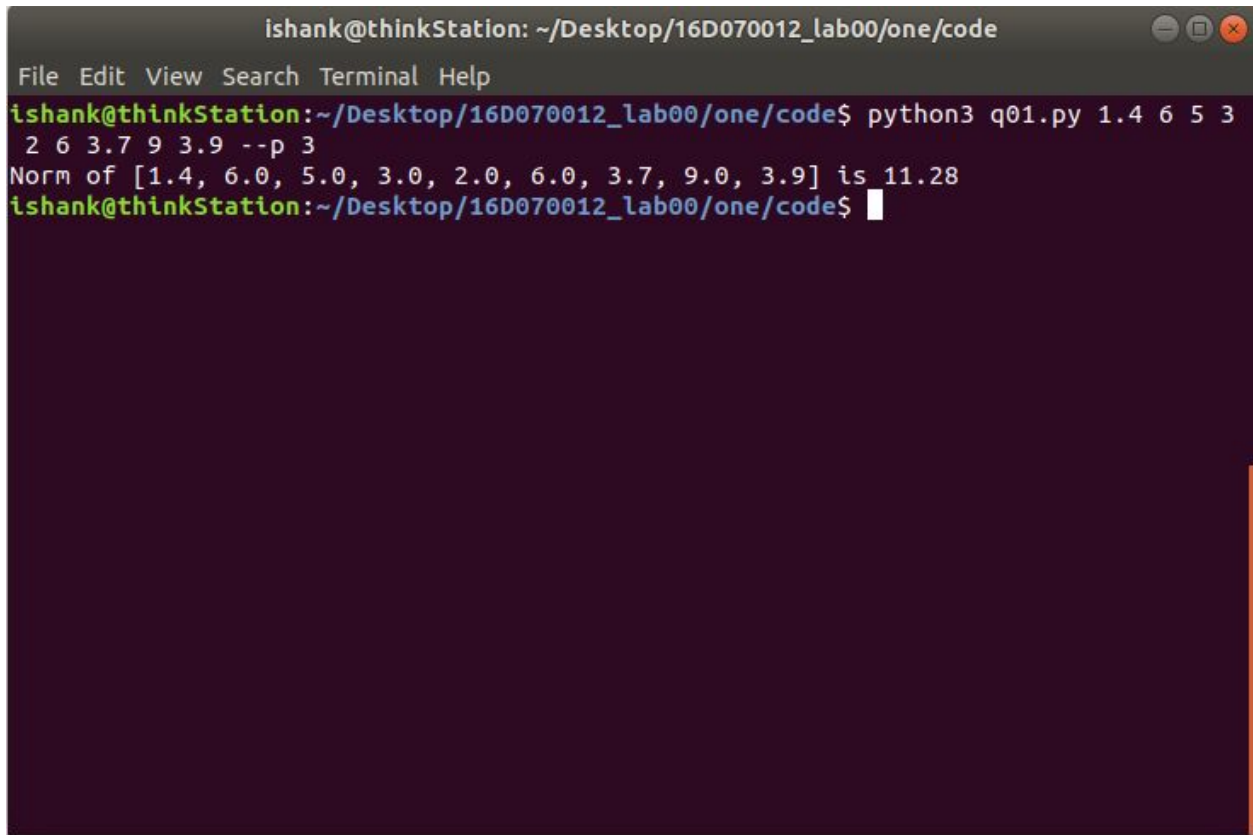


Was a simple problem to solve.

Went through online tutorial on argparse since hadn't since parsing command line lists before. Then verified the working of the code using the given and a few additional test cases.

Sample Run

A terminal window titled 'ishank@thinkStation: ~/Desktop/16D070012_lab00/one/code'. The terminal shows the execution of a Python script 'q01.py' with command-line arguments '1.4 6 5 3 2 6 3.7 9 3.9 --p 3'. The output of the script is 'Norm of [1.4, 6.0, 5.0, 3.0, 2.0, 6.0, 3.7, 9.0, 3.9] is 11.28'. The terminal window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The background is dark purple with light green text for the prompt and output.

