM:/bin$ echo $SHELL
/bin/bash
[11/14/21]seed@VM:/bin$
```

/bin/bash is in the shell variable

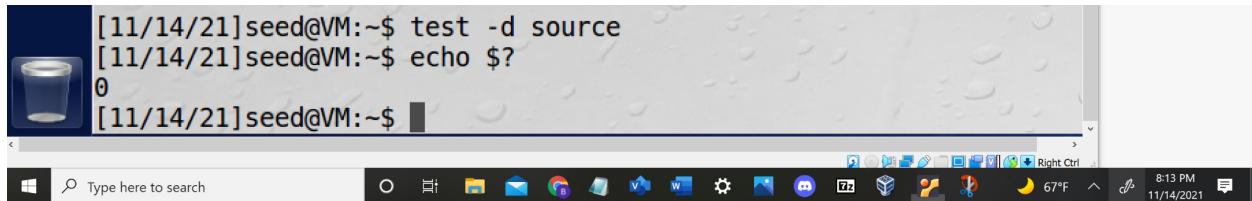
4. After performing Exercise 3, use the test command to evaluate whether the shell variable contains a reference to the Bash shell and use the echo command to determine the result.

(Note that this provides one way to verify from within a script that the script user is set up to use the Bash shell.)



```
[11/14/21]seed@VM:/bin$ test '/bin/bash' = $(echo $SHELL)
[11/14/21]seed@VM:/bin$ echo $?
0
[11/14/21]seed@VM:/bin$
```

5. Make certain that your home directory is your current working directory. Use the command to verify that your source directory exists. How might knowledge of this command be useful when you create scripts for yourself or others?



```
[11/14/21]seed@VM:~$ test -d source
[11/14/21]seed@VM:~$ echo $?
0
[11/14/21]seed@VM:~$
```

Might be good if you have a script that uses the sources folder.

6. Switch to your source directory and make a copy of the corp_phones file (which will give you a valuable backup of the corp_phones file) with the name corp_phones_bak. Using the tr command as a filter for output from another command, display to the screen the contents of corp_phones_bak so that all uppercase letters from A to M are lowercase.

7. Make your home directory your current working directory. Use the vi or Emacs editor to open the veg_choice script you created in Chapter 6. Enter a line in that script to ensure the script uses the Bash shell. Run veg_choice to ensure your change works properly.

The screenshot shows a Linux desktop environment running in Oracle VM VirtualBox. The desktop has a light gray textured background. A terminal window is open in the foreground, displaying a bash script named `veg_choice`. The script prompts the user for their favorite vegetable and checks if it's broccoli. If it is, it says "Broccoli is a healthy choice." If not, it says "Don't forget to eat your broccoli also." The terminal command `:wq` is visible at the bottom. In the background, there's a file manager window showing a list of files and a system tray with icons for battery, signal, and date/time (8:42 PM, 11/14/2021). The taskbar at the bottom includes icons for various applications like File Explorer, Mail, and Browser.

```
#!/bin/bash
echo $SHELL
echo -n "What is your favorite vegetable? "
read veg_name
if [ "$veg_name" = "broccoli" ]
then
    echo "Broccoli is a healthy choice."
else
    echo "Don't forget to eat your broccoli also."
fi
:wq
```

```
VM% veg_choice
/bin/bash
What is your favorite vegetable? banna
Don't forget to eat your broccoli also.
VM%
```

8. Edit the `veg_choice` script again, but this time change the line `if ["$veg_name" = "carrots"]` to use the `test` command.

