

# Introduction

Chul Min Yeum

Assistant Professor

Civil and Environmental Engineering  
University of Waterloo, Canada

**AE121: Computational Method**

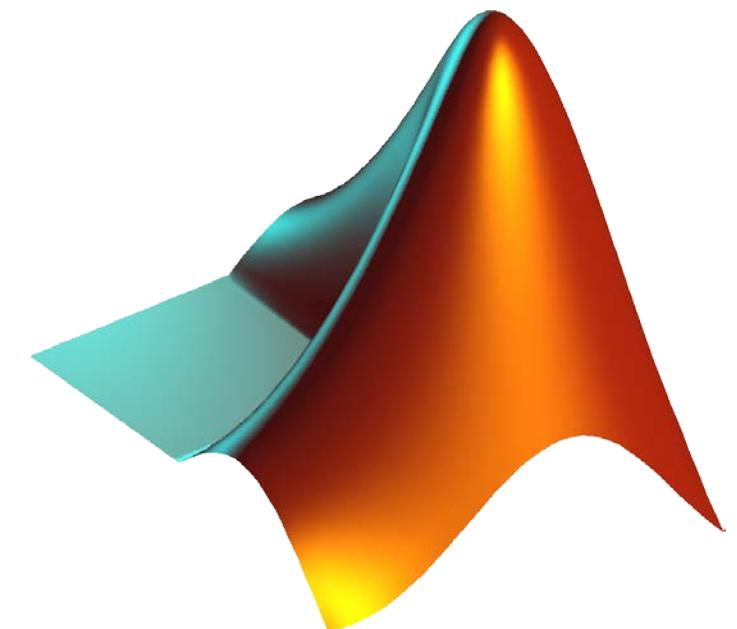


**UNIVERSITY OF WATERLOO**  
**FACULTY OF ENGINEERING**

Last updated: 2019-05-02

# What is MATLAB?

MATLAB is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numeric computation.



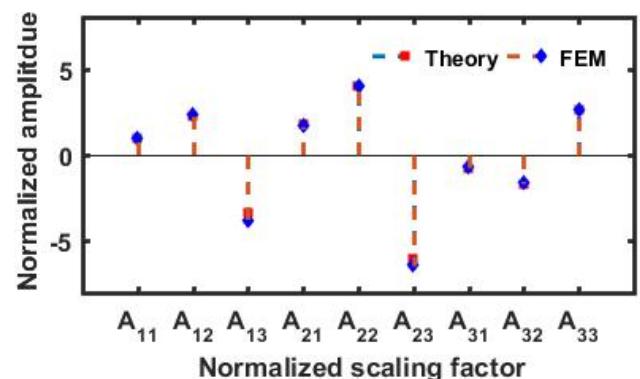
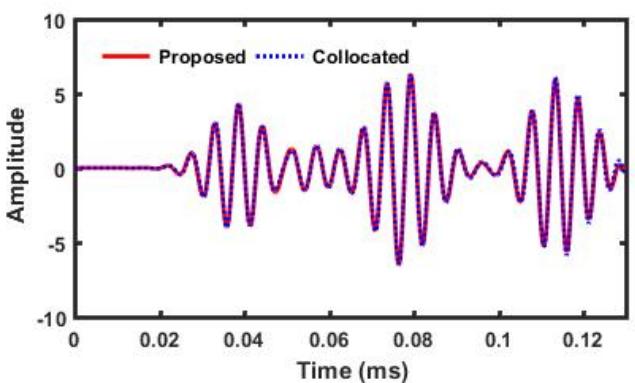
# Why Do We Learn MATLAB or Computer Programming?

- Why do we use a calculator?
- Why do we use MS Excel?
- Why do we use Maple or Wolfram|Alpha?
- Why do we use Origin?
- Why do we use MS PowerPoint?

