

```
Adyant@DESKTOP-HTKVPBP MINGW64 ~/OneDrive/Desktop/unix/lab8 (main)
$ pip install mesg
DEPRECATION: Loading egg at c:\python3\lib\site-packages\vbboxapi-1.0-py3.11.egg is deprecated. pip 24.3 will enforce this behaviour change. A possible replacement can be found at https://github.com/pypa/pip/issues/12330
ERROR: Could not find a version that satisfies the requirement mesg (from versions: none)
ERROR: No matching distribution found for mesg

[notice] A new release of pip is available: 23.3.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip

Adyant@DESKTOP-HTKVPBP MINGW64 ~/OneDrive/Desktop/unix/lab8 (main)
$ chmod +x lab8.sh

Adyant@DESKTOP-HTKVPBP MINGW64 ~/OneDrive/Desktop/unix/lab8 (main)
$ ./lab8.sh
Using bc:
./lab8.sh: line 10: bc: command not found

Using comm:
Common lines:
apple
orange

Using chown:
-rw-r--r-- 1 Adyant 197121 0 Mar 27 14:19 example.txt
./lab8.sh: line 21: sudo: command not found
-rw-r--r-- 1 Adyant 197121 0 Mar 27 14:19 example.txt

Using chgrp:
./lab8.sh: line 25: sudo: command not found
-rw-r--r-- 1 Adyant 197121 0 Mar 27 14:19 example.txt

Using cron:
./lab8.sh: line 30: crontab: command not found
./lab8.sh: line 31: crontab: command not found

Using dd:
0+1 records in
0+1 records out
14 bytes copied, 0.0001684 s, 83.1 kB/s

Using diff:

Using finger:

Displays information about a user on a specified system running the
Finger service. Output varies based on the remote system.

FINGER [-l] [user]@host [...]
```

```
$ ./lab8.sh
Using bc:
./lab8.sh: line 10: bc: command not found

Using comm:
Common lines:
apple
orange

Using chown:
-rw-r--r-- 1 Adyant 197121 0 Mar 27 14:18 example.txt
./lab8.sh: line 21: sudo: command not found
-rw-r--r-- 1 Adyant 197121 0 Mar 27 14:18 example.txt

Using chgrp:
./lab8.sh: line 25: sudo: command not found
-rw-r--r-- 1 Adyant 197121 0 Mar 27 14:18 example.txt

Using cron:
./lab8.sh: line 30: crontab: command not found
./lab8.sh: line 31: crontab: command not found

Using dd:
0+1 records in
0+1 records out
14 bytes copied, 0.0001615 s, 86.7 kB/s

Using diff:

Using finger:

Displays information about a user on a specified system running the
Finger service. Output varies based on the remote system.

FINGER [-l] [user]@host [...]

-l          Displays information in long list format.
user        Specifies the user you want information about. Omit the user
            parameter to display information about all users on the
            specified host.
@host       Specifies the server on the remote system whose users you
            want information about.
```

```
@host    specified host.
         Specifies the server on the remote system whose users you
         want information about.
```

```
Using find:
./example.txt
./file1.txt
./file2.txt
./source.txt
./symlink.txt
./target.txt
```

Using ftp:

```
Using lock:
This command doesn't exist in Windows.
```

```
Using ln:
Symbolic links behave differently in windows. Skipping.
```

```
Using lp:
The 'lp' command doesn't exist in windows.
```

```
Using lpstat:  
The 'lpstat' command doesn't exist in Windows.
```

```
Using mesg:
The 'mesg' command doesn't exist in Windows.
```

~~Adaptation DECKTOR_HTM/DBB_MINGUC4 /cc=Deine/Dackler/Luxin/7ch8 (main)~~

```
Adyant@DESKTOP-HTKVPBP MINGW64 ~/OneDrive/Desktop/unix/lab8 (main)
```

```
$ bc --help
bash: bc: command not found
```

```
Adyant@DESKTOP-HTKVPBP MINGW64 ~/OneDrive/Desktop/unix/lab8 (main)
```

```
$ lp --help
By image:
  lp SCREENSHOT [COUNT]
```

SCREENSHOT is a path to a Letterpress screenshot.

```
Run server:
  lp --listen=ADDRESS
```

ADDRESS is the bind address for a server.

For example, you could run:
 lp --listen=127.0.0.1:8081
and then visit <http://127.0.0.1:8081> in your browser.

You can specify a language in your environment as LP_LANG, so for example
you could run:
 LP_LANG=de lp [args]

```
Adyant@DESKTOP-HTKVPBP MINGW64 ~/OneDrive/Desktop/unix/lab8 (main)
```

```
$ lp --listen=127.0.0.1:8081
* Serving Flask app 'lp.server'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:8081
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 807-905-541
127.0.0.1:8081
127.0.0.1:8081
```