Lecture 01
Bits, Bytes, Codes, Variables

Erdal Yılmaz

Cornell University

June 24, 2012
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

Instructor
Erdal Yilmaz
4106 Upson Hall
ey45@cornell.edu

Teaching Assistant
Jyoti Pandey
jp833@cornell.edu
Syllabus

- Staff
- Course Information
  - Introduction to programming
  - Learn a high-level programming language
  - Programming concepts
  - Problem solving
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

MATLAB from MathWorks
Syllabus

- Staff
- Course Information
- Software
- Course Websites
  - Lectures and Labs
  - Office Hours
  - Textbook
  - Homeworks
  - Quizes
  - Final Exam
  - Grading
  - Academic Integrity
  - Student with Disabilities

Course Websites

web http://www.cs.cornell.edu/courses/cs1109/2012su
shortcut http://www.cs1109.info

for homeworks http://cms.csuglab.cornell.edu
shortcut http://hw.cs1109.info
Session: 001
Time: 10:00 AM - 11:05 AM
Lecture: 211 Upson Hall on MW
Lab: Upson B7 on TR

Session: 002
Time: 11:30 AM - 12:35 PM
Lecture: 215 Upson Hall on MW
Lab: Upson B7 on TR
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

Where

- Upson B7
  - Erdal
    - Sundays 1pm-2pm
    - Fridays 10am-noon
  - Jyoti
    - Wednesdays 2pm-3pm
    - Sundays 2pm-4pm
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

Title: Insight Through Computing
A MATLAB Introduction to Computational Science and Engineering

Authors: Charles F. Van Loan, K.-Y. Daisy Fan
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- **Homeworks**
  - Assigned on Wednesdays at 1pm
  - Due on next Monday by 10am
  - One or two multi-part questions
  - Upload to CMS website
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities
In-class quizzes:
- Short answers
- Anytime, lecture or lab

Online quizzes:
- Easy
- Multiple choice
- Assigned on Mondays at 1pm
- Due on Wednesdays by 10am
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

either one prelim and a group project
or just final exam
vote on CMS survey today
check your email after class!
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

\[ \text{HW} = \text{Normalize homeworks to 100 ... and take average} \]
\[ \text{QZ} = \text{Normalize quizes to 100 ... and take average} \]
\[ \text{FL} = \text{Normalize prelim and project to 100 ... and take average} \]
\[ \text{or Normalize Final Exam result to 100} \]
\[ \text{Grade} = (\text{FL} \times 0.3) + (\text{HW} \times 0.5) + (\text{QZ} \times 0.2) \]
\[ S/U \quad \text{Grade} \geq 75 \]
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

Read Code of Academic Integrity
Submit your own work
Acknowledge any help received
Syllabus

- Staff
- Course Information
- Software
- Course Websites
- Lectures and Labs
- Office Hours
- Textbook
- Homeworks
- Quizes
- Final Exam
- Grading
- Academic Integrity
- Student with Disabilities

- Contact Student Disability Services: at 420 CCC, (607) 254 4545
- Send an email and talk to me
Bits, Bytes

- Bit (Binary Digit) ∈ \{0, 1\}
- Byte ≡ 8 bits (e.g. 01000001)
- KB = 2^{10}(\approx 10^3) bytes
- MB = 2^{20}(\approx 10^6) bytes
- GB = 2^{30}(\approx 10^9) bytes
- TB = 2^{40}(\approx 10^{12}) bytes
- ...
A character is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)
Representation of Text

- A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)
- A **string** is a sequence of characters (e.g. ‘CS 1109’)
Representation of Text

- A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)
- A **string** is a sequence of characters (e.g. ‘CS 1109’)
- \{0, 1\}
A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)

A **string** is a sequence of characters (e.g. ‘CS 1109’)

\[
\{0, 1\} \leftrightarrow \{A, B, \ldots, a, b, \ldots, 0, 1, 2, 3, \ldots\}
\]
Representation of Text