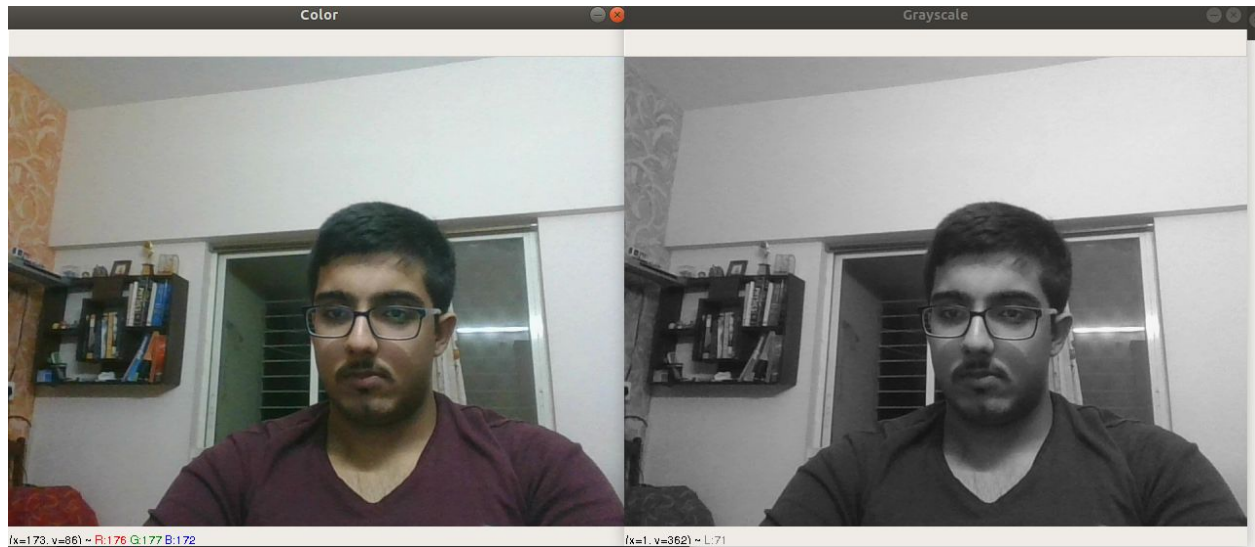
```
ishank@thinkStation: ~/Desktop/16D070012_lab00/one/code
File Edit View Search Terminal Help
ishank@thinkStation:~/Desktop/16D070012_lab00/one/code$ python3 q01.py 1.4 6 5 3
2 6 3.7 9 3.9 --p 3
Norm of [1.4, 6.0, 5.0, 3.0, 2.0, 6.0, 3.7, 9.0, 3.9] is 11.28
ishank@thinkStation:~/Desktop/16D070012_lab00/one/code$
```

What I learnt

Argparse simplifies passing command line arguments of variable length and generating appt. Error messages.

Problem involved use of openCV and matplotlib plotting/image viewing functionality

Sample Run for Video capture from webcam, for other parts it was interactive



What I learnt

OpenCV has a lot of built-in functions for quick image processing tasks.

OpenCV uses BGR ordering of frames while matplotlib uses RGB, it is crucial to convert to right format before displaying/processing images

Problem involved some straightforward numpy manipulations.

Sample Run

```
ishank@thinkStation: ~/Desktop/16D070012_lab00/two/code
File Edit View Search Terminal Help
ishank@thinkStation:~/Desktop/16D070012_lab00/two/code$ python3 q01.py
Original array:
[[ 4.  1.  2.  3.]
 [ 8.  5.  6.  7.]
 [23. 12.  0.  1.]
 [29. 94. 58. 38.]]

Cropped array:
[[ 5.  6.]
 [12.  0.]]

Padded array:
[[ 0.5  0.5  0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5  0.5  0.5]
 [ 0.5  0.5  5.   6.   0.5  0.5]
 [ 0.5  0.5 12.   0.   0.5  0.5]
 [ 0.5  0.5  0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5  0.5  0.5]]

Concatenated array: shape=(6, 12)
[[ 0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5]
 [ 0.5  0.5  5.   6.   0.5  0.5  0.5  0.5  5.   6.   0.5  0.5]
 [ 0.5  0.5 12.   0.   0.5  0.5  0.5  0.5 12.   0.   0.5  0.5]
 [ 0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5]
 [ 0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5  0.5]]
ishank@thinkStation:~/Desktop/16D070012_lab00/two/code$
```

What I learnt

Numpy is a useful tool for DIP and CV manipulations