The tests are cumulative. Here are some sample test questions, mostly from old tests. Each question is put in the first lecture in which you have learned all the topics to answer it. The course does not cover Lectures 17, 18, 26, 32, and 40, so none of the questions use that material. Some lectures are listed with no questions; the material in those lectures is used in questions listed in later lectures.

Parts of the questions listed in lectures 37 and 38 can be done in lectures 33 and 35, which are subject to test 4. To make sure they are not overlooked, I have duplicated those parts of the questions and put them in the earlier lectures.

Part I: Matlab and Solving Equations

Lecture 1: Vectors, Functions, and Plots in Matlab

Lecture 2: Matlab Programs

1. Write a well-commented MATLAB **script** program to plot the functions $f(x) = x + \sin(x)$ and $g(x) = x^2$ on the same graph, on the interval $[1, 7]$.

Lecture 3: Newton's Method and Loops

1. Write a well-commented MATLAB **function** program that calculates the sum of the squares of the first n integers.

2.
   - For $f(x) = 4x^2 - 2$, do 2 iterations of Newton's method, starting with $x_0 = 1$.
   - Formulate the residual and the error of your final approximation.

3. Write a well-commented MATLAB **script** program to plot the function

$$f(x) = \sum_{k=1}^{50} \sqrt{k} \sin(2\pi k x)$$

   on the interval $x \in [0, 1]$.

Lecture 4: Controlling Error and Conditional Statements

1. The sequence of numbers

$$\cos(1), \cos(2), \ldots, \cos(999), \cos(1000)$$

   has some positive elements and some negative ones. Write a well-commented MATLAB **script** program that calculates the sum of the positive elements.

2. Write a well-commented MATLAB **function** program to do Newton's method for a function f until $|f(x)| < tol$. Let f, f', x_0 and tol be the inputs and the final x be the output.

Lecture 5: The Bisection Method and Locating Roots

1. The function $f(x) = 3x^2 - 30$ is continuous and $f(-3) < 0 < f(5)$, so it has a zero on the interval $[-3, 5]$.
   - Perform 3 iterations of the bisection method to narrow down this interval.
   - How many iterations will it take before you know the zero within 10^{-7}?

2. Write a well-commented Matlab **function** program to do n steps of the bisection method for a function f with starting interval $[a, b]$. If $|f(x)| > tol$ after n iterations, print a warning. Let f, a, b, n, and tol be the inputs and the final x be the output.

3. Write a well-commented Matlab **function** program that will find the roots of a function f on an interval $[a, b]$.

Lecture 6: Secant Methods

1. Write a well-commented Matlab **function** program to do n steps of the Regula Falsi method for a function f with starting interval $[a, b]$. Let f, a, b and n be the inputs and the final x be the output.

2.
   - For $f(x) = 4x^2 - 1$, do 2 iterations of the secant method method, starting with $x_0 = 0$ and $x_1 = 1$.
   - Formulate the residual and the error of your final approximation.

Lecture 7: Symbolic Computations

Part II: Linear Algebra

Lecture 8: Matrices and Matrix Operations in Matlab

Lecture 9: Introduction to Linear Systems

Lecture 10: Some Facts About Linear Systems

1. Write a well-commented Matlab **function** program that takes an input n, produces a random $n \times n$ matrix A and random vector $\bar{b}$, solves $A\bar{x} = \bar{b}$ (using the built in command) and outputs the norm of the residual.

Lecture 11: Accuracy, Condition Numbers and Pivoting

1. What is the condition number of a matrix? How do you find it with Matlab? What are the implications of the condition number when solving a linear system? What is the engineering solution to a problem with a bad condition number?

Lecture 12: LU Decomposition

1. (a) Using pivoting if appropriate, find the LU decomposition of

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 5 \end{bmatrix}.$$

   (b) Using your LU decomposition, solve $A\mathbf{x} = \mathbf{b}$ where $\mathbf{b} = \begin{bmatrix} 7 \\ 11 \end{bmatrix}$.

