MARYLAND APPLIED
GRADUATE ENGINEERING

ENPM808X Software Development for Robotics (0101, RO01)

Term: Fall 2023
Professor: Dr. Tommy Chang
Pronouns: He/Him   **Teaching Assistant:** Jay Prajapati
Office Phone: N/A   **Pronouns:** He/Him
Email: chang177@umd.edu   **Email:** jayp@umd.edu
Office Hours: **Office Hours:** Thursdays 2:30pm - 3:30pm
        Location: Virtual
        Date and times: Mondays, 5:30pm - 6:30pm
        and by appointment
Credits: 3
Course Dates: From August 28th – December 19th
Course Times: Tuesday 7:00-9:40 pm and Online
Classroom: JMP 2121

Canvas/ELMS link: https://umd.instructure.com/courses/1348107

Course Description

As the robotics industry continues to grow and evolve, software's role in these products and systems is also becoming more critical. From embedded controls to advanced perception and learning, software permeates today's robots. Building off domain expertise developed in other robotics courses, this course teaches the tools and processes to develop professional quality software for deployed systems and products. Students will learn the best practices of taking new ideas or prototypes, and understanding what it takes to build the complex software that is so important to today's commercialized robotic systems. The course is split into two parts: the first will review the C++ programming language, object-oriented programming (OOP) concepts, version control, testing, and agile software development processes; the second will introduce the popular Robot Operating System (ROS) framework with intensive programming assignments and projects. Students should be proficient in using Linux, along with programming skills in C/C++ and a solid understanding of object-oriented programming concepts.

Prerequisites

- ENPM809Y (Introductory Robot Programming) or equivalent.
- Familiarity with Linux and the command-line.
- Proficiency in C/C++ and object-oriented programming.

Learning Outcomes

After successfully completing this course you will be able to:

- Meet the rising expectations of robotics software engineers in industry and government.
- Apply industrial best practices to code, build, test, and maintain large-scale C++ software.
- Design software projects using UML diagrams and employ good Object-Oriented Programming (OOP) practices and design patterns.
- Apply agile development, pair programming, and test-driven development (TDD) approaches to software implementation.
- Design appropriate software architecture and debugging techniques for real-time robotic systems.
- Build a simulated robotic system using ROS and Gazebo.

- Utilize the Linux shell effectively and write competent Bash scripts – two important and desirable skills for both robotics software engineers and Linux system engineers.

Course Materials

Required Resources
- Book:

  - **Required**: Software Engineering: The Current Practice, Vaclav Rajlich. CRC Press, 2011. (ISBN 9781439841228). Estimated cost = $45

  - **Recommended**: Game Programming Patterns, Robert Nystrom. Genever Benning, 2014. (ISBN 9780990582908). Content available for free on author's website http://gameprogrammingpatterns.com/contents.html. Estimated cost = $0

  - **Recommended**: A Concise Introduction To Robot Programming With ROS2, Francisco Martín Rico. Chapman and Hall/CRC, 2022. (ISBN 9781000629811). Estimated cost = $45

- Application/Software:

  - **Required**: Ubuntu Linux 22.04, ROS2 Humble Hawksbill, Gazebo, CMake, g++ compiler, ccache, gprof, valgrind, clangd, bear, and any C++ IDE / editor that supports Language Server Protocol (LSP) and runs on Linux. Popular free editors include Visual Studio Code, Eclipse, gedit, Emacs, Vim, etc. See a list of LSP clients at https://langserver.org/#implementations-client.   Estimated cost = $0

- Total Estimated costs of required course materials: $45

Supplemental Resources (no purchase required)
- Book:
  - **Reference**: Professional C++ (5th Edition), Marc Gregoire. Wrox, 2021. (ISBN 9781119695400).

  - **Reference**: Design Patterns in Modern C++: Reusable Approaches for Object-Oriented Software Design, Dmitri Nesteruk. Apress, 2018. (ISBN 9781484236024).

- Application/Software:
  - **Optional**: Docker, Windows Subsystem for Linux (WSL), VcXsrv, VMware (Free download from Terpstore: https://terpware.umd.edu/search?q=vmware).

Course Structure
This course includes both on-campus and online sections. To attend synchronously online, log into ELMS-Canvas at the time of the Section 0101 class **[Tuesday/7pm]** and select "Video Conference" from the left side menu. This will open a Zoom link to the live classroom.

For asynchronous online students, all lectures will be recorded and made available on ELMS-Canvas under "Panopto Recordings/Video Lectures" within 24 hours of the class time. Be sure to review the recorded lecture in a timely manner.

