

# Syllabus

<b>COURSE INFORMATION</b>	<b>ME 261 Mechanical Engineering Problem Solving</b> <b>3 credits</b> <b>Required</b>	<b>Fall 2025</b>
<b>INSTRUCTOR:</b>	Dr. Yanxing Wang <a href="mailto:yxwang@nmsu.edu">yxwang@nmsu.edu</a>	Office: JH134    Phone: 646-4369    email:
<b>TEACHING ASSISTANT</b>	David Vazquez	email: <a href="mailto:a305030@nmsu.edu">a305030@nmsu.edu</a>
<b>CATALOG DESCRIPTION:</b>	Introduction to programming in MATLAB. Numerical techniques for root finding, solution of linear and nonlinear systems of equations, integration, differentiation, and solution of ordinary differential equations.	
<b>PREREQUISITES:</b>	Math 192: CALCULUS II	
<b>TEXT BOOK:</b>	None	
<b>BOOK RECOMMENDED:</b>	<b>Applied Numerical Methods with MATLAB for Engineers and Scientists, by Steven C. Chapra</b>	
<b>CLASS FORM</b>	<b>In-person in classroom</b>	
<b>EXAM FORM</b>	<b>In-person in computer labs</b>	
<b>SOFTWARE REQUIRED</b>	MATLAB, VPN: <a href="http://software.nmsu.edu">http://software.nmsu.edu</a> <ul style="list-style-type: none"><li>• You need to use VPN to put your computer on the network of NMSU when using MATLAB</li></ul>	

# Syllabus

<b>CLASS SCHEDULE:</b>	Lectures:    Tue 2:30pm - 3:20pm    JH 134 Fri 2:00pm – 2:50pm    JH 134  •    Lecture notes will be uploaded to Canvas before classes  Lab:    Tue 3:30pm - 6:00pm JH 245
<b>OFFICE HOURS:</b>	9:30 a.m.-12:00 p.m. Wed or by email  <b>JH 134</b>

# Syllabus

<b>COURSE OBJECTIVES</b>	<ul style="list-style-type: none"><li>• Students will learn a variety of numerical methods that are useful in both basic and advanced engineering calculations.</li><li>• Students will learn how to formulate algorithms and write programs to solve engineering problems.</li><li>• Students will develop an appreciation for the hazards and limitations of numerical solutions, including accuracy, stability, and computer limitations of memory and speed.</li></ul>
<b>TOPICS COVERED</b>	<ul style="list-style-type: none"><li>• MATLAB Program Environment</li><li>• MATLAB Functions</li><li>• Roots of Equations</li><li>• Linear systems of equations</li><li>• Nonlinear systems of equations</li><li>• Interpolation and Curve fitting</li><li>• Numerical differentiation and integration</li><li>• Solution of ordinary differential equations</li></ul>

# Syllabus

<b>GRADES:</b>	<p>Attendance: 10%</p> <p>Homework: 10%</p> <p>Exam1: 25%</p> <p>Exam2: 25%</p> <p>Exam3: 30%</p> <p>A score between: 85 – 100 A (including A- and A+) 75 – 84 B (including B- and B+) 65 – 74 C (including C- and C+) 60 – 64 D 0 – 59 F</p> <ul style="list-style-type: none"><li>• In principle, no makeup exams will be provided. Exceptions are given only when you have a good excuse and discuss at least two work days before the exams.</li></ul>

# Syllabus

Assignment Policies:	
	<ul style="list-style-type: none"><li>• All computer programs must be written in MATLAB as instructed and well commented.</li><li>• Homework will be assigned on Canvas. Due date will be clearly specified.</li><li>• All Homework and Labs must be submitted through CANVAS by the assignment due date and time.</li><li>• Late submission will lead to <b>half grades</b>.</li><li>• Collaboration in the form of discussion of formulation of solutions or results is encouraged; however, each individual must work independently to create the solution and computer programs.</li><li>• Homework solutions will be given after graded.</li></ul>

