# Lab 5 – Python and OpenCV

Prior to commencing with the tasks, ensure that you:

1. Install the Raspberry Pi Camera Module **prior** to plugging in the Pi.
2. Synchronise the time on the Pi, either by connecting to a different network, i.e. a Hotspot, or by using: `sudo date -s 'YYYY-MM-DD hh:mm:ss`
3. Clone the Lab5 Repository from GitHub Classroom **into the ee347 folder**, using this link. Open the ee347 folder in VSCode and work from there. **Do not move the .venv folder.**
4. Commit and Push to GitHub after each task. Each task should be completed in the individual taskX.py scripts provided.

## Tasks

1. Load task1.jpg and display it in an OpenCV window.
2. Load task1.jpg, convert it to grayscale, and save as task2.jpg.
3. Load task1.jpg, draw a red line, a green square and a blue circle on the image. Save as task3.jpg.
4. Load task1.jpg, run Sobel edge detection on the image, and save as task4.jpg.
5. Use the Raspberry Pi camera to capture an image and save it as task5.jpg. You will need to use the picamera2 package to access the Raspberry Pi camera.
6. Use an OpenCV window to display a live preview of the camera output. Ensure the orientation matches.
7. Use cv2.waitKey() to capture an image each time the space key is pressed. Save the captured image as task7.jpg.
8. Adjust the code from task 7 to save the image to a numpy file task8.npy
9. Use task5.py to capture an image of one member of your group. Using haarcascade_frontalface_default.xml, which is provided by default in your OpenCV installation, detect the face of the group member. Draw a bounding box around the face, preview it with an OpenCV window and save as task9.jpg.
10. Using haarcascades, show a live preview in which a bounding box is drawn around **all** faces in the frame.