

TEST CASES

TEST CASE 1: Invoking “Custom Shell” from Ash Shell

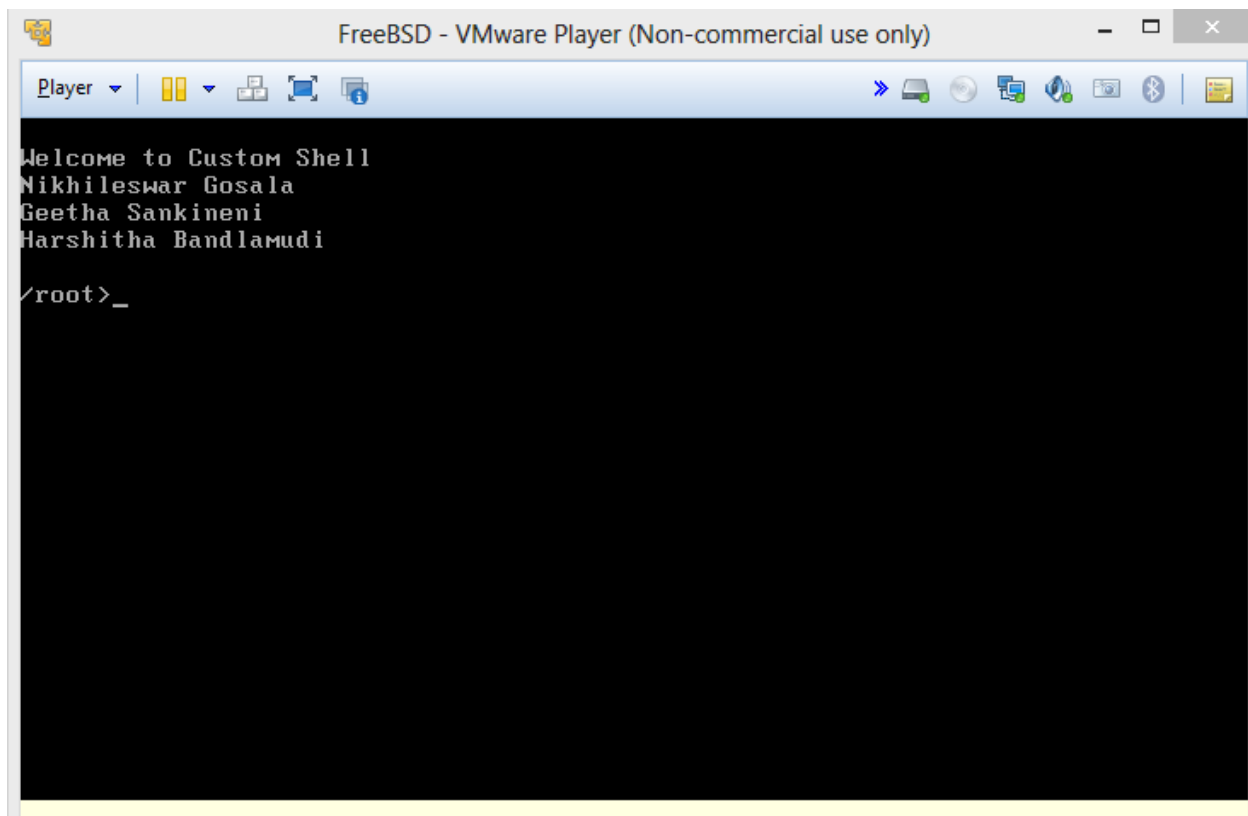
Input: Invoke the shell using “sh Minixshell.sh”

Expected Result: Custom Shell should start

Actual Result: Custom Shell should start

Result: Pass

Screen Shot:



The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal output displays a welcome message and the names of the authors, followed by a root shell prompt.

```
Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>_
```

TEST CASE 2: To read the path / home variables from the environment and print it

Input:

Case 1: echo \$PATH

Case 2: echo \$HOME

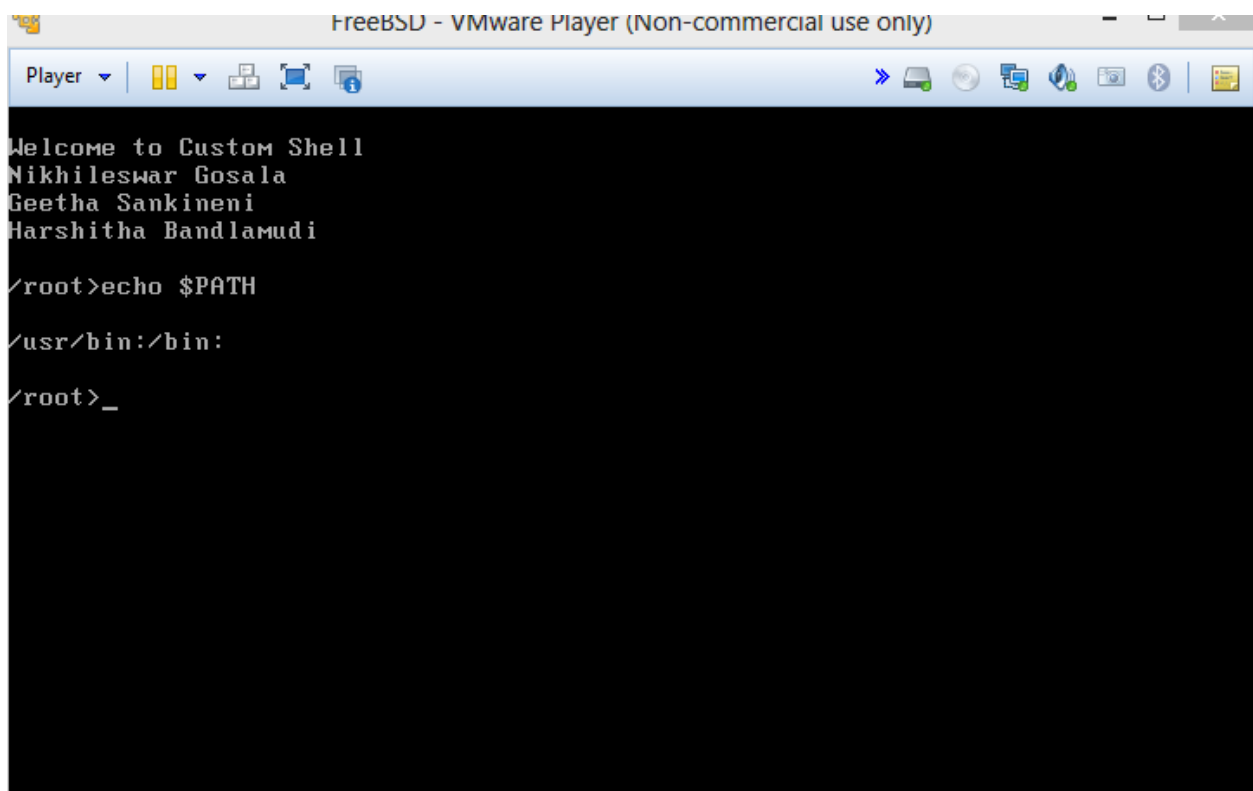
Expected Result: Displays the PATH/HOME value

Actual Result: PATH/HOME value displayed

Result: Pass

Screen Shot:

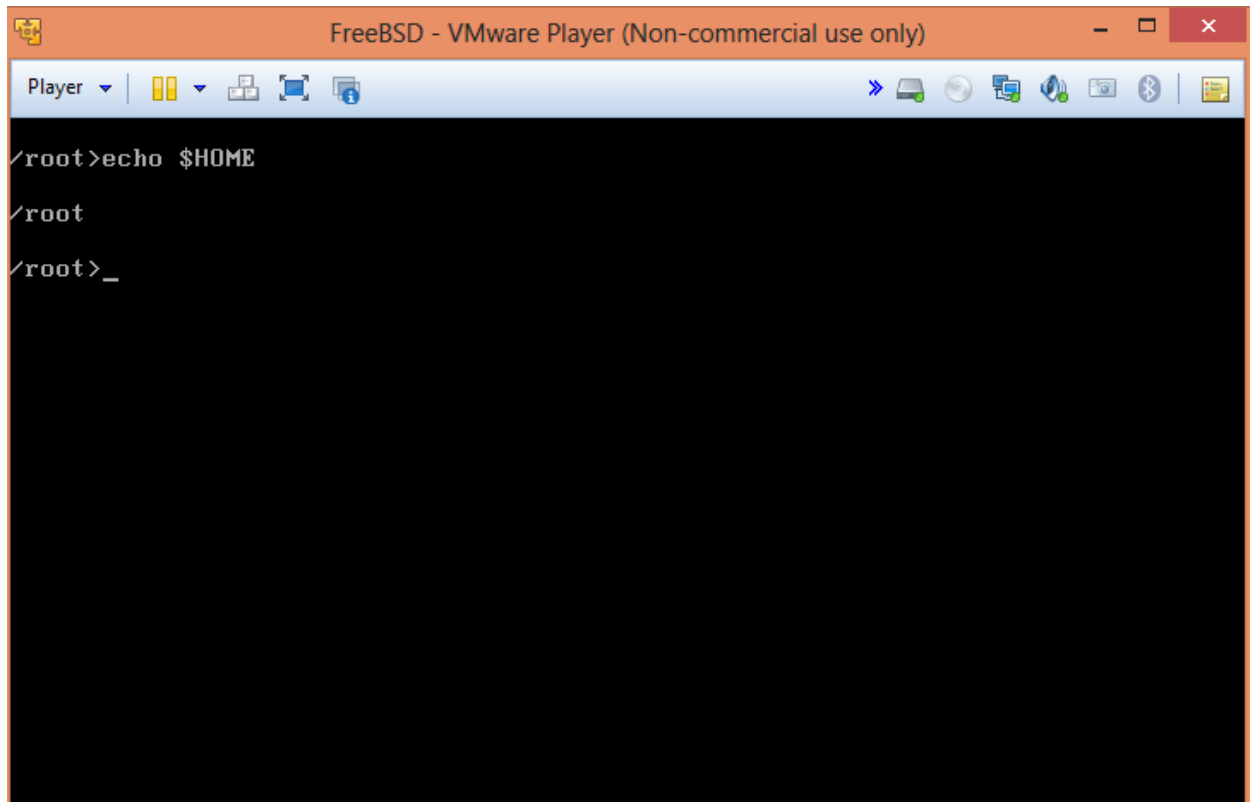
Case 1:

A screenshot of a FreeBSD VM terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal shows a "Welcome to Custom Shell" message followed by the names "Nikhileswar Gosala", "Geetha Sankineni", and "Harshitha Bandlamudi". The user then enters the command "/root>echo \$PATH", and the output is displayed as "/usr/bin:/bin:". The prompt returns to "/root>_".

```
FreeBSD - VMware Player (Non-commercial use only)
Player | [Icons] | [System Icons]
Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>echo $PATH
/usr/bin:/bin:

/root>_
```

Case 2:

The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal prompt is `/root>`. The user has entered the command `echo $HOME`, and the output is `root`. The prompt now shows `/root>_`, indicating the command has been executed and the cursor is ready for the next input.

TEST CASE3: To verify if the prompt is displaying the current directory name

Input: `cd osdi`

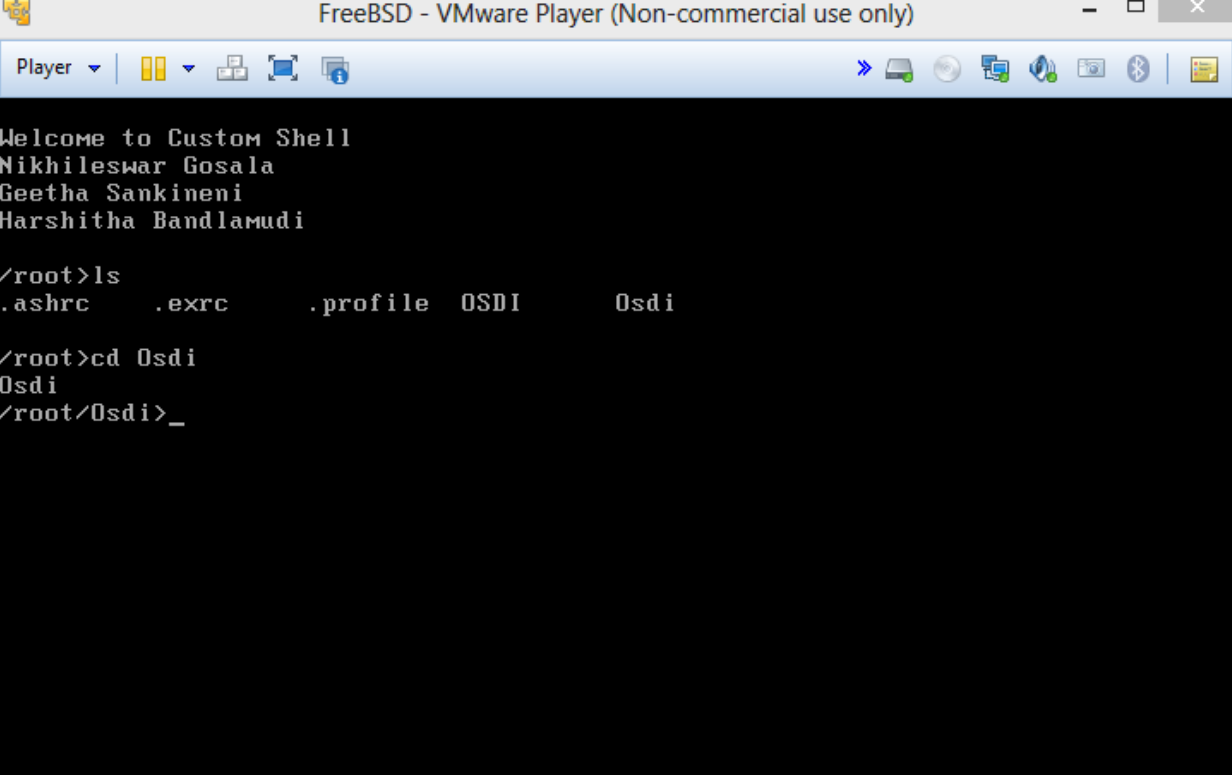
Expected Result: Prompt should display the current directory name

Actual Result: Displays the current directory name

Result: Pass

Screen Shot:

iles



The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal displays the following text:

```
Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>ls
.ashrc      .exrc      .profile  OSDI      OsdI

/root>cd OsdI
OsdI
/root/OsdI>_
```

TEST CASE 4: Customized Shell should be able to execute the utilities defined in /bin and /usr/bin

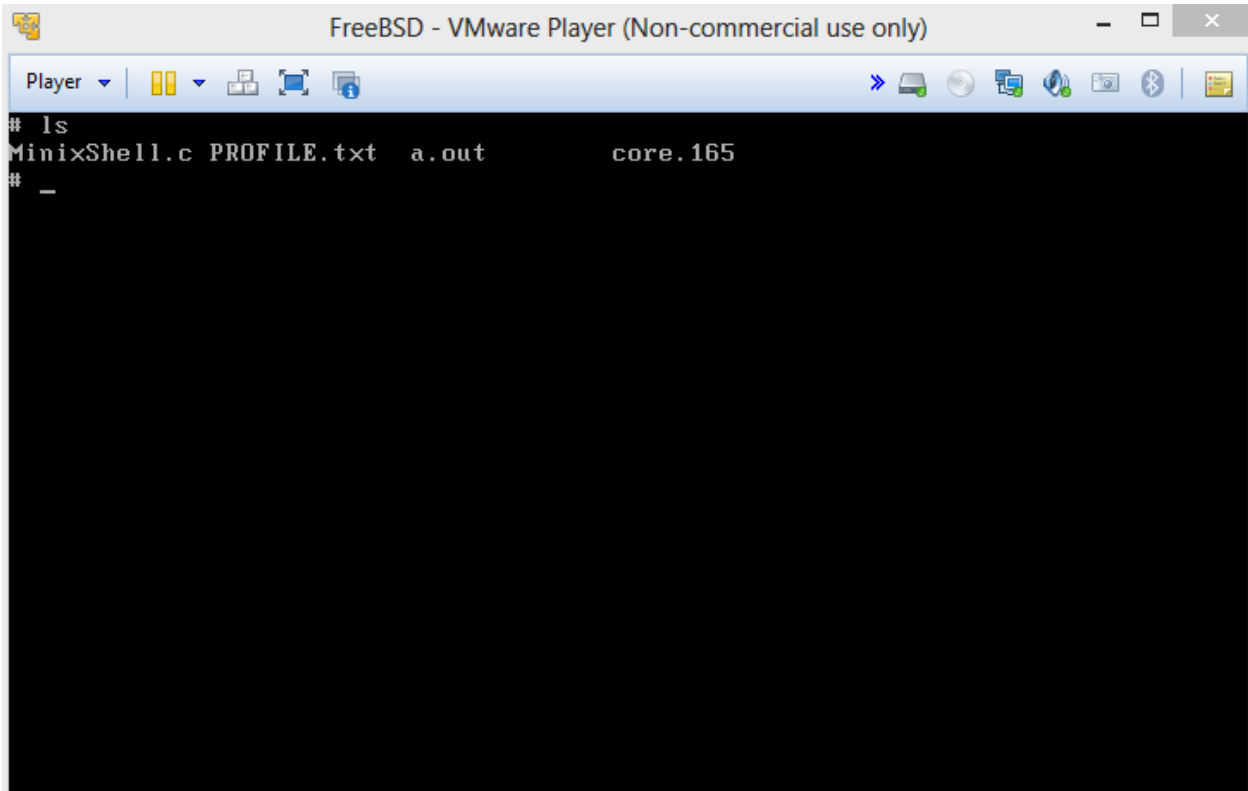
Input: Command "ls"

Expected Result: Displays list of files in the current directory

Actual Result: Displays list of files in the current directory

Result: Pass

Screen Shot:

A screenshot of a VMware Player window titled "FreeBSD - VMware Player (Non-commercial use only)". The window shows a terminal window with a black background and white text. The terminal prompt is "#". The user has entered the command "ls", and the output is displayed as follows:
ls
MinixShell.c PROFILE.txt a.out core.165

The terminal window has a standard toolbar with icons for Player, a dropdown menu, and various system icons like network, storage, and power.