# Introduction to MATLAB

- Very powerful software package
- Many mathematical and graphical applications
- Has programming constructs
- Also has many **built-in functions**
- Can use interactively in the Command Window, or write your own programs
- **Easy to debug your program**
- In the Command Window the >> is the prompt
  - At the prompt, enter a command or expression
  - MATLAB will respond with a result

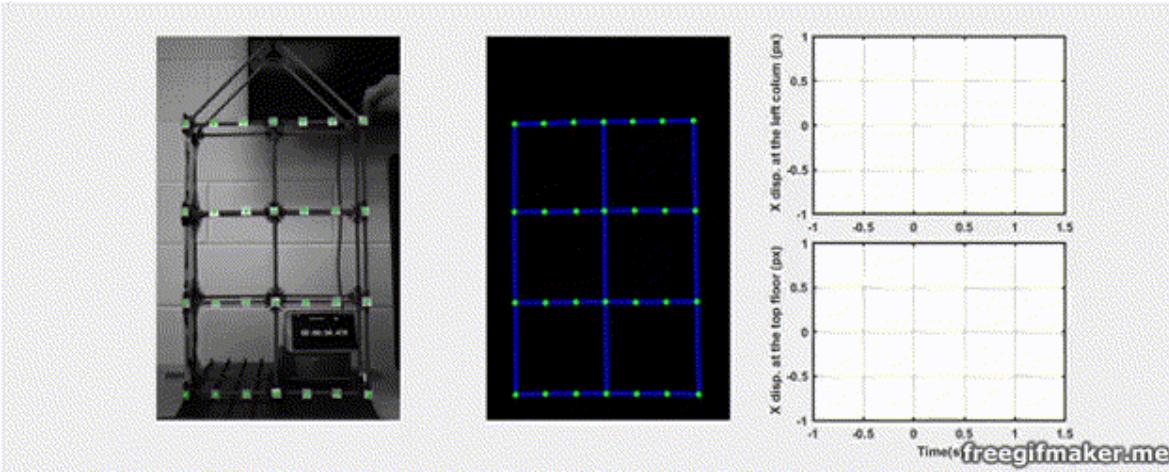
# How Have I Used MATLAB?



[Draw graphs](#)

[Graphic user interface](#)

# How Have I Used MATLAB? (Continue)



Making videos



Haiti earthquake  
in 2010 (3,439 images)

L'Aquila (Italy) earthquake  
in 2009 (414 images)

Florida hurricanes in 2004  
(1,178 images)

Data management

A screenshot of a GitHub repository page for 'chulminy / Vision\_Based\_Automated\_Crack'. The repository has 1 issue, 0 pull requests, 0 projects, and 0 wiki pages. It was updated by chulminy on Jan 10, 2017. The code directory contains files like CompFeatChnPatch.m, CompHaarPatch.m, ComplntChnImg.m, CrackDetection.m, Parameters.m, RunLabeling.m, RunOutput.m, RunTest.m, and RunTrainClassifier.m, all updated or published within the last 2 years.

Program for research

# Course Website

The screenshot shows a GitHub repository page for 'chulminy/AE121'. The repository has 30 commits, 1 branch, 0 releases, and 2 contributors. The README.md file contains information about the course, including the instructor's name, email, and office hours. The page also features a 'Join GitHub today' modal.

Join GitHub today  
GitHub is home to over 36 million developers working together to host and review code, manage projects, and build software together.  
[Sign up](#)

Dismiss

AE121: Computational Method (Spring, 2019) at University of Waterloo

30 commits 1 branch 0 releases 2 contributors

Branch: master New pull request Find File Clone or download

File	Commit Message	Time Ago
chulminy Update README.md	Latest commit dc8e7d6 2 days ago	
lab	update	3 days ago
lecture	update	2 days ago
.gitignore	update	3 days ago
README.md	Update README.md	2 days ago
civil_engineer.png	update	8 days ago

README.md

## AE121: Computational Method (Spring, 2019)



Instructor: Chul Min Yeum ([cmyeum@uwaterloo.ca](mailto:cmyeum@uwaterloo.ca))  
Class: Monday and Tuesday, 2:30 to 4:00 PM at CPH 1319  
Programming Lab: Thursday, 9:30 to 11:30 AM at CPH 1346  
TA: Jason Connelly ([jpconnelly@edu.uwaterloo.ca](mailto:jpconnelly@edu.uwaterloo.ca)) and Juan Park  
Office Hours (Instructor): Monday and Tuesday, 4:00 to 5:00 PM at E2-2313  
Office Hours (TA): Monday to Friday 2:00 to 5:00 PM at E2-2313

Last updated: 2019-04-29

### Announcement

2019-04-25: The course website is published.  
2019-04-29: The course materials are updated [[Link](#)].