# Syllabus

<b>Student Accessibility Services:</b>	Section 504 of the Rehabilitation Act of 1973 and the Americans with Disabilities Act (ADA) covers issues relating to disability and accommodations. If a student has questions or needs an accommodation in the classroom (all medical information is treated confidentially), contact:  Trudy Luken Student Accessibility Services (SAS) - Corbett Center, Rm. 244 Phone: 646.6840 E-mail: <a href="mailto:sas@nmsu.edu">sas@nmsu.edu</a> Website: <a href="http://www.nmsu.edu/~ssd/">www.nmsu.edu/~ssd/</a>
<b>Institutional Equity:</b>	NMSU policy prohibits discrimination on the basis of age, ancestry, color, disability, gender identity, genetic information, national origin, race, religion, retaliation, serious medical condition, sex, sexual orientation, spousal affiliation and protected veterans status. Furthermore, Title IX prohibits sex discrimination to include sexual misconduct, sexual violence, sexual harassment and retaliation. For more information on discrimination issues, Title IX or NMSU's complaint process contact:  Gerard Nevarez or Agustin Diaz Office of Institutional Equity (OIE) - O'Loughlin House Phone: 646.3635 E-mail: <a href="mailto:equity@nmsu.edu">equity@nmsu.edu</a> Website: <a href="http://www.nmsu.edu/~eeo/">www.nmsu.edu/~eeo/</a>
<b>Policies:</b>	Plagiarism is using another person's work without acknowledgment, making it appear to be one's own. Intentional and unintentional instances of plagiarism are considered instances of academic misconduct and are subject to disciplinary action such as failure on the assignment, failure of the course or dismissal from the university. The NMSU Library has more information and help on how to avoid plagiarism at <a href="http://lib.nmsu.edu/plagiarism/">http://lib.nmsu.edu/plagiarism/</a>
<b>Author/Date</b>	Yanxing Wang August 2021

# ME 261: Lecture 1

# Lecture 1: Introduction to MATLAB

## ➤ MATLAB

- A powerful mathematical and graphical software package.  
It has the numerical, graphical and programming capabilities

## ➤ MATLAB Environment

- Command window: used to enter commands and expressions
- Current directory window: shows where files are saved. By default, it is the work director
- Workspace window: shows all variables that have been created.
- Figure window: show figures

## ➤ Ways to Use MATLAB

- Command window: use MATLAB interactively
- Script files or M-files: write program

# Lecture 1: Introduction to MATLAB

## ➤ Variables and Assignment Statement

- Format of an assignment statement:

*variablename = expression* (variable name is always on the left)

The equal sign is the assignment operator, and does not mean equality

- Examples:

```
>> mynum = 6
```

```
mynum =
```

```
6
```

- Just typing the name of a variable will display its value

```
>> mynum
```

```
mynum =
```

```
6
```

- Putting a semicolon at the end of a statement suppresses the output.

```
>> res = 9 - 2;
```

```
>>
```

## ➤ Default Variable Name: *ans*

```
>> mynum + 3
```

```
ans =
```

```
9
```

# Lecture 1: Introduction to MATLAB

## ➤ Rules for Variable Names

- Must begin with a letter. (After that, the names can contain letter, digits, and underscore characters, but it cannot have a space)
- Be case-sensitive
- Can not use ‘reserved words’, such as ‘for’ ‘if’ ...
- Should not use the names of built-in functions
- Be mnemonic

## ➤ Commands Related to Variables

- *who*: shows variables that have been defined in this Command Window (just shows the name of the variables)
- *whos*: show variables that have been defined in this Command Window (this shows more information on the variables)
- *clear*: clears out all variables so they no longer exist
- *clear variablename*: clears out a particular variable

# Lecture 1: Introduction to MATLAB

## ➤ Expressions

- Created using values, variables, operators, built-in functions and parenthesis. Variables must have been already created.

```
>> 2 * sin(1.4)
```

```
ans =
```

```
1.9709
```

## ➤ Operators

- Two kinds of operators: unary operators and binary operators
  - + addition
  - negation, subtraction
  - \*
  - / division (divided by)
  - \ division (divided into)
  - ^ exponentiation

## ➤ Operator Precedence Rule

```
>> 4 + 5 * 3
```

```
ans =
```

```
19
```

# Lecture 1: Introduction to MATLAB

## ➤ Built-in Functions and Help

- *help topicname*: show a list of functions within a particular topic

Example: *>> help elfun*

- *help functionname*: show what a particular function does and how to call it.