On-campus students come to class prepared to engage with the lecture and materials. Online students, be sure to log into Canvas regularly and participate in discussions and activities. Regardless of the section you are enrolled in, participation is expected.

Please note that F1 students enrolled in the on campus section are required to attend in person. If you have a conflict on a particular day, please reach out to me in advance to discuss.

Communication Guidelines

Communicating with the Instructor
My goal is to be readily available to you throughout the semester. I can be reached by email at **chang177@umd.edu**. Please DO NOT email me with questions that are easily found in the syllabus or on ELMS-Canvas (e.g., When is this assignment due? How much is it worth? etc.), but please DO reach out about personal, academic, and intellectual concerns/questions.

While I will do my best to respond to emails within 24 hours, you will more likely receive email responses from me on Mondays, Wednesdays and Fridays from 8:00pm-10:00pm EST.

When constructing an email to me please put "ENPM 808X: Your Topic" in the subject line. This will draw my attention to your email and enable me to respond to you more quickly.

Additionally, please review [These tips for 'How to email a Professor'](#). By following these guidelines, you will be ensured to receive a timely and courteous response.

Finally, if you need to discuss issues not appropriate for the classroom and/or an email, we can arrange to talk by phone, over Zoom, or in person. Send me an email asking for a meeting and we can set something up.

Announcements
I will send IMPORTANT messages, announcements, and updates through ELMS-Canvas. To ensure you receive this information in a timely fashion, make sure your email and announcement notifications (including changes in assignments and/or due dates) are enabled in ELMS-Canvas ([How to change notification settings in CANVAS](#)).

Log into our ELMs-Canvas course site **at least** once every 24-hour period to check your inbox and the Announcements page.

Names/Pronouns and Self-Identifications
The University of Maryland recognizes the importance of a diverse student body, and we are committed to fostering inclusive and equitable classroom environments. I invite you, if you wish, to tell us how you want to be referred to in this class, both in terms of your name and your pronouns (he/him, she/her, they/them, etc.). Keep in mind that the pronouns someone uses are not necessarily indicative of their gender identity. Visit trans.umd.edu to learn more.

Additionally, it is your choice whether to disclose how you identify in terms of your gender, race, class, sexuality, religion, and dis/ability, among all aspects of your identity (e.g., should it come up in classroom conversation about our experiences and perspectives) and should be self-identified, not presumed or imposed. I will do my best to address and refer to all students accordingly, and I ask you to do the same for all of your fellow Terps.

Communicating with your Peers
With a diversity of perspectives and experience, we may find ourselves in disagreement and/or debate with one another. As such, it is important that we agree to conduct ourselves in a professional manner and that we work together to foster and preserve a virtual classroom environment in which we can respectfully discuss and deliberate controversial questions. I encourage you to confidently exercise your right to free speech—bearing in mind, of course, that you will be expected to craft and defend arguments that support your position. Keep in mind, that free speech has its limit and this course is NOT the space for hate speech, harassment, and derogatory language. I will make every reasonable attempt to create an atmosphere in which each student feels comfortable voicing their argument without fear of being personally attacked, mocked, demeaned, or devalued.

Any behavior (including harassment, sexual harassment, and racially and/or culturally derogatory language) that threatens this atmosphere will not be tolerated. Please alert me immediately if you feel threatened, dismissed, or silenced at any point during our semester together and/or if your engagement in discussion has been in some way hindered by the learning environment.

Netiquette Policy
Netiquette is the social code of online classes. Students share a responsibility for the course's learning environment. Creating a cohesive online learning community requires learners to support and assist each other. To craft an open and interactive online learning environment, communication has to be conducted in a professional and courteous manner at all times, guided by common sense, collegiality and basic rules of etiquette.

Grading

Grade Breakdown

Assignment	Breakdown	Final Percentage %
Homework :	15 homework, total about 635 points	50%
Mid-term project:	2 phases, total 600 points	20%
Final project:	4 phases, total 650 points	30%

Course Assignments

Assignments
- This course is focused on learning sound software engineering principles, **not** learning a programming language. You must have proficient programming skills to execute programming assignments while spending the majority of your time on course topics.
- There will be a total of 21 graded assignments, out of which 6 are associated with the midterm and final projects. These assignments serve a dual purpose: firstly, provide students with opportunities for teamwork, critical thinking, problem-solving, and hands-on programming experience; secondly, they equip students with valuable skills that can enhance their employability in the industry. For more information about these assignments, please refer to the section on Course Schedule.