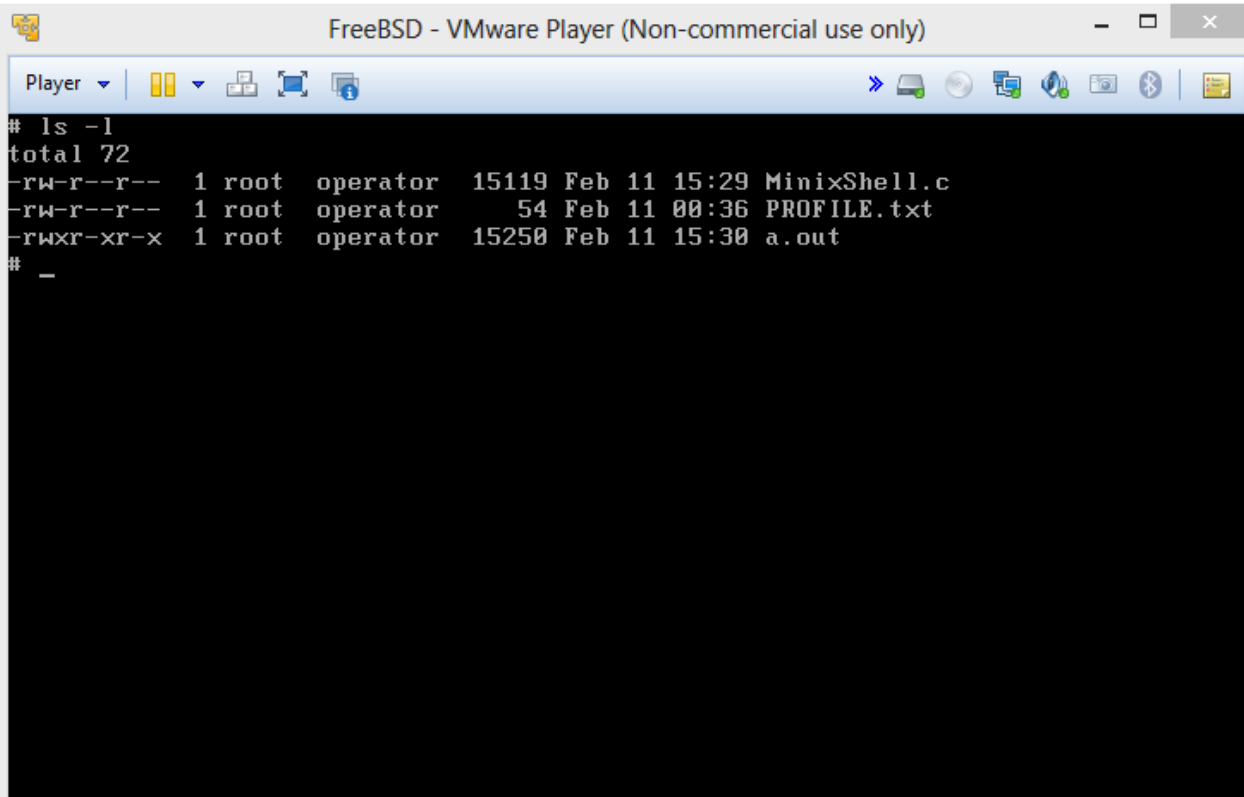
TEST CASE 5: To list the total files in the directory and subdirectories, the names of the files in the current directory, their permissions, the number of subdirectories in directories listed, the size of the file, and the date of last modification.

Input: "ls -l"

Expected result: Display list of files in present directory with details

Actual result: Display list of files in present directory with details

Result: Pass

Screen Shot:

The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal displays the output of the command `ls -l`. The output shows three files: `MinixShell.c`, `PROFILE.txt`, and `a.out`, each with its permissions, size, and modification date. The prompt `#` is visible at the bottom of the terminal.

```
# ls -l
total 72
-rw-r--r--  1 root  operator  15119 Feb 11 15:29 MinixShell.c
-rw-r--r--  1 root  operator    54 Feb 11 00:36 PROFILE.txt
-rwxr-xr-x  1 root  operator  15250 Feb 11 15:30 a.out
#
```

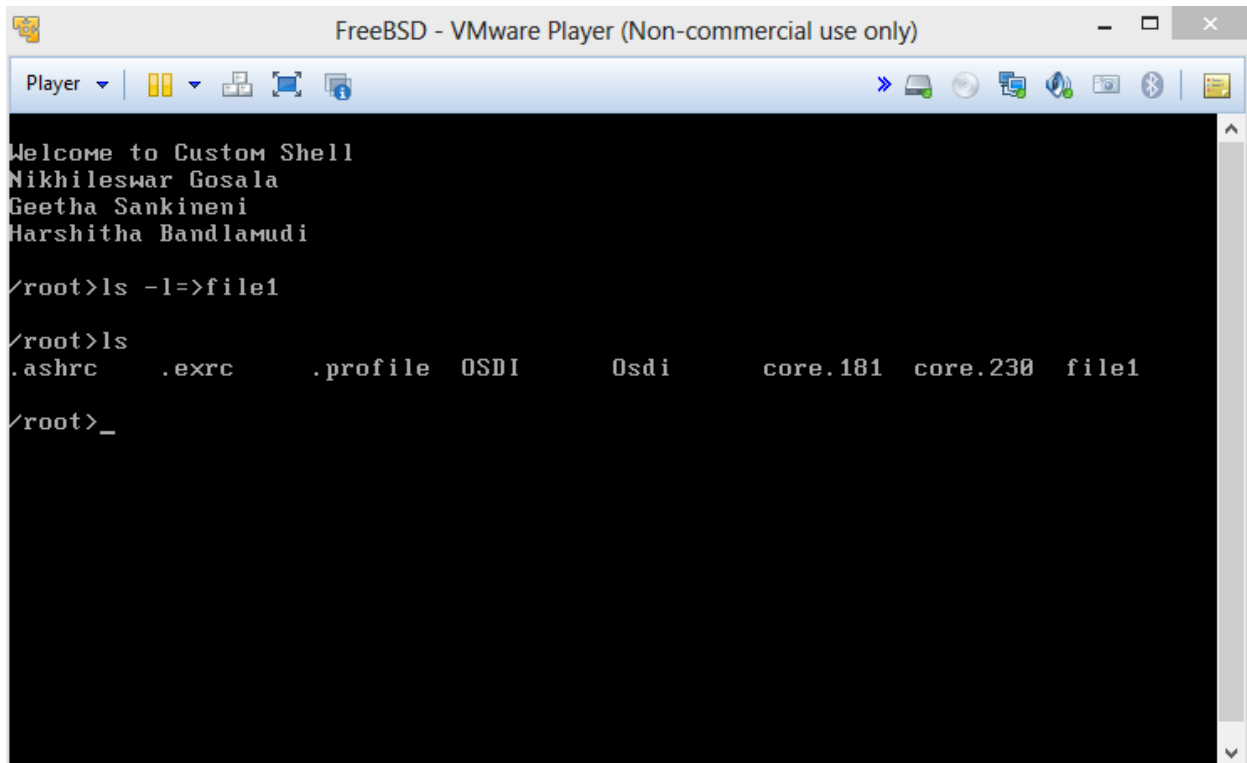
TEST CASE 6: To test the implementation of Indirection Operator

Input: `ls -l=>file1.txt`

Expected Result: Redirects the output to the file named file1.

Actual Result: Redirects the output to the file named file1.

