# IOE 511/MATH 562: CONTINUOUS OPTIMIZATION METHODS, WINTER 2019

## General Information

**Lectures:** Tuesdays and Thursdays, 9:00-10:30 AM, 1014 Dow

**Instructor:** Prof. Marina A. Epelman (me/I in this document), mepelman@umich.edu, 2845 IOE
Office hours: see course site on Canvas

**GSI:** Geunyeong Byeon, gbyeon@umich.edu
Office hours: see course site on Canvas

**Web page:** Maintained on Canvas (pages with course schedule and computer resources are on the open web). Make sure you have access to this course's site as well as the Piazza and Gradescope pages for this course.

## Honor Code and Policies

All students in the class are presumed to be decent and honorable, and all students in the class are bound by the College of Engineering Honor Code (`http://ossa.engin.umich.edu/honor-council/`). You may not seek to gain an unfair advantage over your fellow students; you may not consult, look at, or possess the unpublished work of another without their permission; and you must appropriately acknowledge your use of another's work. Any violation of the honor policies appropriate to each piece of course work will be reported to the Honor Council, and if guilt is established penalties may be imposed by the Honor Council and Faculty Committee on Discipline. Such penalties can include, but are not limited to, letter grade deductions or expulsion from the University. **If you have any questions about these course polies, please consult the course instructor.**

**Homeworks:** You are allowed (indeed, encouraged) to consult with other students *currently enrolled in this class* during the conceptualization of a problem, and may receive help **only** from me and the GSI. All submitted homeworks (including writing, calculations, and computer codes and outputs, if relevant) should represent your own efforts and cannot, in any part, be copied from or inspired/motivated by material written by anybody else.

In particular, you are not allowed to look at, use, or in any way attempt to derive advantage from the existence of solutions to problems in the homeworks, or similar to them, whether these solutions were produced by former students or had been made available by previous or current instructors of this or other courses, or textbook publishers.

You are only allowed to use as reference and information source any and all material presented in lectures and made available on the Canvas site for this course this semester prior to the due date of the assignment, as well as general mathematical knowledge, which, if needed, you can look up in calculus, analysis, or linear algebra textbooks, or online reference guides to these subjects (Wikipedia and similar).

The spirit of the above restrictions is that you shouldn't look to other textbooks, or journals, or code, or the web, or solution manuals, or other people's knowledge, for solutions or hints for approaches to homework problems.

You are also not allowed to share with any individuals or websites or other repositories any documents you obtain on Canvas, Piazza, or Gradescope sites for this class, including but not limited to lecture notes, textbooks, homeworks and exams, or their solutions. Posting any such document or file is a violation of copyright laws. Moreover, although sharing your own answers to homework or exam questions is not a copyright violation, it is a violation of academic integrity, since it encourages other students to cheat.

**Exams:** Each student must complete the exam solely by her or his own efforts. I or the GSI proctoring the exam will answer clarification questions about statements of exam problems, but not give hints. The exam must be completed within the specified time.

Both exams will be closed-book, in-class exams. You will be allowed to use 1 page of your own notes for the midterm (2 pages for the final) and a simple calculator for arithmetic operations only.

## What this Course is About

**Catalog description:** Survey of continuous optimization problems. Unconstrained optimization problems: unidirectional search techniques; gradient, conjugate direction, quasi-Newton methods. Introduction to constrained optimization using techniques of unconstrained optimization through penalty transformations, augmented Lagrangians, and others. Discussion of computer programs for various algorithms.

**Informal description:** The goal of the course is to introduce the students to concepts and methods of constrained and unconstrained continuous nonlinear optimization. The course revolves around three issues in optimization: building optimization models of problems, characterization of their solutions, and algorithms for finding these solutions. As the semester progresses, I will compile a list of topics of all lectures on the web site. The outline of the topics we will cover is as follows:

- Introduction to optimization
- Optimality conditions for unconstrained problems
- Algorithms for unconstrained problems (steepest descent, Newton's, etc.) and analysis of their convergence
- Optimality conditions and constraint qualifications for constrained problems
- Convexity and its role in optimization
- Algorithms for constrained problems (SQP, barrier and penalty methods, etc.)
- If time remains: large-scale optimization problems; conic optimization problems, their applications, and methods for their solution.

Each topic, of course, includes mathematical "detours" introducing the appropriate background, specialized problem classes and their applications, etc.

**Required background:** The formal prerequisites are Math 217, Math 417 or Math 419. In practical terms, I assume a working knowledge of linear algebra as well as calculus/analysis, and a level of mathematical maturity (e.g., ability to follow and recreate the logic of mathematical proofs). Exposure to numerical computing, optimization, and its application fields is helpful but not required at the beginning of the course.

You also need to be able to program the algorithms we will be learning in the class in the Matlab programming language. You have likely used Matlab already, but if you haven't, it is exceptionally easy to use and to learn, and is particularly suitable for the kind of computational work we will be doing. A link to resources to get you started is provided on the web site. (The first programming assignment is likely to be in homework 3, so don't delay reviewing these resources.)