Mid-term project
- This is a 3-week project. In the first-part, a project proposal, I will review it and provide comments before you begin working on it. Once I provide feedback and give you permission to proceed, you will have approximately two weeks to complete the rest of the project.
- You must work in a group of two or three of your choosing. The groups must implement the project as pair programming, while a group of three adds a third role: the design keeper. The design keeper extends what a navigator does in the traditional pair and makes sure that the implementation is following the project design (e.g. UML diagrams) along with good OOP practices and appropriate design patterns.

Final project
- This is a 4-week project, beginning with a proposal stage, followed by 3 weeks of development and completing with an in-class presentation. I will review the proposal and provide comments and approval.
- The final project is expected to take 40 to 80 hours to complete over the 4-week period.
- Final submission includes presentation materials. Teams will present their projects in class.
- Late submissions will not be accepted.
- Ungraded submissions will be due each week to provide feedback.
- You must work in a group of two or three of your choosing. The groups must implement the project as pair programming and follow the same approach taken in the mid-term project.

Grading Assignments

All assignments will be graded according to a predetermined set of criteria (i.e., rubric) which will be communicated to students before the assignment is submitted.

To progress satisfactorily in this class, students need to receive timely feedback. To that end, it is my intention to grade all assignments within **1 week** of their due date. If an assignment is taking longer than expected to grade, students will be informed of when they can expect to see their grade.

Grade Computation

All assessment scores will be posted on ELMS/Canvas page. If you would like to review any of your grades (including the exams), or have questions about how something was scored, please email me to schedule a time for us to meet and discuss.

It is expected that you will submit work by the deadline listed in the syllabus and/or on ELMS-Canvas. Late work will be penalized according to the late work policy described in the **Course Policies and Procedures** section below.

Grade Disputes: I am happy to discuss any of your grades with you, and if I have made a mistake I will immediately correct it. Any formal grade disputes must be submitted in writing and within one week of receiving the grade.

Final letter grades are assigned based on the percentage of total assessment points earned. To be fair to everyone I have to establish clear standards and apply them consistently. It would be unethical to make exceptions for some and not others. Your total percentage score will be rounded to the nearest integer for grade assignment purposes. For example, 89.50 will be rounded to 90 (A-), while 89.49 will be rounded to 89 (B+).

Final Grade Cutoffs

+	97%	+	87%	+	77%	+	67%	+	
A	94%	B	84%	C	74%	D	64%	F	<60%
-	90%	-	80%	-	70%	-	60%	-	

Course Schedule

Week #	Topic	Deliverable
1. **08/29/2023**	- Introduction - Create and share GitHub user account - Linux command-line tools and shell programming - Object Oriented Programming Design Patterns	- Install Linux (Native, Container, or VM) - Reading Assignments - Linux primer; What is the Bash shell? - Reading Assignment - Game Programming Patterns (Ch. 1 Introduction) - (optional) LinkedIn Learning: Linux Command Line; C++ (both are free for UMD students)
2. **09/05/2023**	- Compiling and running in a C++ IDE (any LSP client of your choice) - History of Software Engineering - Software Life Span Models - Object-Oriented Technology - Software Models	- Configure clangd to work with your favorite C++ IDE / Editor; Install ccache, bear - Compiling and Running (cpp boilerplate) - Reading Assignment - Software Engineering: The Current Practice ch. 1-4 - Exercises - Software Engineering: The Current Practice 3.13, 3.14, 3.15, 4.1, 4.2, 4.3, 4.5, 3.10 - Reading Assignment - Game Programming Patterns (Ch. 2.1 Command)
3. **09/12/2023**	- Software Changes - Concepts and Concept Localization - Work products - Dependency search	- Reading Assignment - Software Engineering: The Current Practice ch. 5, 6, 8, 9, 12, 13 - Exercises - Software Engineering: 5.1, 5.3, 5.5, 6.6, 9.4 - Reading Assignment - Game Programming Patterns (Ch. 2.2 Flyweight)
4. **09/19/2023**	- Software Verification - Initial Development - Test-Driven Development - Git Tutorial	- Reading Assignment - Software Engineering: The Current Practice ch. 10, 11,14 - Exercises: Software Engineering: 10.2, 10.3, 10.4, 10.5, 10.10, 10.11, 10.12 - Reading Assignment - Git - Test-Driven Development exercise (with groups!) - Reading Assignment - Game Programming Patterns (Ch. 2.3 Observer)
5. **09/26/2023**	- Git Review - Using testing with Google Mock	- Reading Assignment - CMake and Valgrind