2. Write a well-commented Matlab **function** program that solves the linear systems $A\mathbf{x}_1 = \mathbf{b}_1$ and $A\mathbf{x}_2 = \mathbf{b}_2$ using LU decomposition. Let `A`, `b1`, and `b2` be the inputs and `x1` and `x2` be the outputs.

3. Write a well-commented Matlab **function** program that solves a linear system $A\mathbf{x} = \mathbf{b}$ using LU decomposition. Let A, $\mathbf{b}$ and tol be the inputs and $\mathbf{x}$ the output. If the error (residual) is not less than tol, then display a warning.

Lecture 13: Nonlinear Systems - Newton's Method

1. Write a well-commented Matlab **script** program that will use Newton's method to find a solution to the system of equations

$$\begin{aligned} x^3 - 7y^2 + 1 &= 0 \\ 5y^3 + x + 2z - 2 &= 0 \\ 5z^3 + x - y^2 - 3 &= 0 \end{aligned}$$

starting from the initial guess $(x, y, z) = (1, 1, 1)$.

Lecture 14: Eigenvalues and Eigenvectors

1. Find the eigenvalues and eigenvectors of the matrix

$$A = \begin{bmatrix} -2 & 4 \\ 1 & 1 \end{bmatrix}.$$

Lecture 15: An Application of Eigenvectors: Vibrational Modes

Lecture 16: Numerical Methods for Eigenvalues

1. (a) Find the eigenvalues and eigenvectors of the matrix

$$A = \begin{bmatrix} -2 & 4 \\ 1 & 1 \end{bmatrix}.$$

   (b) Perform one iteration of the power method, starting with $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$. What approximate eigenvalue and eigenvector did you get? If you kept iterating, what would you get eventually?

   (c) Using pivoting if appropriate, find the LU decomposition of A.

   (d) Using your LU decomposition, perform one iteration of the inverse power method, starting with $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$. What approximate eigenvalue and eigenvector did you get? If you kept iterating, what would you get eventually?

2. Write a well-commented Matlab **function** program to do n iterations of the Power Method. Let the matrix A and n be inputs and let `[e v]` (the eigenvalue and eigenvector) be the outputs.

3. Write a well-commented Matlab **function** program `myipmtol` that inputs a matrix and a tolerance, applies the inverse power method until the change in the vector is less than the tolerance, and outputs the estimated eigenvalue and eigenvector.

Part III: Functions and Data

Lecture 19: Polynomial and Spline Interpolation

Lecture 20: Least Squares Fitting: Noisy Data

1. Your friend has a data set represented by vectors $\mathbf{x}$ and $\mathbf{y}$ and is considering using a polynomial interpolation, a spline interpolation, or a least squares approximation.

(a) For polynomial interpolation, explain to them:
- what it is,
- how you get it in MATLAB, and
- in what situations it is better than the other methods.

(b) Do the same for spline interpolation.

(c) Do the same for least squares approximation.

Lecture 21: Integration: Left, Right and Trapezoid Rules

1. Write a well-commented MATLAB **function** program to do the Trapezoid Rule for integration of a function given by data. Let the inputs be vectors `x` and `y`, representing a list of points $(x_i, y_i) = (x_i, f(x_i))$. Assume $x_i < x_{i+1}$ but do not assume the x values are evenly spaced.

Lecture 22: Integration: Midpoint and Simpson's Rules

1. Estimate the integral $\int_{-3}^{1} x^2\,dx$ using L_4, R_4, T_4 and S_4. Calculate the exact value and the errors of each of the approximations.
2. Approximate the integral $\int_0^\pi \sin x\,dx$ using M_4 and S_4. Which do you expect to be more accurate?
3. Write a well-commented MATLAB **function** program to do the midpoint method for integration. Let the inputs be the function f, the endpoints a, b and the number of subintervals n.
4. Write a well-commented MATLAB **function** program to do Simpson's rule for integration. Let the inputs be the function f, the endpoints a, b and the number of subintervals n.

