Python, NumPy and OpenCV basics

In this lab exercise I performed some tasks using Python, numpy and opencv. This was divided in basically 3 main portions:

1. Python
2. NumPy
3. OpenCV

1. **Python:** In this exercise I were to write a program for performing p-norm of a vector using only python standard library. Main challenge that was faced in this exercise was to take the arguments from command prompt. I learnt how to use argparse function and how to use positional and optional arguments.
2. **Numpy:** Here I were to implement two programs namely row manipulation and K-means.
    a. **Row manipulation:** In this I were to find a permutation matrix (P) that were to be used for performing the array conversion. The main challenge in this task was to implement it without for loops. So, I used some functions and properties of numpy and converted a identity matrix into P-matrix. There were three more tasks associated with this:
        i. crop_array: where we were to crop array by given offset_height, offset_width, target_height, target_width. This was performed by taking the slices of the original array.
        ii. Padding: numpy padding function was used for this task.
        iii. Concatenation: Again numpy concatenate function was simply used to perform this task.
    b. **K-Means:** In this task we were to implement K-means without any external library and then to create a 2 dimensional data to make three clusters. This was implemented using the k-means algorithm and numpy. For plotting the data matplotlib was used.
3. **OpenCV:** In this section I were to write three programs:
    a. **Image Conversion:** Read an image from file, display it using matplotlib, openCV and save the original and normalized image. For normalizing the image, whole pixel values were divided by 255 and image was read and saved using several matplotlib and OpenCV functions.
    b. **Display Images:** In this we were to show images from a given directory in a wrap around fashion. To perform this all the images were stored in an list of arras and then displayed one by one. For performing wrap around modulus operator was used.
    c. **Video Input/ Output:** Here we were to read a video/ webcam input and display the video after putting a text and also in greyscale. When the program is called with arguments it will open the video file given by the link of the function argument then put the text on it, show it in colour and greyscale version then saves the updated file. Similar thing was done for webcam input except for saving the video.