	- Continuous Integration - Library Dependency - Introduction to CMake - Quad Chart, Code Coverage Report - Mid-term Phase 0 Deliverables	- Reading Assignment - Software Engineering: The Current Practice ch. 15, 16 - Reading Assignment - Game Programming Patterns (Ch. 2.4 Prototype)
6. **10/03/2323**	- Course mid-point review - Tool of the Trade - Release backlog table, Product Backlogs, Iteration Backlogs, Time / work log - Mid-term Phase 1 & 2 Deliverables	- Valgrind, gdb, and gprof exercises - Mid-term - Phase 0 (proposal) - Reading Assignment - Game Programming Patterns (Ch. 2.5 Singleton)
7. **10/10/2023**	- GitHub CI - Using Doxygen with CMake - OSS in Robotics - Software License	- Mid-term project - Phase 1 - Reading Assignment - Game Programming Patterns (Ch. 2.6 State) - **Project Due**: Mid-term project Phase 0.
8. **10/17/2023**	- Important libraries: Alternatives to Boost, Eigen, OpenCV, PCL, Qt - Model-View-Controller design pattern	- Mid-term project - Phase 2 - Reading Assignment: Model-View-Controller Explained in C++ - **Project Due**: Mid-term project Phase 1.
9. **10/24/2023**	- Introduction to ROS 2 - Installing and Running ROS 2 Docker Image	- Install ROS 2 Humble Hawksbill - Programming Assignment: ROS Publisher/Subscriber - ROSCon Presentation Summary - **Project Due**: Mid-term project Phase 2.
10. **10/31/2023**	- ROS 2 topics: Services, Launch Files, Logging, Colcon Build System - Final Project Deliverables: Plan (Phase 1), Development and Testing (Phase 2), Presentation (Phase 1-3)	- Programming Assignment - ROS Services, Logging, and Launch files
11. **11/07/2023**	- More ROS 2 topics: Handling multiple frames, Coordinate Frames Convention, Representing pose, URDF, Recording and playback with ros bag, rViz - Unit test and Integration test with ROS	- Programming Assignment - ROS 2 tf2, unit testing, bag files - Rviz Assignment - Reading Assignment - Autonomous Navigation - Final Project - Phase 0 (Proposal)
12. **11/14/2023**	- Visualization and Simulation - Gazebo - ActionLib	- Gazebo Tutorials - Programming Assignment - Working with Gazebo (continue) - Reading Assignment - Actionlib and SLAM

13.
11/21/2023
	- Advance ROS Topics: Part 1
	    - Navigation Stack Introduction
	    - FastSLAM
		- Final Project - Phase 1 (Planning)
		- **Project Due**: Final project Phase 0.

14.
11/28/2023
	- Advanced ROS Topics: Part 2
	    - Navigating a known map
	    - Localization with AMCL
		- Reading Assignment - Autonomous Navigation

	- Move_base	- Final Project - Phase 2 (Development and Testing) - **Project Due**: Final project Phase 1.
15. **12/05/2023**	- RTOS: VxWorks - Core Flight System (cFS)	- Extra Credit: Github Profile/Main Page - Final Project Phase 3 (Presentation) - **Project Due**: Final project Phase 2.
16. **12/12/2023**	Reading Day – no class	
17. **12/19/2023**	Final Project Presentation	- **Project Due**: Final project Phase 1-3.

Note: **This is a tentative schedule, and subject to change as necessary** – monitor ELMS-Canvas for current deadlines. In the unlikely event of a prolonged university closing, or an extended absence from the university, adjustments to the course schedule, deadlines, and assignments will be made based on the duration of the closing and the specific dates missed.

Course Policies and Procedures

The University of Maryland's conduct policy indicates that course syllabi should refer to a webpage of course-related policies and procedures. For a complete list of graduate course related policies, visit the [Graduate School website](#). Below are course-specific policies and procedures which explain how these Graduate School policies will be implemented in this class.

Satisfactory Performance
The Graduate School expects students to take full responsibility for their academic work and academic progress. The student, to progress satisfactorily, must meet all the academic requirements of this course. Additionally, each student is expected to complete all readings and any preparatory work before each class session, come to class prepared to make substantive contributions to the learning experience, and to proactively communicate with the instructor when challenges or issues arise.