Result: Pass

Screenshot:

The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal displays a custom shell prompt and a file listing. The text in the terminal is as follows:

```
Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>ls -l=>file1

/root>ls
.ashrc      .exrc      .profile  OSDI      OsdI      core.181  core.230  file1

/root>_
```

TEST CASE 7: Termination of shell using Ctrl + C / Exit

Input: Case1: Ctrl + C

Case2:Exit

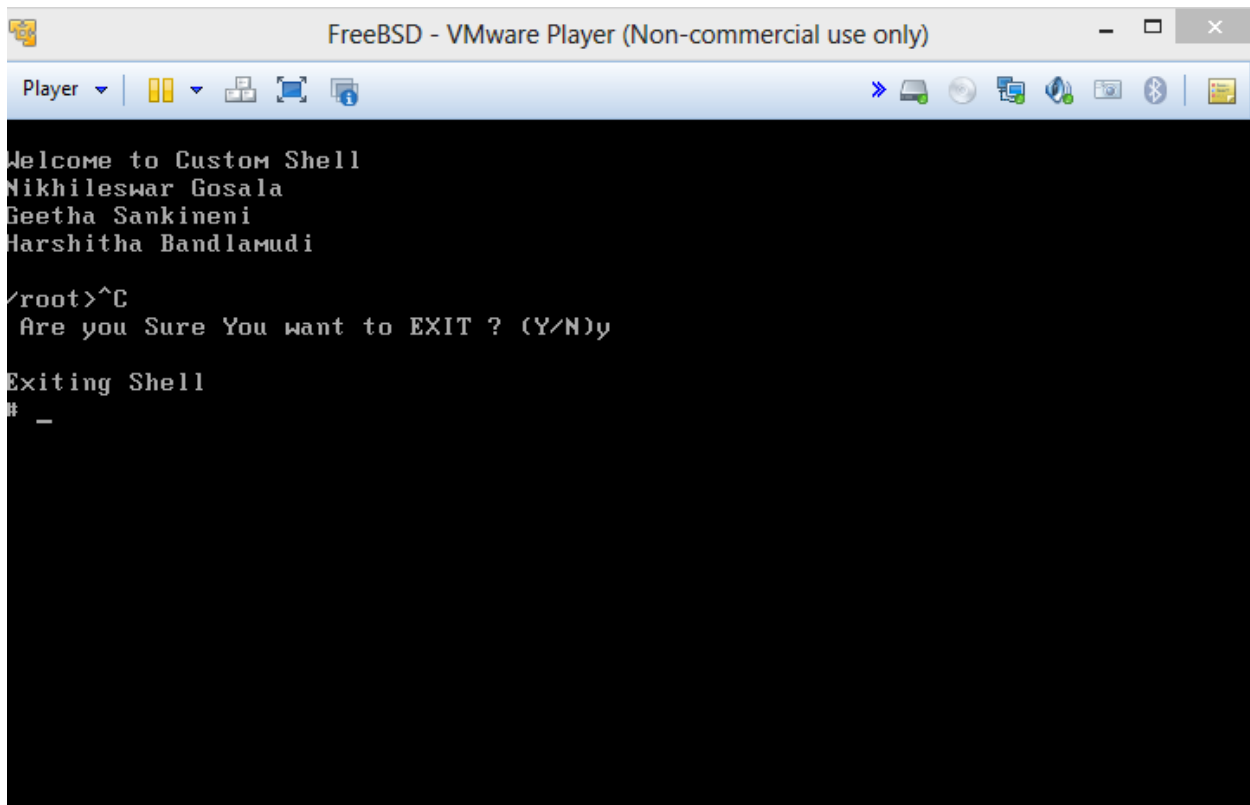
Expected Result: Closes the shell

Actual Result: Closes the shell

Result: Pass

Screen Shot:

Case 1:



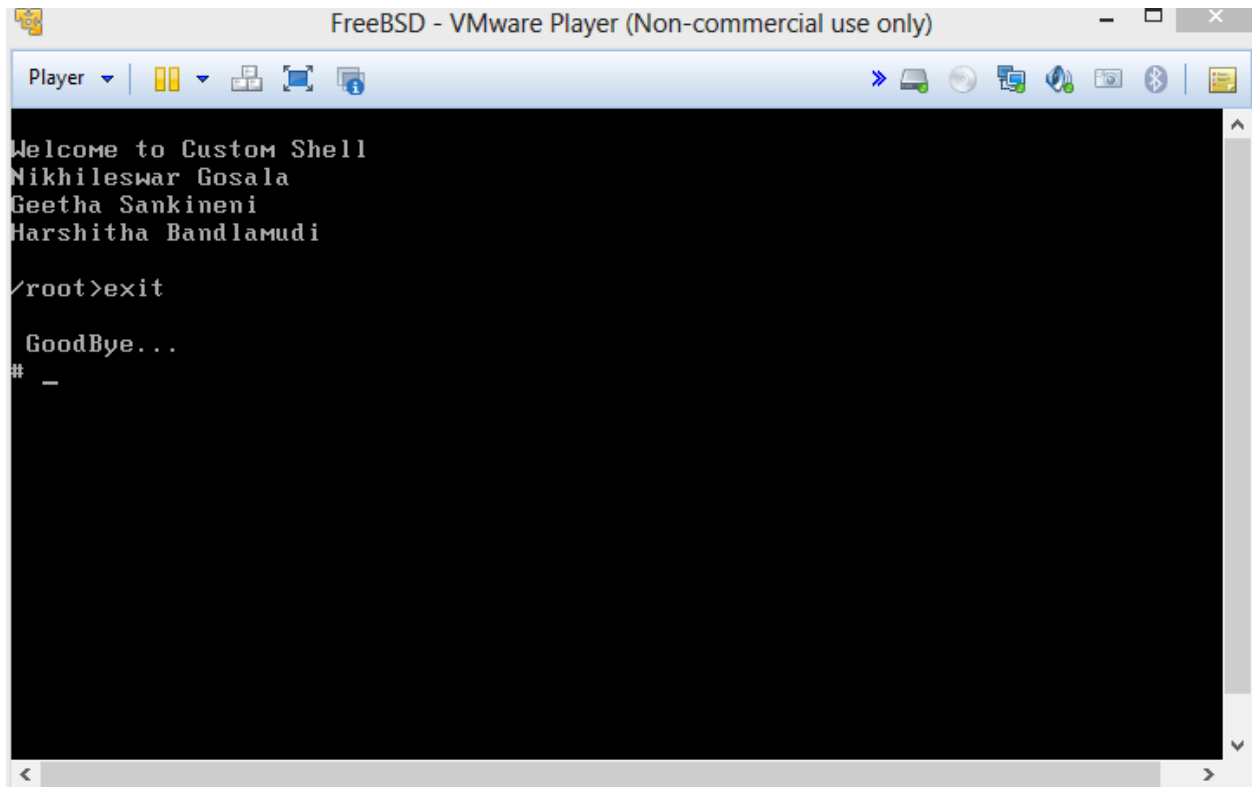
```
FreeBSD - VMware Player (Non-commercial use only)

Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>^C
Are you Sure You want to EXIT ? (Y/N)y

Exiting Shell
# _
```


Case 2:



```
FreeBSD - VMware Player (Non-commercial use only)

Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>exit

GoodBye...
# _
```

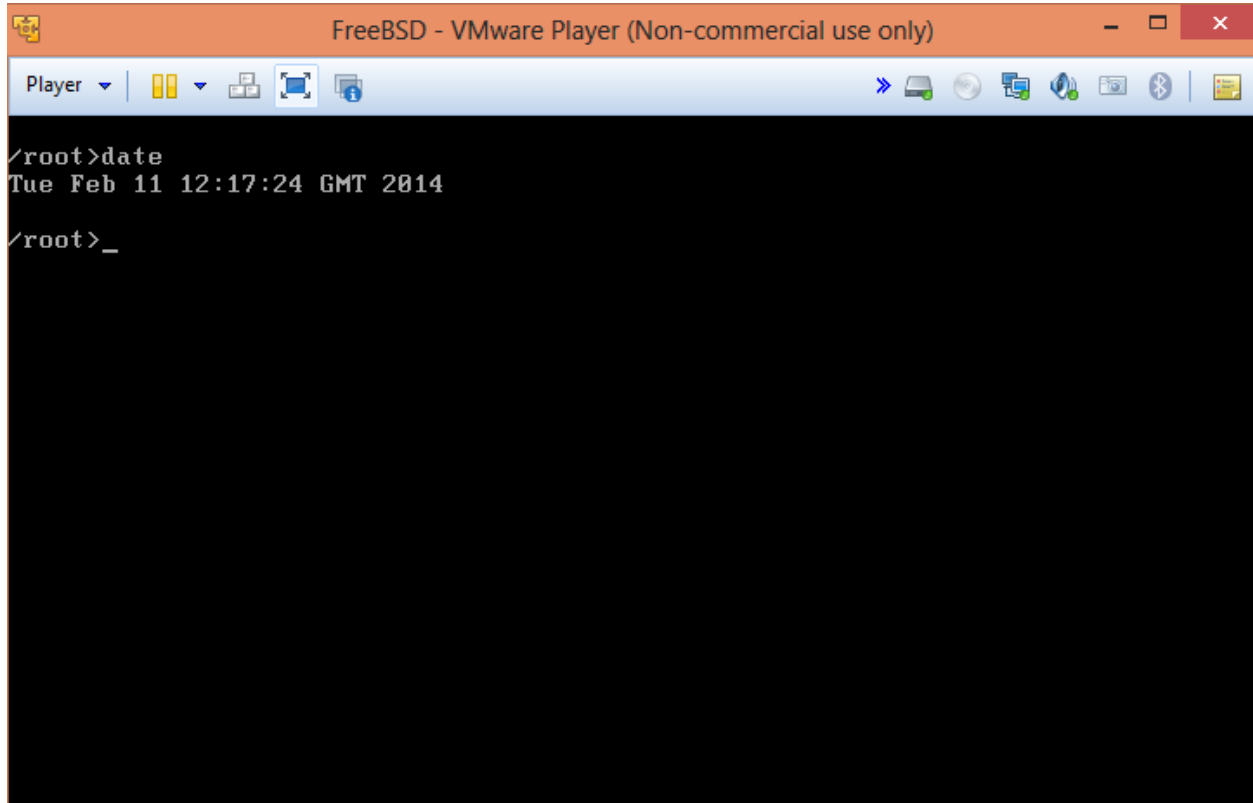
TEST CASE 8: To display the current date

