ME 495: Embedded Systems in Robotics

Dr. Matthew Elwin
elwin@northwestern.edu

9/24/2019

Location: Tech L170
Time: TuTh 9:30-10:50 AM
Office Hours: Tu 4-5, Wed 4-5 (in A275), and by appointment.
URL: https://nu-msr.github.io/me495_site

Textbook: *A Gentle Introduction to ROS*, by J. O'Kane (ISBN 978-1492143239) is optional and can be downloaded for free from (http://www.cse.sc.edu/~jokane/agitr/). Or purchased. **Software:** This course requires a Linux computer with root access. You must use Ubuntu 18.04 LTS or receive explicit permission from me.

You also need the "desktop-full" version of ROS Melodic Morenia. If you require help installing Linux or ROS please contact me or come to my office hours. ROS must be installed by the beginning of the second class.

Other software will be downloaded throughout the class. All software used in the class is free and much of it is developed by volunteers sharing their code with the world. You are encouraged to do the same with your projects.

Course Description: This project-based course provides experience with a variety of software tools and concepts useful for a robotics engineer working with practical embedded systems. The Robot Operating System (ROS) will be used as an example framework, and learning ROS will be a primary goal of the course. By the end of the course, student teams will have completed a robotics project using a real robot.

Prerequisites: Proficiency in at least one programming language is required (`MATLAB` is fine), and experience in Python, C, C++, Linux, and git would all be useful. The primary programming language used in the course will be Python.

Course Structure: The course is divided into five sections, covering the following topics:

1. Core ROS concepts and infrastructure (nodes, messages, catkin, etc..).
2. Basic ROS tools (urdf, tf, robot state publisher, rviz, etc).
3. Simulation, Navigation, and Image Processing with ROS
4. Simulation and Manipulation with ROS.
5. Final Project using Baxter/Sawyer robots and other hardware.

The final project will be conducted in teams and will be demonstrated on **Monday 12/09/2019 from 3pm - 5pm**.

Grading Each of the five sections will be accompanied by an assignment, with each assignment worth one-fifth of your grade. It is imperative to complete all homework assignments so you can be a valuable member of your final project team.

If you make a good faith effort to complete all homework assignments and the final project (at my sole discretion), I will substitute your final project grade for your lowest homework grade, if it will help your grade.

Late homework will be penalized as follows:
24 hours late: lose 10 percent of the total
48 hours late: lose 20 percent of the total
72 hours late: lose 30 percent of the total
Later than 72 hours: 0.

Disability Accommodation: If you require disability accommodations, please register with AccessibleNU (https://www.northwestern.edu/accessiblenu/) in a timely manner.

Academic Integrity: To create a positive and collaborative learning environment, I trust everyone in this class to act with integrity. To enhance this trust, I will not be specifically looking for academic integrity violations. However, if a violation comes to my attention, it will be subject to Northwestern and McCormick's policy on academic integrity, as found here http://www.northwestern.edu/uacc/.

In this class, you will be writing software and likely finding help from your peers or on the internet. You are encouraged to help each other and are permitted to use external websites to find information. However, any code adapted or copied from an external source must be cited in the source code.

Generally, you should use comments to indicate which code originated externally and where it came from. If the code came from a website, cite its URL and the date of access. If the code came from a peer, cite the person's name. If you are unsure whether something needs to be cited, you can either ask me or cite it just to be certain.