I. Numerical Computing

A. Lectures 1-3: Foundations of Numerical Computing

Things you should know

Lecture 1 – Intro to numerical computing

- Understand difference and pros/cons of analytical versus numerical solutions

Lecture 2 – Introduction to Excel and Python

- The four basic elements of a computer: input, output, CPU, memory
- What a spreadsheet is
- What a programming language is

Lecture 3 – Units, Data Types, Error

- Concept of dimensional homogeneity
- Data types: boolean, integer, float, string
- How data types are stored in memory
- Understand the origin of finite precision errors

Things you should be able to do

Lecture 1 – Intro to numerical computing

- Classify math problems: linear/nonlinear, algebraic/differential, single/system, coupled/uncoupled

Lecture 2 – Introduction to Excel and Python

- Excel: navigation, copy/paste/fill, formatting, arithmetic, formulas, locked cells
- Python: console I/O, define variables, arithmetic, use comments
- Use Excel and Python help docs

Lecture 3 – Units, Data Types, Error

- Conversions between "Big 13" units (kg, m, s, N, lbm, ft, hr, lbf, slug, K, °C, °F, °R)
- Put an equation in consistent units for use in a numerical tool
- Define variables of different data types in Excel and Python

B. Lectures 4-5: Structured Programming

Things you should know

Lecture 4 – Functions

- What a function is (map, formula, procedure, abstraction) and when it is useful (name, organize, re-use code)
- Difference between local and global variable scope

Lecture 5 – Conditional Statements

- What a conditional (`if`, `if...else`) statement is and when it is useful (branch code, make decisions)
- The difference between the comparison (`==`) and the assignment operator (`=`)

Things you should be able to do

Lecture 4 – Functions

- Call a function in Excel
- Import modules in Python
- Call a function in Python
- Define a function in Python (with or without arguments, with or without return statements)
- Use local and global variables

Lecture 5 – Conditional Statements

- Use conditional and logical functions in Excel
- Use conditional (`if`, `if...else`) statements, logical (`and`, `or`, `not`) operators and comparison operators (`==`, `>`, `<`, ...) in Python

C. Lectures 6-9: Loops and Arrays

Things you should know

Lecture 6 – Loops

- What a loop is and when it is useful (repeat code, fill arrays, ...)
- What an infinite loop is and why it occurs

Lecture 7 – Arrays

- How an array relates to a vector and a matrix
- What an array variable is ("ice cube tray")
- How Excel stores arrays
- Difference between a Python list, Python tuple

and Numpy array

Lecture 8 – File I/O and Plotting

- How text files are formatted (header, delimiter, values) and what a file extension means (e.g. `.txt`, `.dat`, `.csv`)

Lecture 9 – Debugging

- Define the three types of bugs: syntax, execution, logical
- Identify a bug as either syntax, execution or logical

Things you should be able to do

Lecture 6 – Loops

- Define and use `for` loops and `while` loops

Lecture 7 – Arrays

- Define and use arrays, lists and tuples
- Access elements of arrays, lists and tuples in Python using indices and slicing
- Use Numpy arrays to store data and do element-wise arithmetic
- Use a loop to fill an array

Lecture 8 – File I/O and Plotting

- Import/export data into Excel from/to a text file (`.txt`, `.dat`, `.csv`)
- Make a scatter plot in Excel
- Use `loadtxt` and `savetxt` to read from and write to a text file (`.txt`, `.dat`, `.csv`) in Python
- Make, format and save plots in Python

Lecture 9 – Debugging

- Use the scientific method (observation, hypothesis, experiment) to debug a faulty Python code or Excel worksheet