6.088/6.084 – Robotics Project
Subject Information and Syllabus

Staff:

Prof. Daniela Rus (Course Coordinator) , Rm 32-374, rus@csail.mit.edu, x8-7567
Dr. Nikolaus Correll, Rm 32-375, nikolaus@csail.mit.edu, x3-6292
Teaching Assistant: Alex Patrikalakis, amcp@mit.edu, (617) 795-2277
Course Administrator: Marcia Davidson, marcia@csail.mit.edu, 32-374A, x3-6583

Class meetings:

Lectures: Fridays 10-11:30am, Rm 32-124.
Labs: Mondays and Wednesdays from 2-4pm, T-party area, 32-3xx. You will need card access to Stata. Please email course administrator with your MIT id if you do not have one already.

Course web site:

The RSS web site is `http://courses.csail.mit.edu/6.088/`.

Course Designations:

- **Units:** 12, 2-4-6 (Lectures: 2; Labs: 4; Out-of-class: 6)
- **EECS Department Engineering Concentration:** 6.088 is a 12-unit subject that counts as an Engineering Concentration under the Artificial Intelligence heading.
- **EECS Department Undergraduate Advanced Project:** 6.084 is a 6-unit subject that pairs with 6.UAP. Completing this will give you credit for an undergraduate advanced project plus 6 additional units. If you register for 6.084 you must also register for 6.UAP

Assignments:

- There will be two introductory labs.
- There will be an intensive *challenge* project to create a gadening robot system. The project will be done in teams; each team will tackle a different aspect of the challenge. The deliverables for the challenge project consist of a team challenge proposal, design reviews, a team implementation, a team presentation and demonstration, and an individual final report. Each challenge proposal, presented as a team, frames the team's attack on the posed design problem. The design reviews will be in-class presentations. The purpose of the design reviews is (1) to ensure progress to the project goal following the spiral philosophy and (2) to enable coordination for later integration of team solutions. The implementation is the delivered hardware and software produced by the students over the challenge period. The presentation and demonstration consist of the students describing their approach to the challenge, demonstrating the operation of their implemented design, and discussing its performance. The final report is written individually, and consists of each student's reflections on the challenge project and his or her contribution to the team's effort.
- There will be an integration project that will combine the team solutions into a working gardening robot system. The deliverables will be design reviews, a group implementation and demonstration, extensive performance evaluation, and a class paper. If results are substantive, the class paper will be submitted for publication at an international robotics conference.

Exams:

There will be no midterm or final exam.
The class a final project due the last week of classes, and a final report in the form of a paper due on the last day of classes.

Grading Criteria:

Subject grades are formed from a weighted average as follows:

- Lab reports: 10% (5% Technical, 5% Presentation)
- Team Project design and proposal: 20% (15% Technical, 5% Presentation)
- Team Project implementation and performance: 50% (40% Technical, 10% Evaluation)
- Team Project presentation: 10% (5% Technical, 5% Presentation)
- Final Integration: 10% (5% Technical, 5% Presentation)

Additional policies

Collaboration is encouraged for all assignments. Within teams, teamwork is an absolute necessity, and we expect that teams will work together to generate the technical content of each lab report.

Across teams, we encourage collaboration and discussion. You must explicitly credit any appropriation of code, data, or writing across teams or from other Web sources.

For the final project, full collaboration across class on all aspects of the challenge is necessary. Every student will be expected to contribute a roughly equal share to the design, implementation, evaluation, writing, and presentation of the project.

Should you require any clarification of the policies above, contact a member of the course staff.

Resources:

There will be occasional readings distributed in the form of course notes and papers.

Some other excellent books you should consider for your reference library on robotics are:

- Robot Motion Planning, Latombe, Kluwer Academic Publishers.
- Mobile Robots, Inspiration to Implementation, Jones & Flynn, A. K. Peters.
- Artificial Intelligence, A Modern Approach, Russell & Norvig, Prentice Hall.
- Behavior-Based Robotics, Arkin, MIT Press.
- Robotic Explorations, Martin, Prentice Hall.
- Computational Principles of Mobile Robotics, Dudek and Jenkin, Cambridge University Press.

Programming Languages:

Robots can be programmed in many different languages. In this class, we require that you submit your assignments in Java or C. Additionally, we support only the Linux operating system in this class. If you haven't had a lot of Java exposure you might find one or more of the following books helpful:

- David Flanagan. Java in a Nutshell, 4th edition, O'Reilly, 2002. A reference book rather than a tutorial. Succinct but covers a lot. Assumes knowledge of a language like C. Details at http://www.oreilly.com/catalog/javanut4/.
- Joshua Bloch. Effective Java: Programming Language Guide, Addison Wesley, 2001. The Bloch book explains, in about 60 short items, some key ideas in program style, as well as some subtleties of Java; it's perhaps better appreciated when you have some familiarity with Java and want to delve deeper. Both books are available at Quantum Books.
- Ivor Horton. Beginning Java 2 – JDK 1.4 Edition, Wrox Press, 2002. Tutorial introduction to all parts of Java, including user interface libraries. No knowledge of other languages is assumed.
- Ken Arnold, James Gosling, and David Holmes. The Java Programming Language, 3rd edition, Addison-Wesley, 2000. A brief explanation of Java. Assumes more background; much less explanation about how to use Java's features. User interface libraries not discussed.
- James Gosling, Bill Joy, and Guy Steele. The Java Language Specification. The official reference for Java by its inventors. Good for reference, but not an easy way to learn Java. Available as a book, or online at http://java.sun.com/docs/books/jls/index.html.
- Bruce Eckel. Thinking in Java, 3rd edition, Prentice-Hall, 2002. Also available on-line at Mindview.net (but don't try printing it yourself – it's over 1000 pages long!). Written for someone who can already program but isn't familiar with Java or object-oriented programming notions. Goes into lots of detail on tricky aspects like GUIs, multithreading, and remote method invocation.
- Computer Networks, Andrew Tanenbaum
- Robot Modeling and Control by Mark W. Spong, Seth Hutchinson, M. Vidyasagar
- C Programming Language (2nd Edition) (Prentice Hall Software) (Paperback) by Brian W. Kernighan, Dennis M. Ritchie
- Learning OpenCV: Computer Vision with the OpenCV Library (Paperback) by Gary Bradski, Adrian Kaehler