Lecture 23: Plotting Functions of Two Variables

Lecture 24: Double Integrals for Rectangles

1. Write a well-commented MATLAB **function** program to do the four-corners method for integration of a function $f(x, y)$ on a rectangle $a \le x \le b$, $c \le y \le d$ using m subintervals in x and n subintervals in y. Let the inputs be `(f,a,b,c,d,m,n)`.
2. Write a well-commented MATLAB **function** program to do the center-point method for integration of a function $f(x, y)$ on a rectangle $a \le x \le b$, $c \le y \le d$ using m subintervals in x and n subintervals in y. Let the inputs be `(f,a,b,c,d,m,n)`. Include comments.
3. Complete the program below. Include comments in your code.

```
function I = mycenterright(f,a,b,c,d,m,n)
% Computes an approximate integral of a function of two
% variables f(x,y) on a rectangle.
% Splits into small rectangles and on each one evaluates f at
% the center of the right edge.
%
% Inputs:
%    f -- the function
%    a,b -- define the interval in x, namely a<x<b.
%    c,d -- define the interval in y, namely c<y<d.
%    m -- the number of (evenly spaced) intervals to use in x.
```

```
%    n -- the number of (evenly spaced) intervals to use in y.
% Output: The approximate value of the integral.
```

Lecture 25: Double Integrals for Non-rectangles

1. Describe and give formulas for 2 methods to approximate double integrals based on triangles.

Lecture 27: Numerical Differentiation

1. Suppose you know $f(1) = 2$, $f(3/2) = 3$, $f(2) = 5$, and $f(5/2) = 4$. Compute:
   - the forward difference approximation of $f'(1)$,
   - the backward difference approximation of $f'(3/2)$,
   - the centered difference approximation of $f'(2)$, and
   - the centered difference approximation of $f''(2)$.

2. Complete the program below. Include comments in your code.

```
function df = myderivative(f,a,b,n)
% Computes an approximate derivative of f on an interval.
%
% Uses the forward difference at the first point,
% the backward difference at the last point,
% and the centered difference at all points in between.
%
% Inputs:
%    f -- the function
%    a,b -- define the interval in x, namely a<=x<=b.
%    n -- the number of (evenly spaced) points to use in x.
% Output: Vector of length n with approximate values of f'(x).
```

Lecture 28: The Main Sources of Error

1. Compare and contrast *truncation* error and *loss-of-precision* (also known as loss-of significance) error. Illustrate each with an example.

2. For the function $f(x) = x^2$ you need to find $f'(5)$, but you can't remember whether $f'(x) = 2x$ or $f'(x) = x^3/3$. You remember that

$$f'(5) = \lim_{h\to 0} \frac{f(5+h) - f(5)}{h},$$

so you try:

```
> h=10^(-50)
> ((5+h)^2-5^2)/h
```

which gives the result: `ans = 0`. Is this the correct answer? Explain what happened and why.

3. Explain what would happen if you ran the following MATLAB commands:

```
> format long
> for i=1:30
>     x=10^i+pi
>     mypi=x-10^i
>     error=(pi-mypi)/pi
> end
```

Lecture 29: Reduction of Higher Order Equations to Systems

1. Write the IVP: $\theta'' + .5\theta' + \sin\theta = \sin 2t$, $\theta(0) = 1$, $\theta'(0) = 0$ as a system of first order equations. Give all the MATLAB commands needed to solve this IVP on the interval $0 \leq t \leq 10$.

Lecture 30: Euler Methods

1. Write a well-commented MATLAB **function** program to do n steps of the Euler method for a differential equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t)$, on the time interval $[a, b]$ with $\mathbf{x}(a) = \mathbf{x}_0$. Let the first line be:
`function [T, X] = myeuler(f,x0,a,b,n)`.