Input: date

Expected result: Displays the current date

Actual result: Displays the current date

Result: Pass

Screen Shot:

```
FreeBSD - VMware Player (Non-commercial use only)
Player | [Icons]
/root>date
Tue Feb 11 12:17:24 GMT 2014
/root>_
```

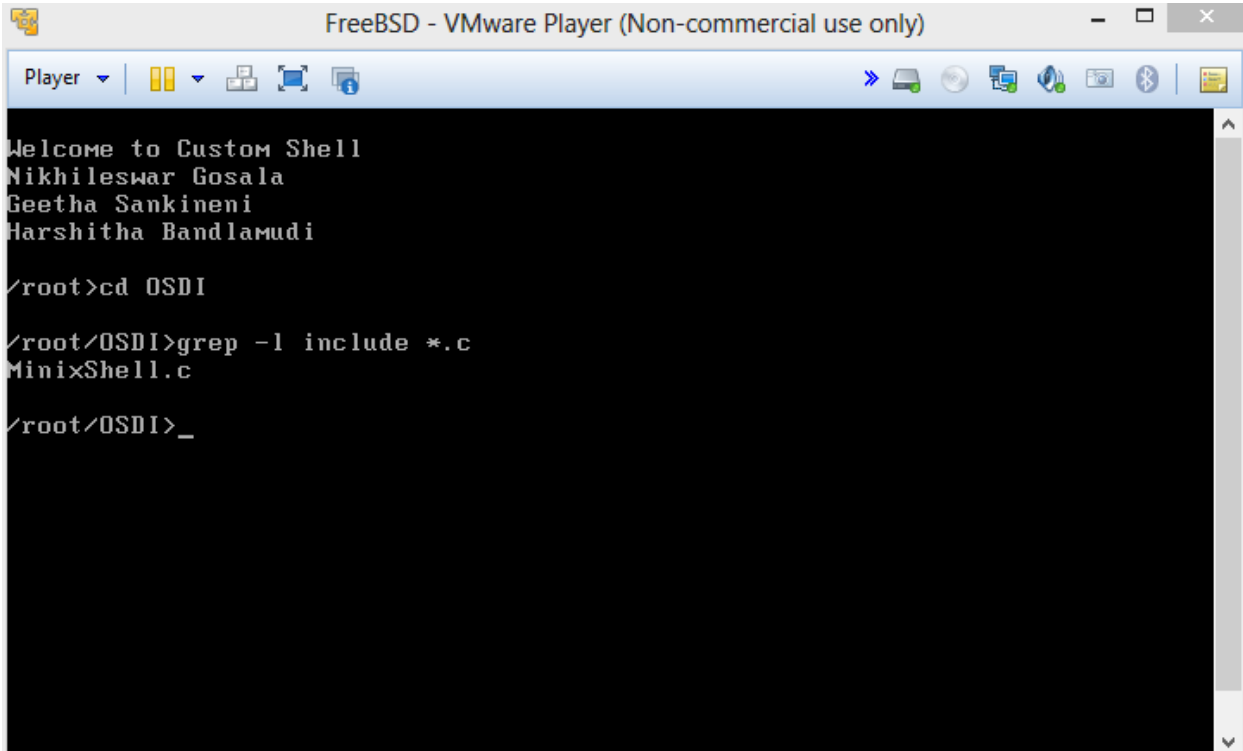
TEST CASE 10: To search the occurrence of include in the file

Input: `grep -l include *.c`

Expected Result : Displays any file which has include in it

Actual Result: Displays the file which has include in it.

Result: Pass

ScreenShot:

The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal displays a custom shell prompt "Welcome to Custom Shell" followed by the names "Nikhileswar Gosala", "Geetha Sankineni", and "Harshitha Bandlamudi". The user enters the command `/root>cd OSDI`. The prompt changes to `/root/OSDI>`. The user then enters `grep -l include *.c`, and the output `MinixShell.c` is displayed. The prompt returns to `/root/OSDI>`.

TEST CASE 10: To implement the facility that allows the input of one command to be run and output to be pasted back as input to another command

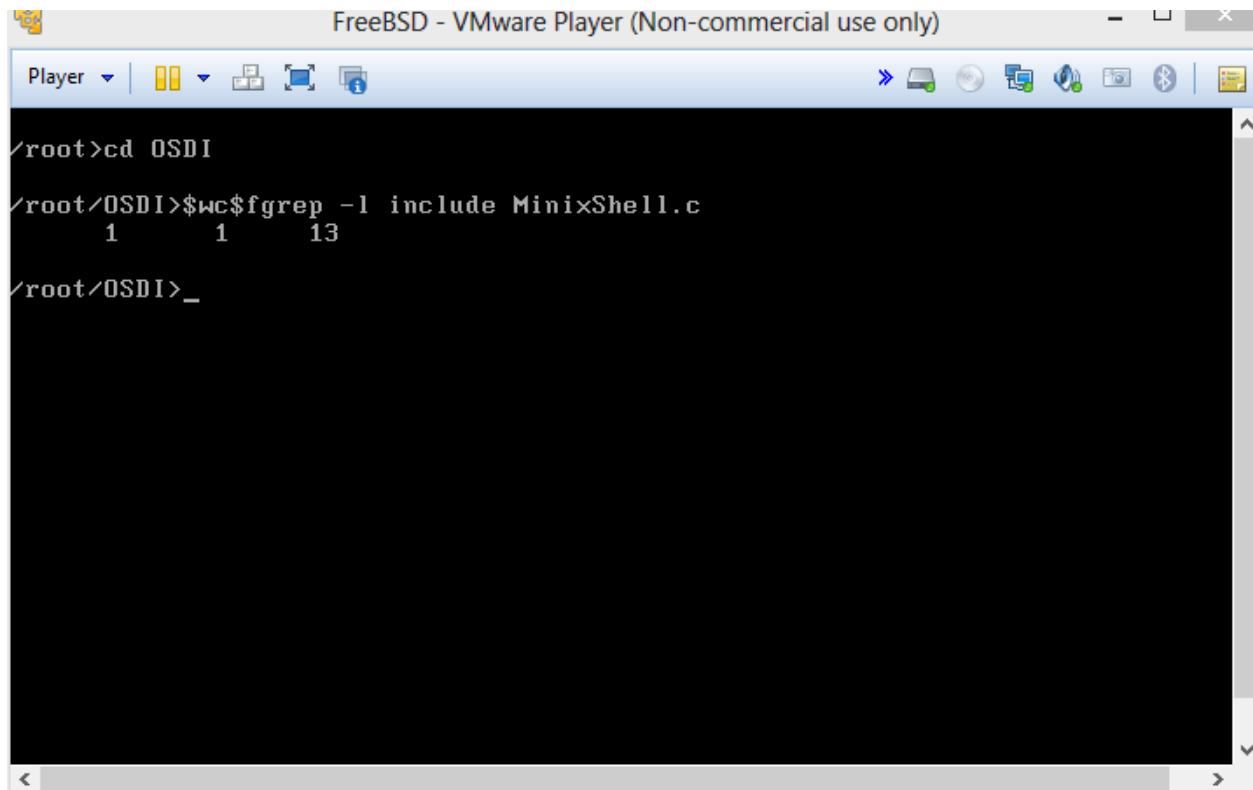
Input: `$wc $ fgrep -l include MinixShell.c`

Expected Result: Output of one command should be given as input of another command

Actual Result: Output of one command is given as input of another command

Result: Pass

Screen Shot:



The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal displays the following commands and output:

```
/root>cd OSDI
/root/OSDI>$wc$(fgrep -l include MinixShell.c
      1      1      13
/root/OSDI>_
```