Schedule of Lectures and Lab Exercises

F	09/05	Lecture: Introduction and Brief History of Robotics	
M	09/08	Lab 1: Introduction to Course Hardware: Robot, Camera, and Hand	
W	09/10	Lab 1: Continue	
F	09/12	Lecture: Kinematics	
M	09/15	Lab 1: Continue	
W	09/17	Lab 1: Due and Presented Lab 2: Introduction to Course Hardware: Networking	
F	09/19	Lecture: Introduction to Visual Servoing	
M	09/22	Lab 2: Continue	
W	09/24	Lab 2: Due and Presented	
F	09/26	Lecture: Introduction to Networking	
M	09/29	Lab: Project Introduction and Team Selection	
W	10/01	Lab: Project Design and Planning	
F	10/03	Lecture: Project Proposals Presentation	
M	10/06	Lab: Project Development	
W	10/08	Lab: Project Development	
F	10/10	Lecture: Object Recognition (Torralba)	
M	10/13		No Class: Columbus Day
W	10/15	Lab: Project Development	
F	10/17	Lecture: Design Review.	
M	10/20	Lab: Project Development	

W	10/22	Lab: Project Development	
F	10/24	Lecture: Networked Robotics	
M	10/27	Lab: Project Development	
W	10/29	Lab: Project Development	
F	10/31	Lecture: Design Review	
M	11/3	Lab: Project Evaluation	
W	11/5	Lab: Project Evaluation	
F	11/7	Lecture: Two Case Studies	
M	11/10		No Class: Veteran's Day
W	10/12	Lab: Project Demonstration	
F	11/14	Lecture: Project Presentations	
M	11/17	Lab: Integration Planning	
W	11/19	Lab: Integration	
F	11/21	Lecture: Design Review for Integration	
M	11/24	Lab: Integration	
W	11/26		No Class: Thanksgiving
F	11/28		No Class: Thanksgiving
M	12/1	Lab: Integration	
W	12/3	Lab: First Integration Demonstration	
F	12/05	Lecture: Project Presentations	
M	12/08	Lab: Integration Evaluation	

W	12/10	Final Project Demonstration Class Paper Due	FINAL EXAM WEEK 12/15-19

2008 Subject Description

This course introduces students to advanced concepts, principles, and algorithms in robotics and embedded systems. This is a project course framed as addressing a grand challenge: to create a robotic gardening system. Solving the grand challenge requires designing and programming robots or embedded systems that interact effectively and autonomously with the real world. Students will learn about the state of the art in robotics and integrate and put to use theoretical knowledge from this course or earlier in the curricula.

Topics covered are: control, motion planning; state estimation; kinematics and inverse kinematics, computer vision, visual servoing, mesh networking, and networked control of multi-robot systems.

Students will develop a distributed gardening robot system. The plants in the garden will be potted cherry tomatoes. Each pot will contain a soil sensor that will be networked with the robots in the system via mesh networking provided by OpenMesh running BATMAN. The robots will be based on iRobot iCreate platforms extended with a SES robot arm from Lynxmotion, custom watering system, eye-in-hand camera, and OpenMesh networking. The robot is controlled using a notebook PC.

The course will be structured with a lectures and laboratories. The lectures will be used to present concepts and algorithms for the course topics. Students will also do design review presentations during lectures. The laboratories will be used to develop and implement the course challenge project.

Students will work in teams to design an integrated solution to the gardening robot system. Students will be introduced to the course platforms and system infrastructure in the first lab. During subsequent labs, students will work in small teams to develop implement and evaluate a robust solution to one component of the project. The final module of the course will integrate all the components and evaluate performance.

6.088/6.084 Learning Objectives:

Students completing 6.088/6.084 will be able to:

1. **Specify** the requirements for an integrated hardware and software design and implementation of an autonomous system performing a specified task;
2. Critically **evaluate** choices of design and architectures;
3. **Use** kinematics, planning, control theory, visual servoing, and networked control to implement controllers, estimators and planners that satisfy the requirements of specified task:
4. **Operate** the system for an extended and specified time;
5. **Communicate** the results of the project design process and the key aspects of the overall project (from concept to end goal).

6.088/6.084 Measurable Outcomes:

Each of these outcomes corresponds to one or more deliverables in the course.

1. An integrated hardware-software system that performs the desired task;
2. Lab briefings that demonstrate mastery of key design skills;
3. Development and delivery of an oral presentation suitable for a professional audience;

4. Completion of a conference-level technical paper that introduces problem, presents the method, analyzes the design and its success or failure, and reflects upon lessons learned and future directions.