Lectures:	Tuesday and Thursday 7:30-8:50pm Van Dyck Hall 211 - Location: `https://go.rutgers.edu/afmrtd56`
Instructor:	Edgar Granados - *eg585 at cs.rutgers.edu* Office Hours: Thursday 2pm Office: 1 Spring St, New Brunswick, NJ 08901 Office #331 (3rd floor) Zoom Link: `https://go.rutgers.edu/jklwrx8g`
TA:	Patrick Meng - pm708 at scarletmail.rutgers.edu Office Hours: Thursdays 11am Office: Office #305 (3rd floor) Zoom Link: `https://go.rutgers.edu/3plofj17`
Discord:	`https://discord.gg/qaVa2RPQ`
Google Signup Link:	`https://go.rutgers.edu/s1g1bbae`

Outline of the Course: This course provides a general introduction to robotics from a computational perspective covering AI foundations of mobile robots and robotic arms. It will provide a view of robots as autonomous agents with a mechanical embodiment, which must observe and act upon their surroundings through the iterative execution of a sensing-planning-actuation loop. For planning and decision-making, the course will introduce motion planning methods ranging from discrete search (e.g., A* and D*) to sampling-based (e.g., PRM and RRT) and optimization algorithms along the underlying foundations, such as the configuration space abstraction. Extensions to planning for systems with significant dynamics (e.g., vehicles at high velocities or fixed-wing flying vehicles) or high-dimensional articulated systems (e.g., robotic arms, closed kinematic chains) will be covered depending on time availability. On the sensing and perception side, the course will cover state estimation, such as robot localization and mapping, for which Bayesian solutions (e.g., Kalman and particle filters) as well as optimization tools will be described.

Source Material: This course will draw upon material from multiple textbooks.

- "Principles of Robot Motion: Theory, Algorithms, and Implementations" (PRM) by Howie Choset, Kevin M. Lynch, Seth Hutchinson, George A. Kantor, Wolfram Burgard, Lydia E. Kavraki, Sebastian Thrun.
- "Probabilistic Robotics" [ProRob] by Sebastian Thrun, Wolfram Burgard and Dieter Fox.
- "Introduction to Autonomous Mobile Robots" [IAMR] by Roland Siegwart, Illah Reza Nourbakhsh and Davide Scaramuzza.
- "Planning Algorithms" (PA) [book download] by Steve LaValle.
- "AI: A Modern Approach" (AIAMA) by Stuart Russell and Peter Norvig.
- "Computational Geometry: Algorithms and Applications" (CGAA) by Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars.
- "Modern Robotics: Mechanics, Planning, and Control" (MR) [book download] by Kevin M Lynch and Frank C. Park

When necessary, information will be provided on how to acquire legal digital copies of reference textbook material related to the sources above. Only portions of the above books will be used.

Students will be greatly benefited by familiarity and experience with basic algorithms and data structures, as well as a grounding in probability and calculus. Programming aptitude in Python (or alternatively an alternative modern programming language, such as C++) is expected.

Tentative Schedule of Lectures:

- 09/03 - Lecture 0: Introduction
- 09/05 - Lecture 1: Configuration Space
- 09/10 - Lecture 2: Lie Groups: Rotations and Poses
- 09/12 - Lecture 3: Rigid Body Motions I
- 09/17 - Lecture 4: Rigid Body Motions II
- 09/19 - Lecture 5: Robot Models
- 09/24 - Lecture 6: Introduction to Motion Planning
- 09/26 - Lecture 7: Path Planning as Classical Search
- 10/01 - Lecture 8: Planning among Polygonal Obstacles: Visibility Graph
- 10/03 - Lecture 9: Combinatorial Planners: GVDs and Trap. Decompositions
- 10/08 - Lecture 10: Sampling Based Motion Planning: PRM & RRT
- 10/10 - Lecture 11: Implementation of Sampling-based Planners
- 10/15 - Lecture 12: Properties of Sampling-based Planners
- 10/17 - In-class test
- 10/22 - Lecture 13: Potential Functions
- 10/24 - Lecture 14: Coupled and Decoupled approaches
- 10/29 - Lecture 15: Linear Quadratic Regulator
- 10/31 - Lecture 16: Nonlinear Optimization
- 11/05 - Lecture 17: Trajectory Optimization
- 11/07 - Lecture 18: Model Predictive Control: iLQR
- 11/12 - Lecture 19: Robust Methods
- 11/14 - Lecture 20: Sensors & basic concepts in probability
- 11/19 - Lecture 21: Factor Graphs
- 11/21 - Lecture 22: Localization
- 11/26 - Lecture 23: Mapping
- 11/28 - Thanksgiving
- 12/03 - Lecture 24: SLAM
- 12/05 - Lecture 25: TBD
- 12/10 - Lecture 26: Review of Material

There will be a Test during the finals week (date/time to be announced).

Tests: Note, that per the above schedule, there will be 2 in-class test:
The first test will take place during the regular lecture time. A missed test draws zero credit for that test. The last test takes place during the university scheduled Final Exam slot. Each test will include opportunities for extra credit (e.g., the perfect score will be 100 but there will be questions totaling in the order of 110 points).

Emergencies will be considered upon submitting a University-issued written verification to the Instructor; for assistance contact your Dean's Office. Also, check the definition of Final Exam Conflicts by SAS and Rutgers.

Assignments: There will be multiple assignments, which will be primarily programming-focused but will also include some practice theoretical questions, which are intended to assist the student in mastering the course content. Some of the questions in the assignments will focus on the design of your code and experimental evaluation. Some of the questions will ask to execute solutions to different problems and collect data. Typically, each assignment will include opportunities for extra credit.

You are strongly encouraged to use Python for the programming assignments. Standard Python libraries are allowed to be used as well as Numpy and Matplotlib. Any example code that may be provided by the TA will be in Python. In general, you are free to use outside libraries for supplemental parts of the projects (graphics, etc). When it comes, however, to the algorithms that are the focus of this course, you are to implement these algorithms in code yourself. Helper functions and existing libraries can be used as a scaffolding to build your code off of, but your final implementation should be your own code.

You can alternatively code in the language of your choice, so long as it is not significantly obscure and does not complicate the grading process. It will not be possible, however, to provide debug information or example code in other programming languages than Python.

The programming assignments will primarily involve modeling a robot's motion and coding robot planning and perception algorithms as well as evaluating their efficiency given the modeling tool developed in the process. Typically you will be asked to submit an electronic version of your code, test runs and a typeset report.

The tentative plan for assignments is the following one:

- Assignment 1 - Setting up the software infrastructure (15%).
- Assignment 2 - Collision-free motion planning (15%).
- Assignment 3 - Control of a robotic system (15%).
- Assignment 4 - Localization of a robot system moving among obstacles using sensor (15%).

No late submissions will be allowed barring exceptional circumstances.

Assignments can be completed either by individual students or teams of 2 students. You will be expected to indicate the contributions of all members of your group. Particular emphasis will be placed on clearly communicating your design and implementation choices in the report of the assignment.

If issues arise during the collaboration between the two team members (e.g., non-responsive team members), the students should inform the TA or instructor as early as possible. Each student, however, is responsible for the submission of a complete assignment by the due date in order to receive a grade.

If you are unable to find somebody to work with early, first post on Discord, and as a last resort, e-mail the TAs. If you choose to work alone, our expectations for the amount of work are the same as for teams of 2.

Larger teams will not be allowed. Submission of an assignment by more than 2 students will result in zero credit for that assignment to all the corresponding team members.

