CVEN 302

Computer Applications in Engineering and Construction

Dr. Tony Cahill

Environmental and Water Resources Division
Instructors

- **Instructor:** Tony Cahill
  - **Office:** WERC 205J
  - **Office Hours:** T/R 3:00 – 4:00PM. Or by appointment via email
  - **Phone:** 979-862-3858

- **TA:** TBD
  - **Office:** Office Hours:
  - **Computer Lab:** Email:
  - **Phone:** 
Finding Your Ways Easier

Zachry Engineering Center, 127B

WERC 205J
Administration

- In-Class Exercise require Civil Student Computer Access
  - Request CIVIL STUDENT user ID and Password from Suite 609 CE/TTI
  - Available to ANY Civil Engineering student
- Regularly check your emails for course information
Course Objectives

- Objectives: Develop efficient, computer oriented solutions to engineering problems

- Learning Outcomes
  - Write simple program modules
  - Test program accuracy
  - Synthesize multiple program modules into larger software packages
  - Solve problems using numerical methods
  - Select appropriate numerical methods to solve engineering problems
  - Distill numerical results into readable format to answer analysis and design questions
Course Description

Textbooks

- *Computer Applications in Engineering and Construction* – Texas A&M University by Thomson Custom Publishing (custom MATLAB manual)

Pre-requisites: ENGR 112, MATH 308 or registration therein

Course website

- [https://ceprofs.civil.tamu.edu/cahill/](https://ceprofs.civil.tamu.edu/cahill/)
Course Organization

- **Homework**
  - Problem sets (10%)
  - One per week generally, hand calculations
  - Grade best 10 out of all (~12)
  - Solutions posted night before due date

- **Programming assignments**
  - 10 in total, programming exercises
  - Challenging problems may be assigned as extra credit
Grading

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem Sets, Class Participation, Quizzes</td>
<td>10%</td>
</tr>
<tr>
<td>Collected HW</td>
<td>10%</td>
</tr>
<tr>
<td>Programming Assignments</td>
<td>20%</td>
</tr>
<tr>
<td>Midterm Exam 1</td>
<td>20%</td>
</tr>
<tr>
<td>Midterm Exam 2</td>
<td>20%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>20%</td>
</tr>
</tbody>
</table>

A: 90+, B=80-89, C=70-79, D=60-69, F<60
Special Notes

Homework

- “An Aggie does not lie, cheat, or steal, or tolerate those who do.”
- No sharing of computer programs or excerpts
- Programs will be scanned for similarities
- HW due at the beginning of classes and no late HW is accepted.

Exams

- Bring your own calculator
- One double-sided 8 ½” X 11” sheet of notes is allowed
Concept of Numerical Methods

- Methods to reformulate math problems so that direct use of arithmetic operations is possible.
- Example – bungee-jumping

\[
d\frac{v}{dt} = g - \frac{c_d}{m} v^2
\]
Analytical solution

\[ v(t) = \sqrt{\frac{g m}{c_d}} \tanh \left( \sqrt{\frac{g c_d}{m}} t \right) \]

Numerical solution

\[ \frac{dv}{dt} \approx \frac{\Delta v}{\Delta t} = \frac{v(t_{i+1}) - v(t_i)}{t_{i+1} - t_i} \]

\[ v(t_{i+1}) = v(t_i) + \Delta t \left[ g - \frac{c_d}{m} v(t_i)^2 \right] \]

\[ \frac{dv}{dt} = g - \frac{c_d}{m} v^2, \ v(0) = 0 \]

Note:

\[ \tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}} \]
Example: Finding Trend via Data
Building Blocks

- Root finding
  \( f(x) = 0, \text{ find } x \)
- Matrices and linear algebra
  \( Ax = B, \text{ solve for } x \)
- Data modeling
  - Regression analysis
  - Interpolation
  - Statistics
- Calculus
  - Differentiation
    evaluate \( \frac{df}{dx} \)
  - Integration
    evaluate \( \int_a^b f(x) \)
- Differential Equations
History of Computers

Abacus, ~ 2nd century BC
Difference Engine, ~1800 AD
ENIAC, ~1940 AD
IBM-PC, ~1980 AD
What is a programming language?
- An artificial language designed to express computations that can be performed by a machine (computer).
- Computers only know binary machine code so source code needs to be translated into binary code before they can be executed.

Assembly language (low-level language)
- Machine specific
- Can be directly translated to binary code

High-level languages
- Traditionally compiled languages
  - C, C++, Fortran, etc.
- Traditionally interpreted languages
  - BASIC, perl, Matlab, etc.
Running a Computer Program

- Compiled languages
  - Source code prog.c → Compiler → Object file prog.o → Linker → Executable file prog.exe

- Interpreted languages
  - Source code prog.bas → Interpreter → Output
    - Data
Matlab

- **Matrix Laboratory**
  - Represent both a software and a programming language
- Widely used in science and engineering for numerical calculation, simulation and analysis
  - Relatively easy to learn and use
  - Powerful plotting, graphics and data manipulation
  - MATLAB programming language
  - Toolboxes (neural network, statistics, etc.)
- GUI (graphical user interface) builder
Matlab Programming Environment

- Intelligent text editor
- Built-in debugger
- Extensive online help
Matlab Graphics

- Extensive suite of publication-quality plot types
- Easy automation using program scripts
- Industry standard for scientific and engineering plotting
Getting Start with Matlab

Alternative Software – GNU Octave

- Free
- Mostly compatible with Matlab
- Available for Windows, Linux and Mac
- Excellent documentation
- Windows installer available at: http://octave.sourceforge.net/
- Graphical user interface available at: http://qtoctave.wordpress.com/download/
GNU Octave with GUI

Starting Octave...
GNU Octave, version 3.2.4
Copyright (C) 2009 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "i686-pc-mingw32".

Additional information about Octave is available at http://www.octave.org.
Please contribute if you find this software useful.
For more information, visit http://www.octave.org/help-wanted.html

Report bugs to <bug@octave.org> (but first, please read
http://www.octave.org/bugs.html to learn how to write a helpful report).

For information about changes from previous versions, type `news`.

>>>