Every effort has been made to evenly distribute the course requirements, and to support your understanding of the course material. However, it is likely that some weeks will require more effort on your part, and some material will require additional help beyond what is immediately available. Please reach out to me or your appropriate TA for these course-related questions, and please be prepared to put in the additional effort.

Questions about Assignments
Please ask all questions you may have about an assignment by 6PM the day before the assignment is due. Any questions asked after that time may not be answered in time for you to make changes to your work.

Late Work Policy
Assignments are due by the beginning of the next class lecture (unless stated otherwise) and must be submitted following the assignment instructions. Some will be submitted through Canvas while others will be submitted through GitHub.

Assignments should be completed by the due date and time listed with the assignment, on the syllabus, and/or in the course calendar. If you are unable to complete an assignment by the stated due date, it is your responsibility to contact your instructor to discuss an extension, at least 24 hours BEFORE the assignment is due. Extensions are not guaranteed, but may be granted at the instructor's discretion.

All graded assignments will be subject to a deduction of points for each calendar day they are submitted late, up to a maximum of three days late. The rate of deduction will increase by 10% with each passing day, starting at 5%. That is, the penalty for being 1 day late is 5%, for 2 days late it's 15%, and for 3 days late it's 25%. Any work submitted more than three days late will not receive feedback and will automatically receive a grade of zero. As for the final project, late submission cannot be accommodated in general due to the limited time remaining for the course.

Religious Observance
It is the student's responsibility to inform the instructor of any intended absences for religious observances in advance. Notice should be provided as soon as possible but no later than the end of the schedule adjustment period.

Academic Integrity
For this course, some of your assignments may be collected via Turnitin on ELMS/Canvas. I have chosen to use this tool because it can help you improve your scholarly writing and help me verify the integrity of student work. For information about Turnitin, how it works, and the feedback reports you may have access to, visit Turnitin Originality Checker for Students

The University's Code of Academic Integrity is designed to ensure that the principles of academic honesty and integrity are upheld. In accordance with this code, the University of Maryland does not tolerate academic dishonesty. Please ensure that you fully understand this code and its implications because all acts of academic dishonesty will be dealt with in accordance with the provisions of this code. All students are expected to adhere to this Code. It is your responsibility to read it and know what it says, so you can start your professional life on the right path. **As future professionals, your commitment to high ethical standards and honesty begins with your time at the University of Maryland.**

It is important to note that course assistance websites, such as CourseHero, or AI generated content are not permitted sources, unless the instructor explicitly gives permission. Material taken or copied from these sites can be deemed unauthorized material and a violation of academic integrity. These sites offer information that might be inaccurate or biased and most importantly, relying on restricted sources will hamper your learning process, particularly the critical thinking steps necessary for college-level assignments.

Additionally, students may naturally choose to use online forums for course-wide discussions (e.g., Group lists or chats) to discuss concepts in the course. However**, collaboration on graded assignments is strictly prohibited unless otherwise stated**. Examples of prohibited collaboration include: asking classmates for answers on quizzes or exams, asking for access codes to clicker polls, etc. Please visit the **Office of Graduate Studies' full list of campus-wide policies** and reach out if you have questions.

If you ever feel pressured to comply with someone else's academic integrity violation, please reach out to me straight away. Also, *if you are ever unclear* about acceptable levels of collaboration, *please ask*!

To help you avoid unintentional violations, *the following table* lists levels of collaboration that are acceptable for each graded exercise. Each assignment will contain more specific information regarding acceptable levels of collaboration.

	OPEN NOTES	USE BOOK	LEARN ONLINE	GATHER CONTENT With AI	ASK FRIENDS	WORK IN GROUPS
Homework Assignments	✔	✔	✔	---	---	---
Team Projects	✔	✔	✔	✔	✔	✔

Course Evaluation
Please submit a course evaluation through Student Feedback on Course Experiences in order to help faculty and administrators improve teaching and learning at Maryland. All information submitted to Course Experiences is confidential. Campus will notify you when Student Feedback on Course Experiences is open for you to complete your evaluations at the end of the semester. Please go directly to the **Student Feedback on Course Experiences** to complete your evaluations. By completing all of your evaluations each semester, you will have the privilege of accessing through Testudo the evaluation reports for the thousands of courses for which 70% or more students submitted their evaluations.

Copyright Notice
Course materials are copyrighted and may not be reproduced for anything other than personal use without written permission.