- A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)
- A **string** is a sequence of characters (e.g. ‘CS 1109’)
- \( \{0, 1\} \leftrightarrow \{A, B, \ldots, a, b, \ldots, 0, 1, 2, 3, \ldots\} \)
- Need a translation between two *alphabets*
- Standards: ASCII (1 byte), Unicode (2 bytes)
Representation of Text

- A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)
- A **string** is a sequence of characters (e.g. ‘CS 1109’)

\[
\{0, 1\} \leftrightarrow \{A, B, \ldots, a, b, \ldots, 0, 1, 2, 3, \ldots\}
\]

- Need a translation between two *alphabets*
- Standards: ASCII (1 byte), Unicode (2 bytes)
- ‘A’ ≡ 65 = (01000001)_2
A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)

A **string** is a sequence of characters (e.g. ‘CS 1109’)

{0, 1} ↔ {A, B, ..., a, b, ..., 0, 1, 2, 3, ...}

Need a translation between two *alphabets*

Standards: ASCII (1 byte), Unicode (2 bytes)

‘A’ ≡ 65 = (01000001)_2

‘B’ ≡ 66 = (01000010)_2
Representation of Text

- A **character** is a symbol of written language (e.g. ‘K’, ‘m’, ‘!’)
- A **string** is a sequence of characters (e.g. ‘CS 1109’)

- \( \{0, 1\} \leftrightarrow \{A, B, \ldots, a, b, \ldots, 0, 1, 2, 3, \ldots\} \)
- Need a translation between two *alphabets*
- Standards: ASCII (1 byte), Unicode (2 bytes)
  - ‘A’ \( \equiv 65 = (01000001)_2 \)
  - ‘B’ \( \equiv 66 = (01000010)_2 \)
  - ‘a’ \( \equiv 97 = (01100001)_2 \)
A variable is a labeled memory location which holds a value.
A variable is a labeled memory location which holds a value

An assignment is storing the result of an expression into a variable

variable = expression
A variable is a labeled memory location which holds a value

An assignment is storing the result of an expression into a variable

variable = expression

\[ a = 5; \]
\[ b = a + 1; \]
\[ c = 2 \times b; \]
A **variable** is a *labeled* memory location which holds a *value*

An **assignment** is storing the result of an expression into a variable

*variable = expression*

\[ a = 5; \]
\[ b = a + 1; \]
\[ c = 2 \times b; \]

*At the end: a stores 5, b stores 6, c stores 12.*
Simple Calculator

- (enter a number) 17
Simple Calculator

- (enter a number) 17
- (select an operation) +

What is missing?

Labels for stored values!

We implicitly refer to them: the first number and the second number. But there is no way we can reuse them!

Variables provide labels for stored values.
Simple Calculator

- (enter a number) 17
- (select an operation) +
- (enter another number) 29

What is missing?
Labels for stored values!
We implicitly refer to them:
the first number
and
the second number
But there is no way we can reuse them!
Variables provide labels for stored values.
Simple Calculator

- (enter a number) 17
- (select an operation) +
- (enter another number) 29
- (press execute)

What is missing?
Labels for stored values!
We implicitly refer to them:
the first number and the second number
But there is no way we can reuse them!
Variables provide labels for stored values.
Simple Calculator

- (enter a number) 17
- (select an operation) +
- (enter another number) 29
- (press execute)
- (result) 46

What is missing?
Labels for stored values!
We implicitly refer to them:
the first number
and
the second number
But there is no way we can reuse them!
Variables provide labels for stored values.
Simple Calculator

- (enter a number) 17
- (select an operation) +
- (enter another number) 29
- (press execute)
- (result) 46

What is missing?
Simple Calculator

- (enter a number) 17
- (select an operation) +
- (enter another number) 29
- (press execute)
- (result) 46

What is missing?
- Labels for stored values!
- We implicitly refer to them: the first number and the second number
- But there is no way we can reuse them!
- Variables provide labels for stored values.
MATLAB Demo

- Command Window
- Workspace
- Command History
- Current Directory
- Help