Lecture 31: Higher Order Methods

1. Describe RK45. What is the command for it in MATLAB?

2. What is variable step size? How is it implemented RK45?

3. (a) Write a well-commented MATLAB **function** program to do n steps of the modified Euler method for a differential equation $\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x})$, on the time interval $[a, b]$ with $\mathbf{x}(a) = \mathbf{x}_0$. Let the first line be:
`function [T, X] = mymodeuler(f,x0,a,b,n)`.

   (b) Give the sequence of MATLAB commands needed to use your function `mymodeuler` to solve the ODE $s''(r) = \cos(r) + s'(r)$ with $s(3) = 5$ and $s'(3) = 4$ on the interval $3 \leq r \leq 11$.

   (c) In what way(s) is MATLAB's `ode45` function better than your function `mymodeuler`?

4. Explain why order matters in engineering problems.

Lecture 33: ODE Boundary Value Problems and Finite Differences

1. By replacing y'' with a central difference, derive the finite difference equation for the boundary value problem

$$y'' = x \quad \text{on } [1, 3] \quad \text{with} \quad y(1) = 2 \quad \text{and} \quad y(3) = 5$$

using 2 subintervals. Solve the equation and plot the solution.

Lecture 34: Finite Difference Method – Nonlinear ODE

Lecture 35: Parabolic PDEs - Explicit Method

1. Derive the explicit finite difference equations for solving the heat/diffusion equation $u_t = cu_{xx}$ on the interval $x \in [0, L]$ with boundary conditions $u(0, t) = a$, $u(L, t) = b$, and $u(x, 0) = f(x)$.

Lecture 36: Solution Instability for the Explicit Method

Lecture 37: Implicit Methods

1. (a) Derive the explicit finite difference equations for solving the heat/diffusion equation $u_t = cu_{xx}$ on the interval $x \in [0, L]$ with boundary conditions $u(0, t) = a$, $u(L, t) = b$, and $u(x, 0) = f(x)$.
   (b) When and why does the explicit finite difference method for the heat/diffusion equation become unstable?
   (c) Derive the implicit finite difference equations for solving the heat/diffusion equation $u_t = cu_{xx}$.

2. For the heat/diffusion equation $u_t = cu_{xx}$ we learned an explicit method, an implicit method, and (in less detail) the Crank-Nicholson method.
   - Compare and contrast the explicit and implicit methods. Which would you recommend, and why?
   - What is the Crank-Nicholson method? Would you recommend it over the other two? Why?

Lecture 38: Insulated Boundary Conditions

1. (a) Derive the explicit finite difference equations for solving the heat/diffusion equation $u_t = cu_{xx}$ on the interval $x \in [0, L]$ with boundary conditions $u(0, t) = a$, $u(L, t) = b$, and $u(x, 0) = f(x)$.
   (b) Explain how to incorporate an insulated boundary at $x = L$.

2. (a) By replacing y'' with a central difference, derive the finite difference equation for the boundary value problem
$$y'' = x \quad \text{on } [1, 3] \quad \text{with} \quad y(1) = 2 \quad \text{and} \quad y(3) = 5$$
   using 2 subintervals. Solve the equation and plot the solution.
   (b) Change the boundary condition $y(3) = 5$ to the insulated condition $y'(3) = 0$. Derive the new equations, solve them, and plot the solution.

Lecture 39: Finite Difference Method for Elliptic PDEs

1. (a) Set up the finite difference equations for the BVP: $u_{xx} + u_{yy} = f(x, y)$, on the rectangle $0 \le x \le a$ and $0 \le y \le b$, with $u = 0$ on all the boundaries. Explain how the difference equations could be solved as a linear system.
   (b) Explain how to incorporate an insulated boundary at $x = a$.

Lecture 41: Finite Elements

Lecture 42: Determining Internal Node Values

1. A finite element solution to a differential equation in two dimensions is of the form
$$U(x, y) = \sum_{j=1}^{n} C_j \Phi_j(x, y)\,.$$
   (a) Explain what the Φ_j are and how you construct/ find/ solve for them.
   (b) Explain what the C_j are and how you construct/ find/ solve for them.

Last modified March 30, 2017.