<https://github.com/chulminy/AE121>

## Course Description

This course offers a practical introduction to computer programming for engineering students using MATLAB. MATLAB is an easy and readable programming language and is an excellent choice for those learning programming for the first time. This course will cover various topics including programming fundamentals, matrix operations, file I/O, numerical methods, and data visualization. Students who successfully complete this course will gain a fundamental understanding of computer programming techniques and be able to operate, analyze, and visualize all types of data.

## Course Description (Continue)

This course incorporates two new learning tools in MATLAB to enhance programming-learning environment: [\*\*Live Editor\*\*](#) and [\*\*MATLAB Grader\*\*](#). The *Live Editor* is a scripting platform to create, edit and run MATLAB scripts as well as add formatted text, images, and equations so that students can learn programming in more informative and interactive ways. A tutorial for each topic is provided in the Live Editor format. The programming lab and homework assignments will be designed using the [\*\*MATLAB Grader\*\*](#). This tool allows students instant feedback on the correctness of their solutions through the pre-designed testers. Thus, students can be highly motivated and inspired to persevere with completing the assignments. You are expected to spend extra 6 hours ([and more...](#)) a week studying class materials as well as working on the programming lab and homework assignments.

## Course Description (Continue)

- This course includes lectures, tutorials, programming labs, and homework assignments:
- **Lectures** will cover the major topics, emphasizing and discussing the important points, and introduce programming examples in tutorials.
- **Tutorials** are intended to review the lectures and prepare for programming lab and homework assignments.
- **Programming-labs** provide an opportunity to practice writing and debugging programs with TA's and instructor's assistance.
- **Homework assignments** are to provide additional practice for programming and evaluate your knowledge and programming skills

## Course Objectives

- Define your problems with programmable logic in MATLAB
- Compose working, efficient, and readable code in various ways
- Debug your code to identify and fix programming errors
- Identify suitable keywords to search for code snippets on the web
- Apply MATLAB to solve mathematical and engineering problems

## Programmer Job Interview

Do you know how to Google?



um, yes!

Welcome aboard

doc

- Defining your problem
- Identifying keywords
- Understanding the answer
- Implementing them in your code

# How to Improve Your Programming Skills

- Step 1: Ensure your program runs without errors or warnings
- Step 2: Understand a model solution provided (e.g., program logic, useful built-in functions, programming styles)
- Step 3: Consider how to modify/improve your own programs
- Step 4: Re-write your program from scratch without copying solutions

Extremely important!!

## How to Complete this Course with Over 90

- Review lecture slides and tutorials before attending programming labs
- Submit all lab and homework assignments
- Do not skim the code in tutorials and assignment solutions. **Type and run all scripts in your editor to familiarize yourself with the syntax**
- Post **at least one question** on a course website every week and help/teach other students
- Replicate built-in functions with your own code
- Employ a MATLAB editor, workspace browser, and debugging tool actively
- **Vectorize your code without loop and selection statements**
- **Use MATLAB instead of MS Excel, Wolfram | Alpha, or a calculator to complete your tasks (for long-term)**

"**Linear algebra**" (AE 115) and "Calculus 1 for Engineering" (MATH 116) are prerequisites for this course, which you learned in the 1A term. If you are not familiar with these topics, please review corresponding course notes or tutorials.

MATLAB (**matrix** laboratory)

This course is not based on any particular textbook. However, lecture slides have been prepared using the following references:

- Attaway, Stormy. "*MATLAB-A Practical Introduction to Programming and Problem Solving,*" Elsevier, 2018.
- Moore, Holly. "*MATLAB for Engineers,*" Pearson, 2017.