Example: *>> help sin*

## ➤ Call a Function

- Format: *functionname(arguments)*

Example: *>> abs(-4)*

# Lecture 1: Introduction to MATLAB

## ➤ Predefined Special Values

Function	Purpose
<b>pi</b>	Contains $\pi$ to 15 significant digits
<b>i, j</b>	Contain the value $i(\sqrt{-1})$ .
<b>Inf</b>	This symbol represents machine infinity. It is usually generated as a result of a division by 0.
<b>NaN</b>	The symbol stands for Not-a-Number. It is the result of an undefined mathematical operation, such as the division of zero by zero.
<b>clock</b>	This special variable contains the current date and time in the form of a 6-element row vector containing the year, month, day, hour, minute, and second.
<b>date</b>	Contains the current data in a character strings format, such as 24-Nov-1998.
<b>eps</b>	This variable name is short for “epsilon”. It is the smallest difference between two numbers that can be represented on the computer.
<b>ans</b>	A special variable used to store the result of an expression if that result is not explicitly assigned to some other variable.

# Lecture 1: Introduction to MATLAB

- **Types** – every expression or variable, has a type associated with it.
  - Float numbers: *single* and *double*
  - Integers: *int8*, *int16*, *int32*, *int64*

The number represents the number of bits used to store the integer and its sign
  - Characters or strings: *char*

Both characters and strings are enclosed by single quotes

```
>> v = 'a'  
v =  
    'a'
```
  - True/false values: *logical*
- In the workspace window, for every variable, the variable name, value and types can be seen.
- By default, numbers are stored as **type double** in MATLAB

# Lecture 1: Introduction to MATLAB

## ➤ Type Casting

- Functions that convert values from one type to another. The name of these functions are the same as the names of the types

Example:    *>> val = 6+3*

*val =*

*9*

*>> valnew = int32(val)*

*>> whos*

# Lecture 1: Introduction to MATLAB

## ➤ Characters and Encoding (1/2)

- A character is represented using single quotes (e.g., 'a', '3')
- Characters are put in an order using what is called a **character encoding**
- All characters are placed in a sequence and given equivalent integer values
- The numeric functions can be used to convert a character to its equivalent numeric values (double, int32)

Example:      *>> numequiv = double('a')*  
                 *numquiv =*  
                 *97*

- The function **char** does the reverse; it converts from any number type to the type char

Example:      *>> char(numequiv)*  
                 *ans=*  
                 *'a'*

# Lecture 1: Introduction to MATLAB

## ➤ Characters and Encoding (2/2)

- Math can be done on characters

Example:      *>> 'a' +2*

*ans=*

*99*

**What's the type of ans?**

- MATLAB can also handle strings, which are sequences of characters

Examples:      *>> double('abcd')*

*ans=*

*97 98 99 100*

*>> char('abcd'+1)*

*ans=*

*bcde*

# Lecture 1: Introduction to MATLAB

## ➤ Random numbers

- The function *rand* can be used to generate random real numbers in the range from 0 to 1. There is no argument passed to the *rand* function.

```
>> rand
```

```
ans =
```

```
0.9501
```

- Multiplying the result by an integer N would return a random real number in the range from 0 to N.

```
>> rand * 10
```

- *rand\*(high-low)+low* Would generate a random real number in the range from low to high.

# Lecture 1: Introduction to MATLAB

## ➤ Summary

- MATLAB environment and how to use MATLAB
- MATLAB variables and assignment statement
- Built-in functions and help
- Variable and expression types
- Characters and encoding

## ➤ Homework #1 (on Canvas)

- Due next Friday