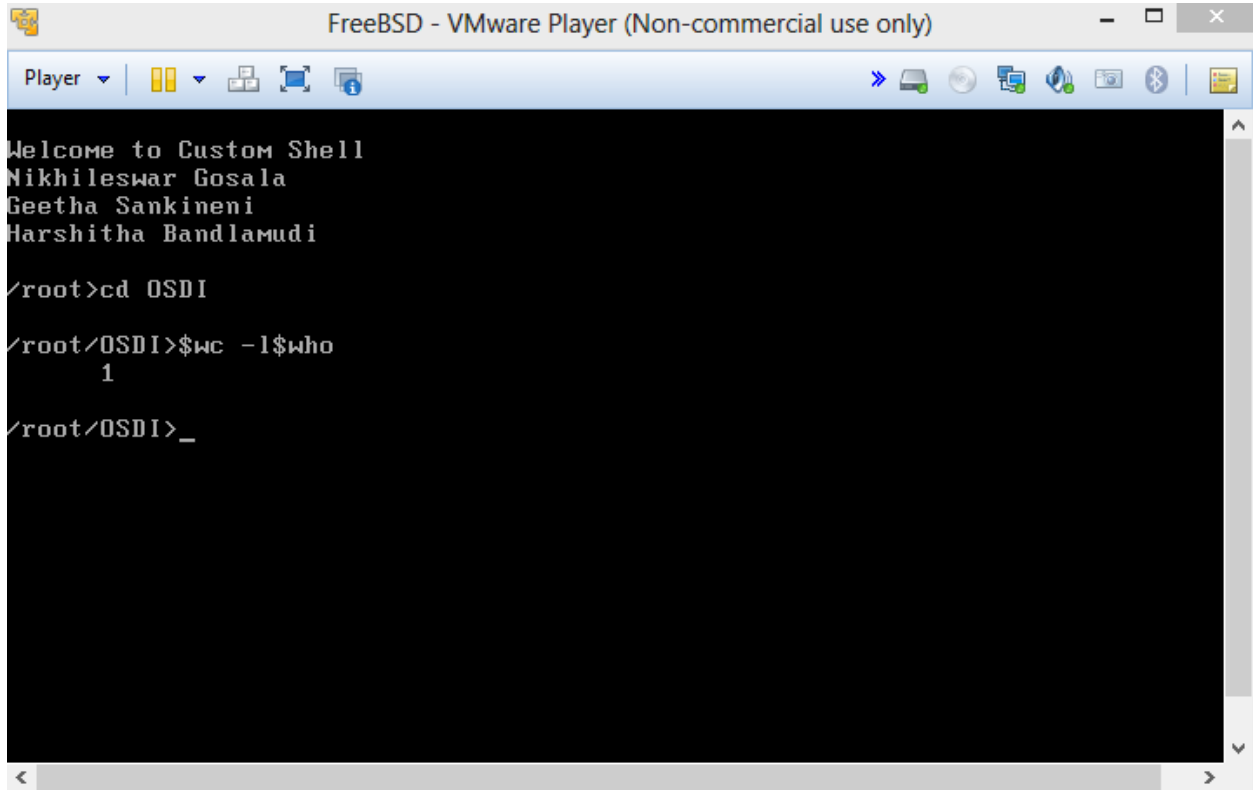
TEST CASE 11: Output of one command is given as input of another command so that it will display the number of users who logon to system

Input: \$wc -l \$(who)

Expected Result: Displays the number of users who logon to system

Actual Result: Displays the number of users who logon to system

Result: Pass

Screen Shot:

The screenshot shows a VMware Player window titled "FreeBSD - VMware Player (Non-commercial use only)". Inside the window is a terminal window with a black background and white text. The terminal displays a "Welcome to Custom Shell" message followed by the names "Nikhileswar Gosala", "Geetha Sankineni", and "Harshitha Bandlamudi". The user is at the root prompt and navigates to the "/root/OSDI" directory. They then execute the command "\$wc -l \$(who)", which outputs the number "1". The prompt returns to the root directory.

```
Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>cd OSDI

/root/OSDI>$wc -l$(who)
  1

/root/OSDI>_
```

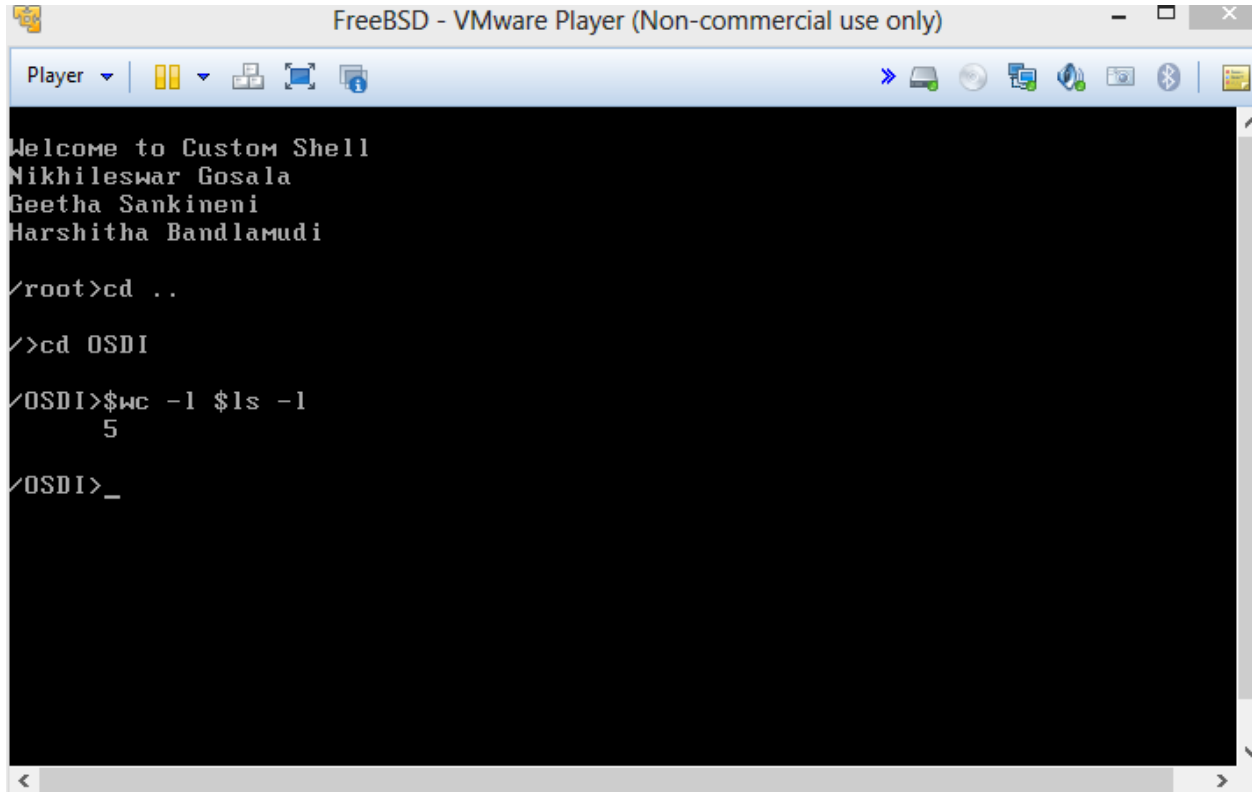
TEST CASE 12: Output of one command is given as input of another command so that it prints number of files in the current directory

Input: \$wc -l \$(ls -l)

Expected Result: Displays number of files in the current directory

Actual Result: Displays number of files in the current directory

Result: Pass

Screen Shot:A screenshot of a FreeBSD VM console window titled "FreeBSD - VMware Player (Non-commercial use only)". The window shows a custom shell interface. The prompt is "/root>". The user enters "cd ..", and the prompt changes to "/>". The user enters "cd OSDI", and the prompt changes to "/OSDI>". The user enters "\$wc -l \$ls -l", and the output is "5". The prompt returns to "/OSDI>".

```
Welcome to Custom Shell
Nikhileswar Gosala
Geetha Sankineni
Harshitha Bandlamudi

/root>cd ..

/>cd OSDI

/OSDI>$wc -l $ls -l
      5

/OSDI>_
```

TEST CASE 13: Implementation of nested piping with support to unlimited pipes

Input: "\$ grep shell.sh \$ grep -v grep \$ ps -ef"