```
lpc                                         visudo
lpinfo                                       vpddcode
lpmove                                       vsftpd
make-ssl-cert                                 xfce4-kiosk-query
mkinitramfs                                  zic
[11/15/21]seed@VM:.../sbin$ cd swap
bash: cd: swap: No such file or directory
[11/15/21]seed@VM:.../sbin$ cd ..
[11/15/21]seed@VM:/usr$ cd ..
[11/15/21]seed@VM:$ cd ~
[11/15/21]seed@VM:~$ veg_choice
What is your favorite vegetable? Banana
Don't forget to eat your broccoli also.
[11/15/21]seed@VM:~$ veg_choice
What is your favorite vegetable? broccoli
Broccoli is a healthy choice.
[11/15/21]seed@VM:~$ vi veg_choice
[11/15/21]seed@VM:~$ veg_choice
What is your favorite vegetable? Bananan
Don't forget to eat your broccoli also.
[11/15/21]seed@VM:~$ veg_choice
What is your favorite vegetable? broccoli
Broccoli is a healthy choice.
[11/15/21]seed@VM:~$
```

9. Create pseudocode and a flowchart for a proposed script that does the following:
Reads and sets the variable M
Reads and sets the variable R
Reads and sets the variable T
Sums M, R, and T in the variable A
Evaluates A to determine if it is greater than 2000
If A is greater than 2000, prints on the screen "A is over 2000."
If A is less than or equal to 2000, prints on the screen "A is 2000 or less."

Echo enter M

Read m

Echo enter r

Read r

Echo enter t

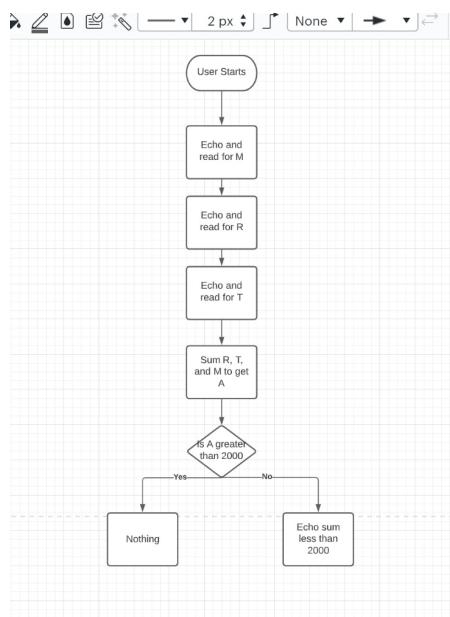
Read t

Let A= \$m +\$r+\$t

If \$A <= 2000 then

Echo A is 2000 or less.

Else



10. Create the script for Exercise 9 and name it evaluate_yourinitials, such as evaluate_jp. Test your script.

VBOX [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
echo -n "Insert M: "
read M
echo -n "Insert R: "
read R
echo -n "Insert T: "
read T
let "A = $M + $R + $T"
if test $A -le 2000;
then {
echo "A is less than 2000"
}
fi
```

"evaluate_BS" 12L, 167C 1,1 All

Windows taskbar: Type here to search, File Explorer, Mail, Google Chrome, Microsoft Edge, File Explorer, Task View, Taskbar icons, 54°F, 11:49 PM, 11/15/2021, Right Ctrl

VBOX [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

```
[11/16/21]seed@VM:~$ sh evaluate_BS
Insert M: 34
Insert R: 45
Insert T: 45
evaluate_BS: line 8: =: No such file or directory
[11/16/21]seed@VM:~$ :wq
:wq: command not found
[11/16/21]seed@VM:~$ vi evaluate_BS
[11/16/21]seed@VM:~$ sh evaluate_BS
Insert M: 345
Insert R: 23456
Insert T: 34
evaluate_BS: line 8: =: No such file or directory
[11/16/21]seed@VM:~$ vi evaluate_BS
[11/16/21]seed@VM:~$ sh evaluate_BS
Insert M: 23453
Insert R: 45
Insert T: 345
[11/16/21]seed@VM:~$ sh evaluate_BS
Insert M: 2
Insert R: 23
Insert T: 34
A is less than 2000
[11/16/21]seed@VM:~$
```

Windows taskbar: Type here to search, File Explorer, Mail, Google Chrome, Microsoft Edge, File Explorer, Task View, Taskbar icons, 54°F, 11:49 PM, 11/15/2021, Right Ctrl

11. Create a shell variable, called CALNOW, that outputs the calendar for the current month.

```
VBOX [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
[11/16/21]seed@VM:~$ CALNOW= 'cal'
      November 2021
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
[11/16/21]seed@VM:~$ echo $CALNOW
[11/16/21]seed@VM:~$ CALNOW=$(cal)
[11/16/21]seed@VM:~$ echo $CALNOW
November 2021 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 1
4 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
[11/16/21]seed@VM:~$
```

The screenshot shows a Windows desktop environment with a running Oracle VM VirtualBox window. Inside the virtual machine, a terminal window is open, displaying the following sequence of commands and their output:

- [11/16/21]seed@VM:~\$ CALNOW= 'cal'
- Output: November 2021 (calendar for November 2021)
- [11/16/21]seed@VM:~\$ echo \$CALNOW
- Output: [empty line]
- [11/16/21]seed@VM:~\$ CALNOW=\$(cal)
- [11/16/21]seed@VM:~\$ echo \$CALNOW
- Output: November 2021 Su Mo Tu We Th Fr Sa 1 2 3 4 5 6 7 8 9 10 11 12 13 1
4 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
- [11/16/21]seed@VM:~\$

The desktop taskbar at the bottom shows various pinned icons and system status information (54°F, 11:54 PM, 11/15/2021).