**General Objective:** Develop abilities to

- Write mathematical models of natural and engineered systems
- Solve these models using numerical techniques

Numerical methods implemented using
- Spreadsheets
- A structured programming language

---

**Why numerical methods?**

1. Ready access to powerful computational capabilities
2. Extremely powerful problem-solving tools. Allow you to handle
   - Large systems of equations
   - Non-linearities
   - Complicated geometries
3. Help you intelligently use "canned" programs involving numerical methods
   - Understand errors inherent in numerical calculations
   - Select methods
4. Allow you to solve problems for which canned programs are unavailable

5. Learn to use computers by writing programs

6. Reinforce your understanding of mathematics

Course Prerequisites

• Calculus of several variables
• Familiarity with personal computers
• Co-requisite: Introduction to differential equations

Instructor
Ben Koopman
322 Black
bkoop@ufl.edu
352-392-7104
See “Contacts” at class Web site for the latest info

Instructor
Ben Koopman
322 Black
bkoop@ufl.edu
352-392-7104
See “Contacts” at class Web site for the latest info

TA
Wei Bai
318 Black
wbai.825@gmail.com

Reader
Yuan Gan
318 Black Hall
ilomse@ufl.edu

Text

Recommended Reading

Software
Microsoft Excel (and a laptop that runs it)
- Excel must include Solver and Excel Visual Basic for Applications

➤ You will not be able to complete this course without a laptop running Excel with Solver and Excel VBA

Backup media (email, Jump drive, external hard drive, Internet storage)

Class Schedule... At the Web site
Lab Problems

Objective
- Introduce specific techniques needed for programming or computational methods

Implementation
- Assigned where appropriate
-Due at 9:00 pm on the day indicated
- 179 min grace period to allow for computer glitches

Lab grading (each problem)
- 1 pts - Complete solution with correct answer
- 0 pts - Incomplete or incorrect solution

Late labs accepted up to 1 week past the due date (maximum 2/3 credit)
In-class Problems

Objective

► Allow you to apply new information while it is still fresh in your mind

Implementation

► Problems are given in the lectures

► Time is provided during the class period for you to work the problems

► Tentative solutions are presented by volunteers during the class period. (Everyone gets to volunteer at least once.)

Take-home Problems

Objectives

► Provide opportunities to integrate modeling, numerical methods, & programming

► Provide opportunities to develop problem-solving skills

► Widen scope of resources used
Problems are given in the lectures

- Time is allowed in class for you to begin work
  - Complete the problem at home

- A tentative solution will be presented by a volunteer during the next class period.

Submittal of In-class and Take-home Problem Solutions

- Due 6 days after a lecture topic is completed (9:00 pm)
- 179 min grace period to allow for computer glitches

- Late problem sets are accepted up to 1 week past the due date (maximum 2/3 credit)

- Grading of In-class and Take-home Problems
  - 3 pts - Complete solution with correct answer
  - 2 pts - Complete solution with incorrect answer
  - 1 pts - Incomplete solution, but with significant effort
  - 0 pts - Negligible effort to obtain solution
Exams

• Exam 1: Covers the first four Visual Basic for Applications Lessons

• Exams 2 and Final: Cover subject material
  (Each 2 hr; 50% conceptual, 50% on computer)

• Date of final: Monday, 12 December, 10:00 am–12:00 noon

→ Exams are cumulative

No early or late finals will be given

Grading

Weights

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab Problems</td>
<td>5%</td>
</tr>
<tr>
<td>In-class and Take-home Problems</td>
<td>30%</td>
</tr>
<tr>
<td>Exam 1</td>
<td>10%</td>
</tr>
<tr>
<td>Exam 2</td>
<td>20%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>30%</td>
</tr>
<tr>
<td>Class participation</td>
<td>5%</td>
</tr>
</tbody>
</table>

(based on team points given for presentations beyond the 1st)

Grading scale

<table>
<thead>
<tr>
<th>Grade</th>
<th>4.00 grade pts</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4.00</td>
</tr>
<tr>
<td>A-</td>
<td>3.67</td>
</tr>
<tr>
<td>A+</td>
<td>3.33</td>
</tr>
<tr>
<td>B</td>
<td>3.00</td>
</tr>
<tr>
<td>B-</td>
<td>2.67</td>
</tr>
<tr>
<td>B+</td>
<td>2.33</td>
</tr>
<tr>
<td>C</td>
<td>2.00</td>
</tr>
<tr>
<td>C-</td>
<td>1.67</td>
</tr>
<tr>
<td>D</td>
<td>1.33</td>
</tr>
<tr>
<td>D-</td>
<td>1.00</td>
</tr>
<tr>
<td>E</td>
<td>0.67</td>
</tr>
</tbody>
</table>

Grading scale (based on overall course average)
Collaboration

• Helping each other learn the subject material is encouraged
• However, you must prepare your lab and homework assignments individually

Plagiarism

• Copying another student’s work or the instructor’s work will be penalized