# MRT Assignment 1
# ROS and Python

## 1 Task-1

If you have completed the ROS wiki tutorials(which you should have by now), you would have made listener and talker nodes, communicating via a rostopic. In reality, we may require to transfer complex messages and not just strings, images being a part of the such messages. Your task is to make a publisher which would take images from your webcam(you can even use any compatible video file) which would publish the images to a rostopic[1](name it "*htg_sensei's_feed*"....just kidding, give any other appropriate name) . You might need something called `cv_bridge` for converting the image feed to a standard ros message.

## 2 Task-2

In task-2, you are required to make a subscriber(in a separate python file) to the rostopic you created above and do some processing to the image(You'll need `cv_bridge` again to revert to OpenCV format). For each image, use Canny edge detection[1] method to find the edge image of the subscribed image. Then display the edge image and the original image horizontally stacked as a continuous feed.

## 3 Task-3

After making the above nodes, run both the nodes together and create an rqt_graph to visually analyse the connections between nodes. Summarize your learning in a LaTeX report, embedding atleast one image from the horizontally stacked images, and one image of the `rqt_graph`.

## 4 Resources

1. For learning LaTeX create an account on overleaf. You can learn from this tutorial.
2. cv_bridge

## 5 Extras

To maintain repository of solutions to the MRT assignments, git is a handy tool which you would have learnt from the previous tutorials. Create a remote git repository and push your codes and report into the remote repository.

[1]Take appropriate values for parameters. For example, the rate of publishing messages can be set to 30, and the parameters of Canny algorithm can be 100,200.