Expected Result: Nested pipe is implemented

Actual Result: Nested pipe is implemented

Result: Pass

Screen Shot:

```

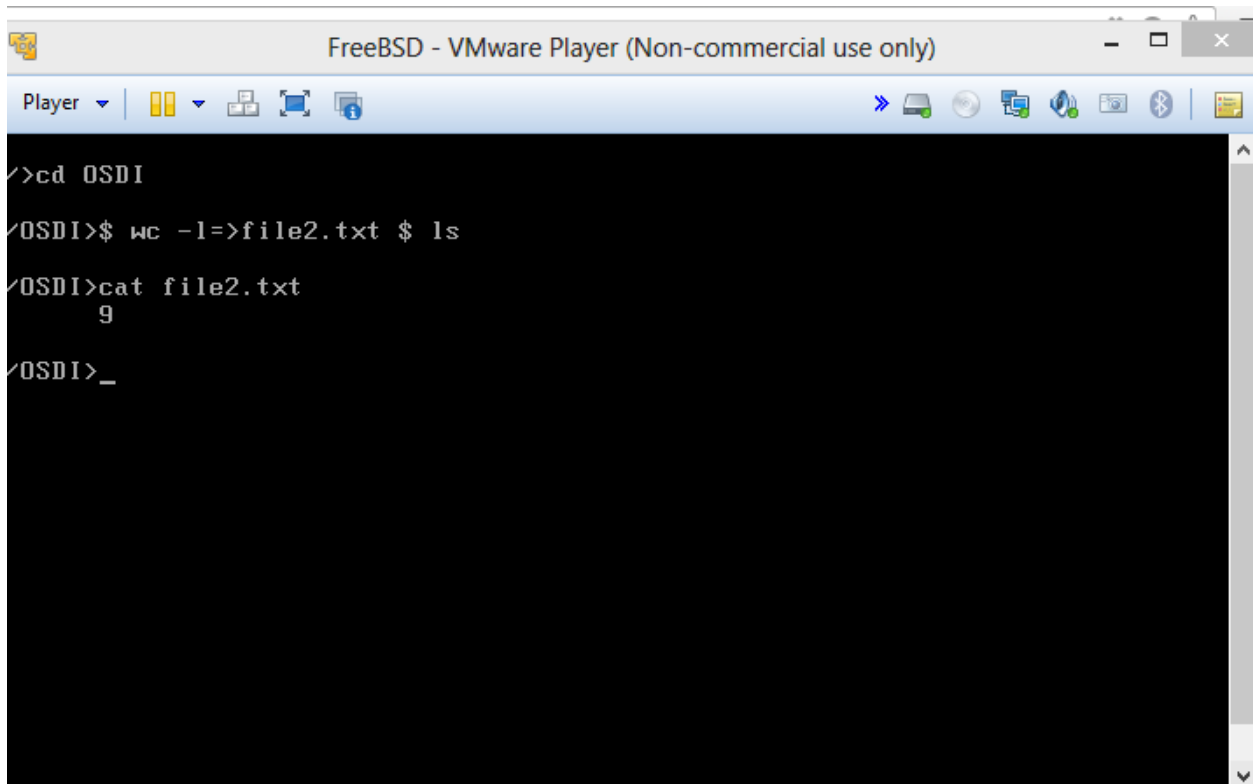
-o rw,rslabel=devman
W 0 77 4 0 144 ? 0:00 /usr/sbin/log
W 12 89 4 0 120 ANY ? 0:35 /usr/sbin/random
W 12 104 4 0 1112 ANY ? 0:00 /usr/sbin/e1000 instance=
0
W 12 108 4 0 1052 ANY ? 0:00 /usr/sbin/inet
W 12 111 4 0 84 ANY ? 0:00 /usr/sbin/printer
W 0 114 4 0 748 ANY ? 0:00 /usr/sbin/ipc
R 0 83 1 1 220 ? 3:52 devmand -d /etc/devmand -
d /usr/pkg/etc/devmand
S 0 118 1 1 184 (select) vfs ? 0:00 update
S 0 120 1 1 328 (sig susp) pm ? 0:00 cron
S 0 125 1 125 260 (select) vfs ? 0:00 syslogd
S 0 128 1 1 364 (select) vfs ? 0:00 dhcpd
S 0 130 1 1 348 (select) vfs ? 0:00 nonamed -L
S 0 139 1 139 2324 (select) vfs ? 0:00 /usr/pkg/sbin/sshd
S 0 140 1 140 680 (wait) pm co 0:00 -sh
S 0 141 1 141 200 (tty) vfs c1 0:00 getty
S 0 142 1 142 200 (tty) vfs c2 0:00 getty
S 0 143 1 143 200 (tty) vfs c3 0:00 getty
S 0 412 140 140 916 (pipe) vfs co 0:00 a.out
S 0 414 412 140 592 (wait) pm co 0:00 sh -c ps -ef
W 0 416 414 140 272 vfs co 0:00 ps -ef

```

TEST CASE 14: Implementation of pipe with indirection operator**Input:** \$wc -l=> file2.txt \$ ls

cat file2.txt

Expected Result: Output of the command is redirected to the file**Actual Result:** Output of the command is redirected to the file**Result:** Pass**Screen Shot:**



```
>cd OSDI
/OSDI>$ wc -l=>file2.txt $ ls
/OSDI>cat file2.txt
9
/OSDI>_
```

TEST CASE 15:Implementation of basic calculator**Input:**

Case 1: X =10

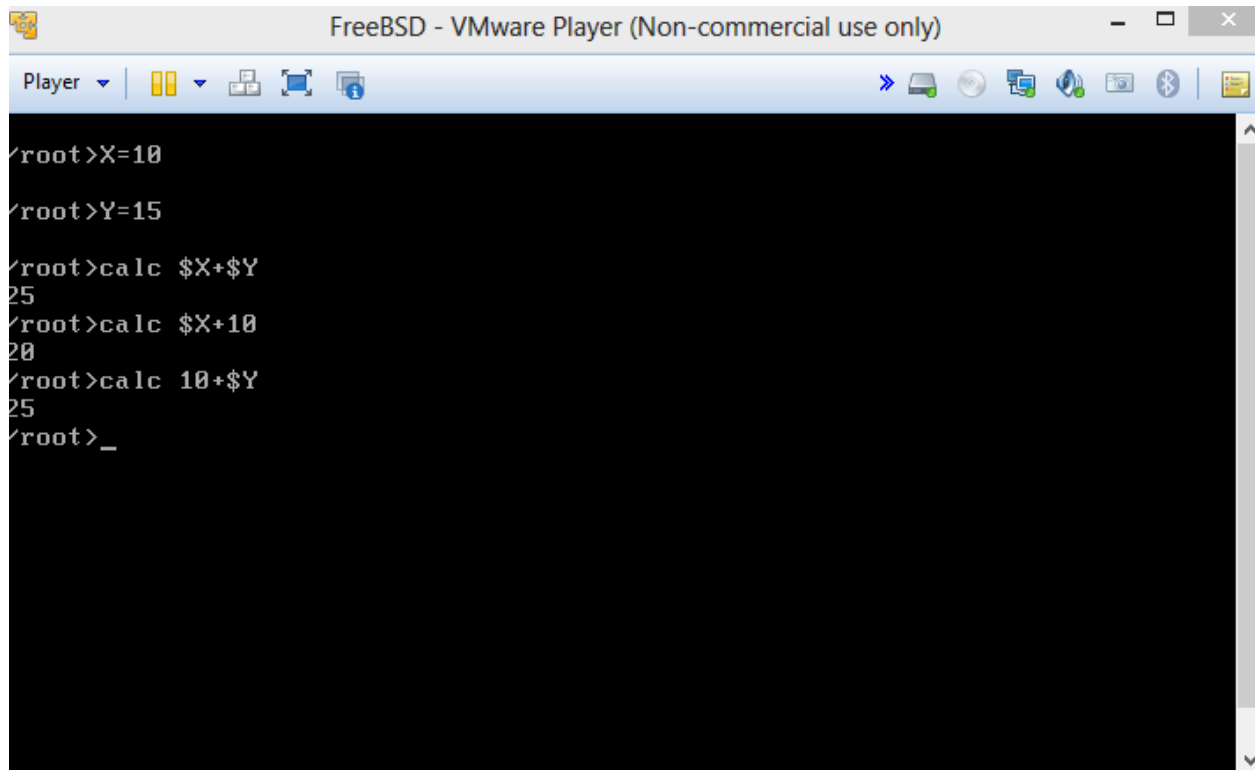
Y = 15

calc \$X + \$Y

calc \$X +10

calc \$10+Y

Expected Result: Shell should perform addition of numbers**Actual Result:** Displays the sum of numbers**Result:** Pass

Screen Shot:A screenshot of a FreeBSD VM terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal shows a series of commands and their outputs: setting X to 10, Y to 15, and then performing three addition calculations using the 'calc' command. The results are 25, 20, and 25 respectively. The prompt changes from root to _ after the last command.

```
/root>X=10
/root>Y=15
/root>calc $X+$Y
25
/root>calc $X+10
20
/root>calc 10+$Y
25
/root>_
```

Case 2:

Input : X =20

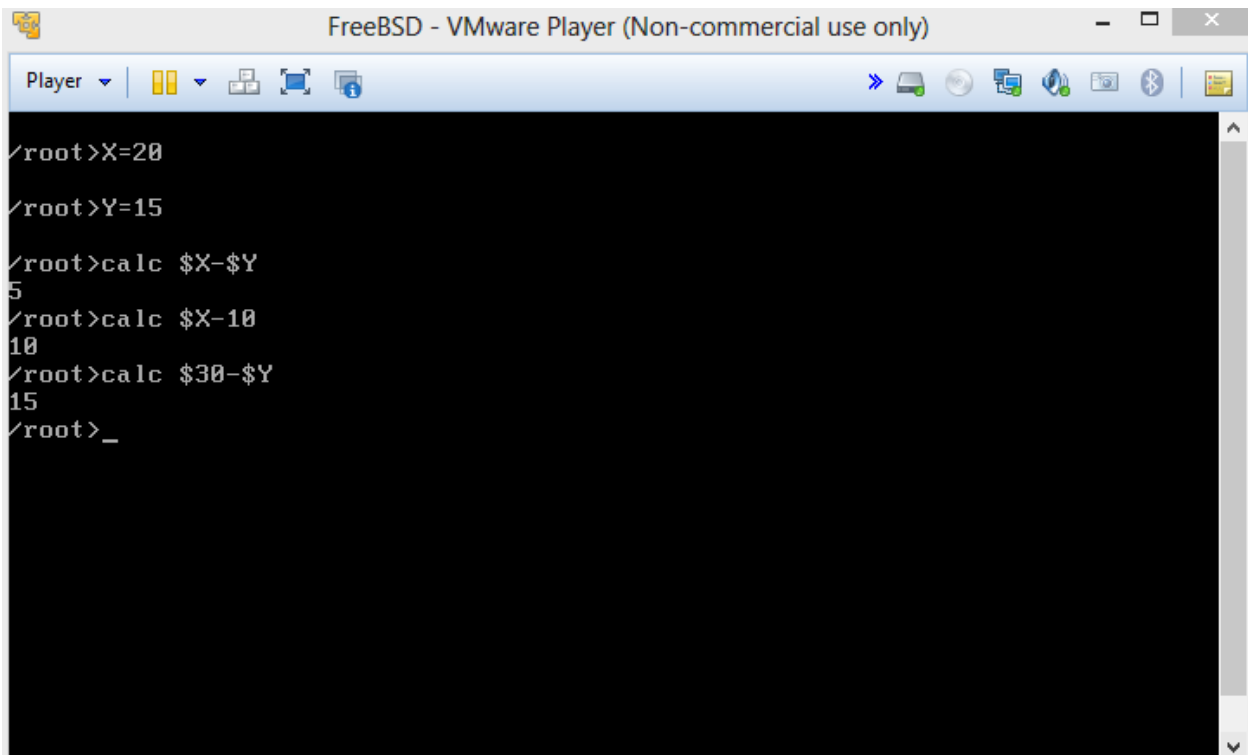
Y = 15

calc \$X - \$Y

calc \$X -10

calc \$30 - Y

Expected Result: Shell should perform subtraction of numbers**Actual Result:** Displays the difference of numbers**Result:** Pass**Screen Shot:**



```
FreeBSD - VMware Player (Non-commercial use only)

Player | [Icons] | [Icons]

/root>X=20

/root>Y=15

/root>calc $X-$Y
5
/root>calc $X-10
10
/root>calc $30-$Y
15
/root>_
```

Case 3:

Input: X =10

Y = 15

calc \$X * \$Y

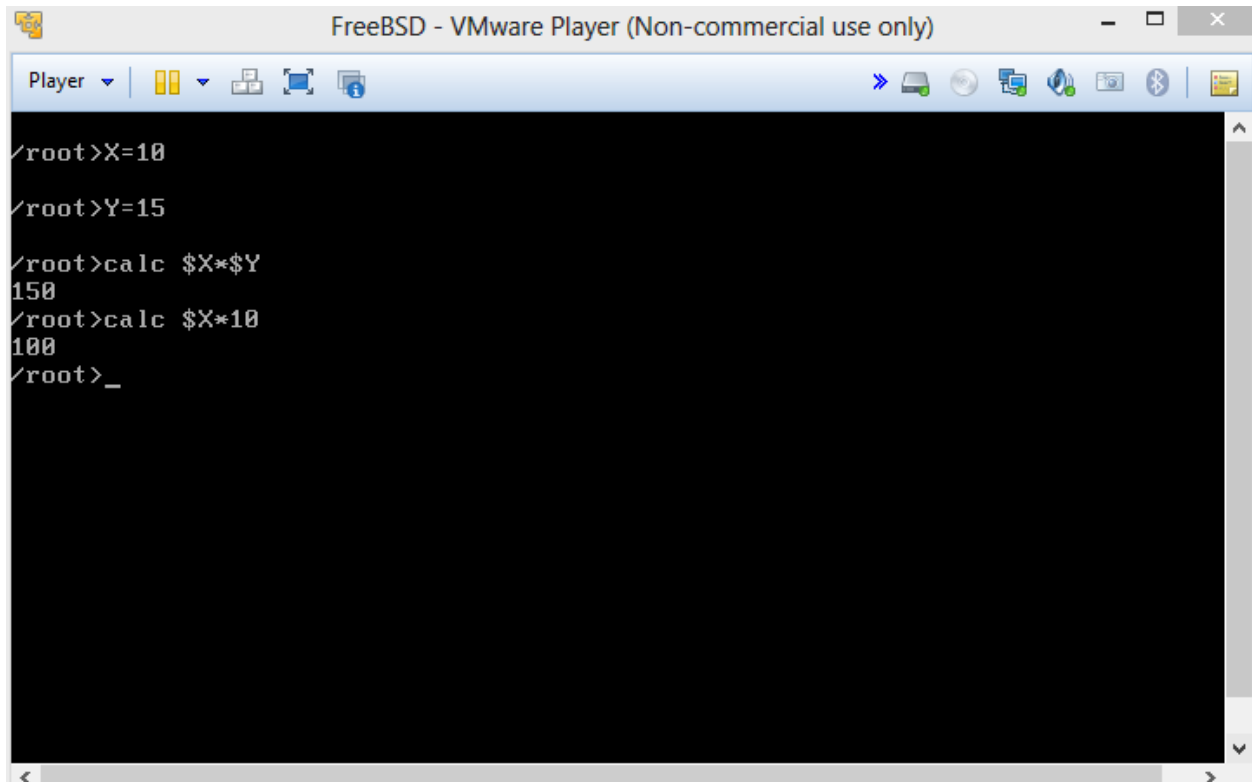
calc \$X *10

Expected Result: Shell should perform multiplication of numbers

Actual Result: Displays the product of numbers

Result: Pass

Screen Shot:



The screenshot shows a terminal window titled "FreeBSD - VMware Player (Non-commercial use only)". The terminal displays the following commands and outputs:

```
/root>X=10
/root>Y=15
/root>calc $X*$Y
150
/root>calc $X*10
100
/root>_
```

Case 4:

Input: X =20

Y = 5

calc \$X / \$Y

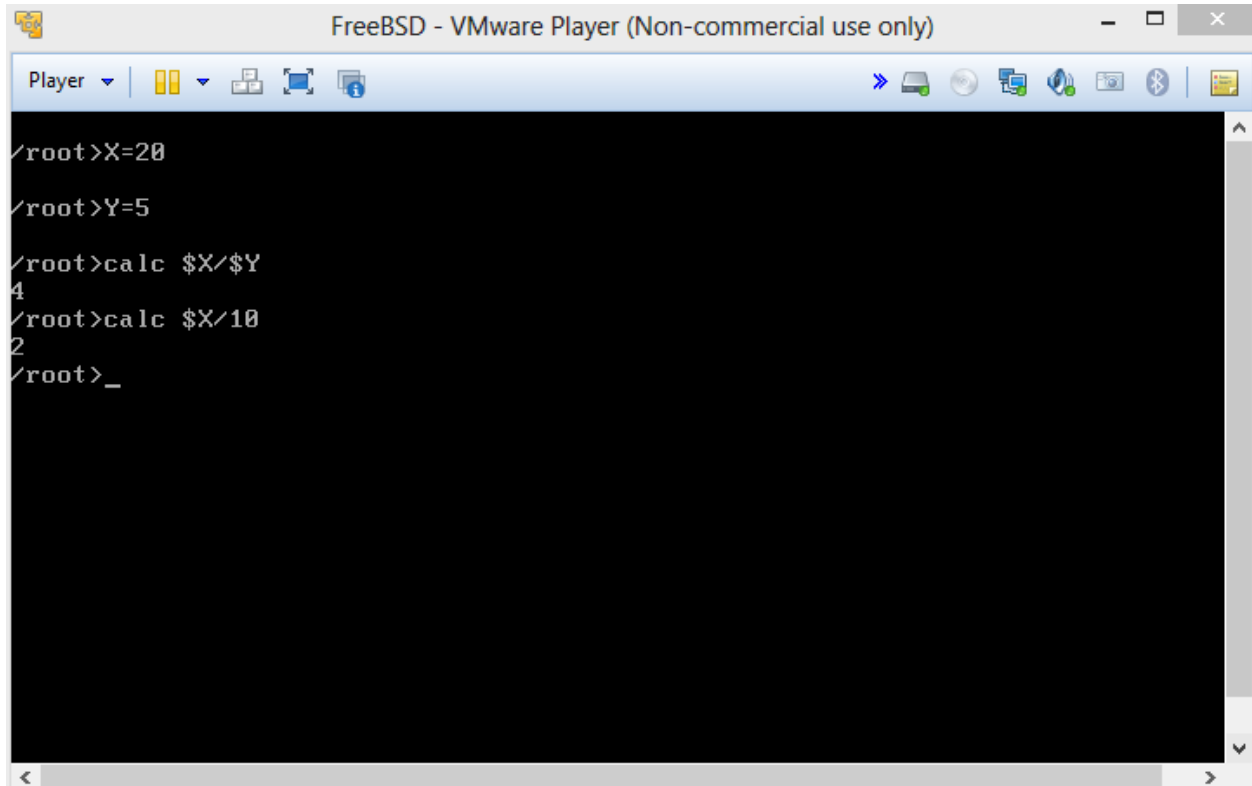
calc \$X /10

Expected Result: Shell should perform division of numbers

Actual Result: Displays the division of numbers

Result: Pass

Screen Shot:



```
FreeBSD - VMware Player (Non-commercial use only)

Player | [Icons] | [Icons]

/root>X=20

/root>Y=5

/root>calc $X/$Y
4
/root>calc $X/10
2
/root>_
```

EXTRACREDIT:

- Calculator can be directly used without another program from the command line using calc
Eg: calc 5 +4
calc \$X+\$Y
- Nested piping works with support to unlimited pipes
Eg:\$ grep shell.sh \$ grep -v grep \$ ps-ef

- Environment variables can be set by using “=” operator. They can be read using the same or echo command
HOME=/usr