# 2018 Syllabus

<table>
<thead>
<tr>
<th>Instructors</th>
<th>Pito Salas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule</td>
<td>Monday, 9am to 11:50, in</td>
</tr>
<tr>
<td>Location</td>
<td>Goldfarb Vershbow computer classroom.</td>
</tr>
<tr>
<td>On-demand office Hours</td>
<td><a href="http://www.calendly.com/pitosalas/ftf">http://www.calendly.com/pitosalas/ftf</a></td>
</tr>
<tr>
<td>Prerequisites</td>
<td>Cosi12b, 21a, and Juniors and above</td>
</tr>
<tr>
<td>Expectations</td>
<td>Success in this 4 credit hour course is based on the expectation that students will spend a minimum 10 hours outside of the weekly 3 hour meeting on learning, coding, and doing other work.</td>
</tr>
<tr>
<td>Email contact:</td>
<td><a href="mailto:pitosalas@brandeis.edu">pitosalas@brandeis.edu</a></td>
</tr>
<tr>
<td>Office</td>
<td>Volen 134</td>
</tr>
</tbody>
</table>

## Course Description

Robots are everywhere, working quietly behind the scenes in labs and factories, on highways and in the home and now constantly on the front page. It is a rich area of Computer Science Research and at the same time a challenging arena of Applied Computer Science and Engineering. This makes for a very attractive context for learning.

This class provides students the opportunity to collectively work together towards a practical long term goal, to be able to deliver packages around campus with the help of an autonomous robot. Overall, the course is centered around researching and coding under the ROS framework. Instead of having conventional lectures, class time will be used to bounce ideas off of one another and to decide on a plan of action. As long as the overall goal is kept in mind, students will be able to try out and experience using numerous different software and hardware solutions.

Learn about the behavior of robotics in real time situations, following the long term development of the Brandeis Campus Rover. Applications of implementation of
theory to a real environment. Coding primarity in Python. Concepts of Arduino, ROS, probability, localization, navigation, and more are surveyed on, with a deeper focus in the second half of the semester on relevant topics of choice. Learn to work in a team and demonstrate results in an engineer-like setting. Learn how to approach problems in a meaningful way and tackle emerging technologies. Usually offered every semester.

This new experimental course will use a lecture/teamwork/lab format, currently with a single weekly meeting combined with extensive out-of-classroom team and lab work. Students taking this course will be challenged to do a lot of independent research and problem solving in quite a technical area. While there are lectures, the bulk of the work is working with a few teammates. You will need to be resourceful and persistent.

Learning Objectives

The purpose of this course is to improve students’ understanding of the computer science and engineering of robotics, getting exposure to some of the big ideas, algorithms and structures that come into play. Students will work in small teams and with little guidance to investigate how to address certain challenges, and be highly self-sufficient and motivated to drive projects to their conclusion.

Outcomes

- Students will be able to demonstrate understanding of how a simple wheeled robot works, how to control it through software, learning some key algorithms along the way, and implementing the code for a series of more and complicated challenges.

- Students will learn to program the all-important “Robot Operating System”, ROS, which is the leading real-time distributed operating system for research and industrial robotics. They will demonstrate understanding of the key concepts of ROS, nodes, topics, commands and services.
• Students will demonstrate effective working in teams, designing new algorithms and solving problems of navigation and robotics, brainstorming, collaborating, implementing, testing and demonstrating the results of their work.

• Students will learn and demonstrate professional and agile software engineering processes, including writing elegant, readable, documented code, working in rapid iterations, each with a goal and a demo, and performing weekly standups.

Course Structure

This course is still in an experimental phase and so you should expect that the content of the course will evolve as the semester progresses. With only one meeting a week, a lot if not most of the work will happen outside that meeting, with students working independently and in teams. There is a very high expectation and reliance on students’ independence, initiative, self-motivation, and problem solving skills.

Week 1-3

Introduction to robotics with mini lectures covering some of the basics. Students will form teams of 3 students to learn how to program and control a simple Robot. Programming a “simple robot” is not simple, because once your software is interacting with the ‘real world’, odd things happen (see * Martin - Real Robots Don’t Drive Straight).

Students will experiment individually and with their team to write programs for the mBot robot and think through how to deal with sensors and motors and process the noisy information that comes at their programs. This will culminate in a competition among the teams to write software to have their mBot’s traverse mazes of growing challenge. The goal for this section at a high level is to develop a visceral understanding of the noise, inaccuracies, environmental effects and non-reproducability of naive approaches to programming a robot.

Week 4-6
Mini lectures on ROS the Robot Operating System combined with two online courses in autonomous vehicles, localization and mapping. The goal here is for students to really understand how to program in ROS and understand the subtleties of localization and obstacle detection without yet resorting to advanced algorithms like SLAM.

Week 7-13

We break into three teams, Inside Navigation, Outside Navigation and Human Interaction. Each team will pursue high level objectives in each of the areas. They will begin by reading some of the seminal papers in their area and share their learnings with the class. They will organize themselves to decide what steps towards their objectives they will take and how they will organize their work.

Course Logistics

Grading

Grading will be based on the following:

- ~35%: Participation: Engagement with the course, presence in classroom discussions, following up on commitments and obligations, contributing in the further design of the course with ideas and content. This will be assessed by the instructor’s personal observations.

- ~30%: Individual work contributing to programming projects and team projects, documentation of this work in the lab notebooks, demonstrating seriousness and independence. This will be assessed by reviewing the lab notebooks, personal observation and peer and self assessments.

- ~35%: Final Deliverables: There will be a well defined, team based, final deliverable in the form of code and documentation. This will be assessed via a rubric.

Team Work
Much of the work will be in teams. Some of the steps will have teams of 2 or so students, but the bigger ones will have teams of 5 or maybe even all students together.

**Prerequisites**

This course is a “structured independent study.” We will accept up to 8 students, with some restrictions. You need to be a Junior or above, and have done well in Cosi 11a and Cosi12b and have a proven aptitude as a programmer, independent worker, and good team member. If you want to apply, just send an email to me pitosalas@brandeis.edu.

**Course Materials**

1. [Podcast about Self-Driving Deep Learning](#) - A really good background about applying deep learning to autonomous navigation. Just 1 hour.

2. [Programming Robots with ROS](#) - This is an excellent book that introduces ROS from the bottom up. You should get a copy. Beware, ROS is pretty complicated. It’s real-time, distributed operating system which is installed both on the robot and on the controlling laptop (if there is one). It can be quite difficult to configure and finnicky. But it is very very powerful!

3. [Artificial Intelligence for Robotics](#) - A fantastic introduction to the basics of SLAM and localization, doing some of the elementary mathematics that is the foundation of this core technique in navigation.

4. Papers selected by instructor. [Selection of key papers](#)

**Change Policy**

The instructor reserves the right to make changes to this syllabus and the associated curriculum web site if he deems it necessary. Any changes will either be announced in class or through e-mail. All students are responsible for finding out about such changes. Each student must be aware that not all assignments are listed in the
syllabus. Students must use their common sense and not look for loopholes in the syllabus because, ultimately, the instructor has the final say in all matters. If you are confused on any assignment, ask the instructor for clarification.

By deciding to stay in this course, you are agreeing to all parts of this syllabus. In fairness to everyone, the syllabus must apply equally to all students without exception.