Beyond other extra credit opportunities, students will receive 6% extra credit if they typeset in LaTeX their assignment reports. For instance, if a team was to receive a score of 62/100 and they provide a typeset homework, then their score will be 66/100, i.e,. 62 points +6% of 62 points (3.72 bonus points rounded to 4). Resources on how to use LaTeX will be provided by the instructor on Canvas. If teams use a work editor, e.g., MicroSoft Word, for their reports they will receive a 3% bonus instead.

Grading: The course evaluation will be primarily based on assignments (60% of the grade) and in-class tests (40% of the grade). Extra credit will typically be available as part of the assignments and the tests.

For students that are on the casp of potentially getting a higher letter grade, a participation bonus can be considered (up to 5% bonus of the total number of points collected, assigned at the discretion of the instructors and TAs). This bonus will depend on the level of participation in lectures and recitations. By default your bonus participation grade is 0, i.e., if you typically come to the lectures/recitations but you do not actively participate during the lectures or the recitations (e.g., no questions by the student or no answers to questions posed by the instructor), your participation bonus will be 0.

The mapping of scores to letter grades will be determined at the end of the semester. As a **rough** guide, the following rule may be used for the final grade **(it may be adapted close to the end of the semester)**:

Final Grade	A	B+	B	C+	C	D	F
Range	>= 90	80-89	70-79	60-69	50-59	40-49	≤ 39

Questions about Grading: If you have a question or complaint regarding the points you received on specific parts of an assignment, or an exam, submit a request to the TA, stating specifically but very briefly what parts of that assignment/exam you wish to be reviewed. Please refrain from verbal arguments about grades with the instructor or with any of the TAs. The deadline for submitting such requests is the last lecture.

Academic Integrity: Tests are to be treated as individual efforts. Assignments are not to be treated as collective efforts beyond the participation of the group members! Discussions between teams are not allowed on how to solve specific questions in assignments. Do not discuss assignments with students that are not currently taking the class.

A severe penalty will be given to any assignment which indicates collusion or cheating. The usual penalty for cheating on an assignment or an exam is failure in the course. At a minimum your grade in the corresponding exam/assignment will be reduced. Using another team's listing or having another person "ghost write" an assignment will be considered cheating.

Turning in work without properly citing the sources of the content included in your work is plagiarism. All kinds of sources are included in this definition, even those downloaded from the web (in which case an operable link must be cited) or verbatim software extracted from Large Language Models, such as ChatGPT. Plagiarism from the web or other sources is considered cheating and has the same effects to those mentioned above. Even with a reference, submitting an answer to a homework question, verbatim from any source and without any contribution on your part, draws zero credit.

You should carefully study the website of Rutgers University on Academic Integrity (`http://academicintegrity.rutgers.edu/`) and the corresponding policy, as well as the corresponding website from the department of Computer Science (`https://www.cs.rutgers.edu/academic-integrity/introduction`). Your continued enrollment in this course implies that you have read these policies, and that you subscribe to the principles stated therein.

Disability Accommodations Students in need of disability accommodations can register for the corresponding accommodations and consult the policies and procedures of the Office of Disability Services website: `https://ods.rutgers.edu`

Availability of Support If, at any point, you experience anything impacting your performance or ability to participate in this class, please reach out to the instructor. Please also see the academic, health, and mental wellness resources on the syllabus as well as others searchable at `https://success.rutgers.edu/` for further support. Other helpful resources:

- Student Support Services: `https://www.rutgers.edu/academics/student-support`
- The Learning Centers: `https://rlc.rutgers.edu/`
- The Writing Centers (including Tutoring and Writing Coaching) `https://writingctr.rutgers.edu`
- Rutgers Libraries: `https://www.libraries.rutgers.edu/`
- Office of Veteran and Military Programs and Services: `https://veterans.rutgers.edu`
- Student Health Services: `http://health.rutgers.edu/`
- Counseling, Alcohol and Other Drug Assistance Program & Psychiatric Services (CAPS): `http://health.rutgers.edu/medical-counseling-services/counseling/`
- Office for Violence Prevention and Victim Assistance: `www.vpva.rutgers.edu/`