CS 1132 lecture 1

I. Motivation
   a. Quickly get up-to-speed with Matlab programming environment
      i. Already know another language
      ii. Grad student who needs to learn technical computing before doing research
   b. Example of utility: visually check work
      i. Demo: 4-bar linkage

II. Instructor
   a. Physicist
   b. LIGO (simulation, visualization)
   c. SpaceX (engineers design autonomous systems)

III. Topics, goals
   a. Translate a problem's solution into an algorithm
      i. Algorithm: unambiguous, step-by-step procedure for doing something
   b. Implement algorithms in Matlab syntax
   c. Visualize data and simulations
   d. Poll: students' goals
   e. Topics
      i. Matlab environment, built-in functions
      ii. Arrays (vector, matrix)
      iii. Vectorized computation
      iv. Control flow (if/else, loops)
      v. User-defined functions
      vi. Strings and cell arrays
      vii. Graphics
      viii. Input/output (files)
   f. Programming fundamentals (requires practice)
      i. Top-down design
      ii. Modular development (to reduce redundancy)
      iii. Useful documentation
         1. Distinguish "what" from "how"
      iv. Thorough testing
         1. How can you be confident in your results when no one can give you the "right answer"?

IV. Syllabus
   a. Learning components
      i. Read textbook, watch videos, complete activities
      ii. Attend lectures (7 wks), take notes, participate
      iii. Attend lab, complete lab exercises
      iv. Complete programming assignments
      v. Ask questions in office and consulting hours, or on discussion board
   b. Assessment
      i. Feedback loop to improve learning

c. Assignments
    i. Resubmission allowed after feedback returned
    ii. Late submissions (within 24hr) penalized
d. Test
    i. May replace with a second test
e. "S" requires mastery of material (course score above 85%)
f. Alternatives
    i. CS 1112: more beginner-friendly at start
g. Academic integrity
    i. End product isn't valuable; experience producing it is

V. Demo
a. Course website
b. Matlab interface
    i. Command window
    ii. Workspace window
    iii. Files window
c. Built-in functions
d. Variables
e. Example script

VI. Script input, output
a. Most computation follows pattern: gather inputs, perform calculations, produce outputs
b. input() function
c. Prompt in single quotes
d. Assign result to variable
e. disp() function

VII. Example: change in sphere area

```
radius= input('Enter radius [mi]: ');




area= 4*pi*r^2;






disp('Surface area [mi^2]: ')
disp(area)
```

VIII. Program development tips

a. Know what is given (inputs, assumptions)
b. Be goal-oriented
    i. Write final output statements
    ii. Work backwards
c. If you don't have a value you need, make up a name for it
    i. Work backwards to compute its value
    ii. Helps break down steps