**Readings:** The required text is *Fundamentals of Nonlinear Optimization: a Constructive Approach* by Robert M. Freund and Jorge R. Vera. It is not yet published, and the authors very generously made a working copy available for students enrolled in the course via Canvas. You are welcome to download it for your reference, but do not distribute it or post it online in any fashion — doing so would be a severe copyright violation. Additional course materials will be distributed through the course web site.

Additional reference books, many available electronically through UofM library:

- "Nonlinear Programming; Theory and Algorithms" by M.S. Bazaraa, H.D. Sherali and C.M. Shetty
- "Lectures on Modern Convex Optimization" by A. Ben-Tal and A. Nemirovski
- "Introduction to Linear Programming" by D. Bertsimas and J. Tsitsiklis
- "Nonlinear Programming" by D. Bertsekas
- "Convex Analysis and Optimization" by D. Bertsekas
- "Convex Optimization" by S. Boyd and L. Vandenberghe
- "Linear and Nonlinear Programming" by A. Nash and A. Sofer
- "Numerical Optimization" by J. Nocedal and S. Wright
- "Nonlinear Optimization" by A. Ruszczynski
- K. Sigman, *Matlab Primer* (several editions, earlier ones available on-line).
- ...and many others.

## Course Format

**Lectures:** Attendance and attention in lectures is expected. Courtesy to the instructor and other students is expected: please arrive on time and silence audible notifications on all your electronic devices before the start of each lecture; if using laptops or tablets to take notes, make sure that your device is not a distraction to your neighbors, etc.

Lectures will use some slides, but will include a significant amount of writing on the board, and each students will be expected to take notes. A student is expected to independently discover and recover the material covered in any lecture skipped. Lectures will be recorded, but keep in mind that the lecture capture system does not do a good job with video of the whiteboards, so recordings are not a reliable substitute for note-taking.

**Grading:** Midterm (30%), final exam (40%), and weekly (or so) homework (30%). Midterm will take place during class time on 2/28/19 (Thursday before Winter break). The final exam will take place according to the University schedule, on Thursday, 4/25/19 at 1:30 pm–3:30 pm.

**Homeworks** Weekly homework assignments will be posted a week before the due date, and will be due at 11:55 PM on the due date; your submissions will need to be uploaded electronically. There will be a 12-hour grace period after each due date, during which you can double-check that the right version of the submission file(s) appears on the website and upload a new version if necessary. Solutions to each of the homework assignments will be posted after the grace period ends. There will be no grade penalty for submitting the assignment during the grace period, but after the due date the homework is considered "closed" — questions about this homework will not be answered by the instructors or on Piazza during the grace period. All submissions must be turned in by uploading them through the web (not by emailing them to the instructors), and *no submissions will be accepted after the end of the grace period.* Your lowest homework score will be dropped from the total in calculations of the course grade.

Your written answers must be submitted in PDF format through Gradescope. You must name your submitted file `uniquename_hwk#.pdf` — for example, if I were submitting my solutions to the first homework, I would name my file `mepelman_hwk1.pdf`. Some assignments will also require you to submit one or several Matlab files — those will have to be submitted via Canvas, in addition to the Gradescope submission of the written part of the assignment.[1]

[1]It is unfortunate that we need to use two different websites for submission of written and code-based portions of

**You are required to type** rather than hand-write your submissions. In addition to making the grading process faster and easier for the instructors, typing your answers is beneficial to you. Indeed, to receive full credit, your answers — their steps, and the logic connecting them — must be clearly presented, well-written, and easy to understand. When you prepare your solutions electronically, you can review and edit your writing, notation, explanations, and derivations for correctness, clarity, and ease of understanding, which often leads to higher scores.

Not surprisingly, homework solutions in this course will contain a lot of mathematical expressions. Although many word processing applications include some capability for typesetting math, I highly recommend that you learn the basics of LaTeX typesetting system, if you haven't already, and use it to prepare your solutions. A sample file using a lot of mathematical notation frequently used in this course will be made available with the first homework as a reference/starting point. Website `www.overleaf.com` also contains many samples, instructions, and FAQs for getting started with LaTeX.

**Exams** No makeup or late exams will be given, unless necessitated by serious emergencies or medical issues. If you are unable to take the scheduled exam due to such circumstances, notify me immediately; I may ask for explanatory documentation from appropriate sources.

**Disability Statement**

The University of Michigan is committed to providing equal opportunity for participation in all programs, services and activities. Request for accommodations by persons with disabilities may be made by contacting the Services for Students with Disabilities (SSD) Office located at G664 Haven Hall. The SSD phone number is 734-763-3000. Once your eligibility for an accommodation has been determined you will be issued a verified individual services accommodation (VISA) form. Please present this form to me at the beginning of the term, or at least two weeks prior to the need for the accommodation (e.g., midterm exam).

the homeworks. Gradescope does not allow for submission and testing of Matlab code, but for written assignments, it allows us to provide detailed, better organized feedback. We think the benefits of using Gradescope for written assignments make it worth the extra step required to submit your Matlab code separately.