Tips for Succeeding in this Course

1. **Participate.** I invite you to engage deeply, ask questions, and talk about the course content with your classmates. You can learn a great deal from discussing ideas and perspectives with your peers and professor. Participation can also help you articulate your thoughts and develop critical thinking skills.
2. **Manage your time.** Students are often very busy, and I understand that you have obligations outside of this class. However, students do best when they plan adequate time that is devoted to course work. Block your schedule and set aside plenty of time to complete assignments including extra time to handle any technology related problems.
3. **Login regularly.** I recommend that you log in to ELMS-Canvas several times a week to view announcements, discussion posts and replies to your posts. You may need to log in multiple times a day when group submissions are due.

4. **Do not fall behind.** This class moves at a quick pace and each week builds on the previous content. If you feel you are starting to fall behind, check in with the instructor as soon as possible so we can troubleshoot together. It will be hard to keep up with the course content if you fall behind in the pre-work or post-work.
5. **Use ELMS-Canvas notification settings.** Pro tip! Canvas ELMS-Canvas can ensure you receive timely notifications in your email or via text. Be sure to enable announcements to be sent instantly or daily.
6. **Ask for help if needed.** If you need help with ELMS-Canvas or other technology, IT Support. If you are struggling with a course concept, reach out to me and your classmates for support.

Student Resources and Services

Taking personal responsibility for your learning means acknowledging when your performance does not match your goals and doing something about it. I hope you will come talk to me so that I can help you find the right approach to success in this course, and I encourage you to visit the Counseling Center's Academic Resources to learn more about the wide range of resources available to you. Below are some additional resources and services commonly used by graduate students. For a more comprehensive list, please visit the Graduate School's Campus Resources Page.

Accessibility and Disability Services
The University of Maryland is committed to creating and maintaining a welcoming and inclusive educational, working, and living environment for people of all abilities. The University of Maryland is also committed to the principle that no qualified individual with a disability shall, on the basis of disability, be excluded from participation in or be denied the benefits of the services, programs, or activities of the University, or be subjected to discrimination. The Accessibility & Disability Service (ADS) provides reasonable accommodations to qualified individuals to provide equal access to services, programs and activities. ADS cannot assist retroactively, so it is generally best to request accommodations several weeks before the semester begins or as soon as a disability becomes known. Any student who needs accommodations should contact me as soon as possible so that I have sufficient time to make arrangements.

For assistance in obtaining an accommodation, contact Accessibility and Disability Service at 301-314-7682, or email them at adsfrontdesk@umd.edu. Information about sharing your accommodations with instructors, note taking assistance and more is available from the Counseling Center.

Writing Center
Everyone can use some help sharpening their communication skills (and improving their grade) by visiting The Graduate School's Writing Center and schedule an appointment with them. Additionally, international graduate students may want to take advantage of the Graduate School's free English Editing for International Graduate Students (EEIGS) program.

Health Services
The University offers a variety of physical and mental health services to students. If you are feeling ill or need non-emergency medical attention, please visit the University Health Center.

If you feel it would be helpful to have someone to talk to, visit UMD's Counseling Center or one of the many other mental health resources on campus.

Notice of Mandatory Reporting

Notice of mandatory reporting of sexual assault, sexual harassment, interpersonal violence, and stalking: As a faculty member, I am designated as a "Responsible University Employee," and I must report all disclosures of sexual assault, sexual harassment, interpersonal violence, and stalking to UMD's Title IX Coordinator per University Policy on Sexual Harassment and Other Sexual Misconduct.

If you wish to speak with someone confidentially, please contact one of UMD's confidential resources, such as CARE to Stop Violence (located on the Ground Floor of the Health Center) at 301-741-3442 or the Counseling Center (located at the Shoemaker Building) at 301-314-7651.

You may also seek assistance or supportive measures from UMD's Title IX Coordinator, Angela Nastase, by calling 301-405-1142, or emailing titleIXcoordinator@umd.edu.
To view further information on the above, please visit the Office of Civil Rights and Sexual Misconduct's website at ocrsm.umd.edu.

Basic Needs Security

If you have difficulty affording groceries or accessing sufficient food to eat every day, or lack a safe and stable place to live, please visit UMD's Division of Student Affairs website for information about resources the campus offers you and let me know if I can help in any way.

Veteran Resources

UMD provides some additional supports to our student veterans. You can access those resources at the office of Veteran Student life and the Counseling Center. Veterans and active duty military personnel with special circumstances (e.g., upcoming deployments, drill requirements, disabilities) are welcome and encouraged to communicate these, in advance if possible, to the instructor.