[Purchase a MATLAB license...](#)

# Course Outline

Class	Topics		
Class 01	Introduction (MATLAB and Course)	Class 13	Text Manipulation
Class 02	Basic MATLAB Programming	Class 14	Plotting
Class 03	Vector & Matrices I	Class 15	Data Structure
Class 04	Vector & Matrices II	Class 16	File I/O
Class 05	Built-in Functions I	Class 17	Symbolic Function
Class 06	Built-in Functions II	Class 18	Matrix Algebra
Class 07	Review Class	Class 19	Review Class
Class 08	Selection Statement	Class 20	Numerical Technique I
Class 09	Loop Statement	Class 21	Numerical Technique II
Class 10	Pseudocode	Class 22	Advanced Topic I: Advanced Function
Class 11	Function	Class 23	Advanced Topic II: Graphics
Class 12	Review Class	Class 24	Review Class

## Homework Assignments

Students should be encouraged to work in groups through collaborative learning, but to submit their assignments individually. Students **must not share their solutions with other students**. The TA will run software to check your scripts to detect plagiarism. Please do not think that you may be able to deceive graders. Although you pass all the problems in MATLAB Grader, you will get a 0 grade if your solution is considered plagiarism and may be reported to Engineering Undergraduate Associate Dean as [an academic offense](#). [Each homework will be uploaded on Saturday night and must be submitted before the Monday class on the following week \(ten days\)](#). After the due date, homework assignment online will be closed, and you are not able to submit your works. Again, **no late submission is accepted**, and any missed homework will be given a grade of 0. Thus, students who are not able to submit their assignment must inform the TA in advance with an email or at the office hour if you have a valid excuse.

All communication will be made through [\*\*this course website\*\*](#), especially for this web page. The instructor will frequently make a note in the "[\*\*Announcement\*\*](#)" section after updating the web page. Students can configure email notification for by [\*\*"watching"\*\*](#) this course website or use a version control system for tracking its changes (The link will be provided on every announcement to track the changes). Thus, **students are responsible for checking the website regularly** for any relevant course information or announcements. We will post an important announcement through LEARN, but LEARN will be only used for such announcement notification. Lecture slides, homework assignments, tutorials, and solutions are only accessible through this course website.

## Communication (Continue)

In this course, the instructor, TAs, and the students are encouraged to engage in online discussions to create and facilitate a **collaborative learning experience**. Students are invited to ask questions and answer them and share their knowledge and resources. **Please direct your communications to the Issues board (tab) on this website.** However, if there is a good reason not to use the discussion forum (e.g., personal matters, a question that might reveal your solution of your report, etc.), please directly contact the TA via email ([jpconnelly@edu.uwaterloo.ca](mailto:jpconnelly@edu.uwaterloo.ca)) or use the office hours.

## Communication (Continue)

- To post questions or comments, students should make their own Github account. Students do not need to make their accounts using school emails if they already have accounts or they would have anonymity.
- Please be respectful of your peers, instructor, and others in your posts and comments.
- Asking good questions is another way of learning by summarizing and explaining what you know and do not know. Thoughtful questions and answers will help all students in this class. Before posting a question, please ask yourself whether you are truly stuck (meaning that the answers cannot be readily found on the web) and your question are complete and legible.
- In addition, please check if the same question was asked before.
- Inappropriate posts and comments will be immediately deleted and closed without notice.

The final grade will be based on the total marks earned from two exams (midterm and final), nine programming lab assignments, and ten homework assignments. Percentages of the grade components are provided as:

- **Programming lab assignments (20%)**
- **Homework assignments (20%)**
- **Midterm exam (25%)**
- **Final exam (35%)**

You can get all marks from  
programming lab and  
homework assignments

Note that assignments and exams are cumulative.

# Weekly Workflow

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
<b>Lecture</b>	14:30 – 16:00	14:30 – 16:00					
<b>Programming lab</b>				09:30 – 11:30			
<b>Assignment open</b>						00:00	
<b>Lab assignment due</b>				11:30			
<b>Homework assignment due</b>	14:30						
<b>Instructor's office hours</b>	16:00 – 17:00	16:00 – 17:00					
<b>TA's office hours</b>			11:30 – 12:30	11:30 – 13:00	12:30 – 13:30		
<b>Best time for tutorial study</b>							
<b>Best time for starting homework</b>							
<b>Best time for posting questions</b>							