Math 405/607E: Numerical Solution of Differential Equations
Final Exam, December 2016

Family Name: ____________________ Given Name: ____________________

Signature: ____________________ Student Number: ____________________

Course: 405 or 607E: ____________________

Question:	1	2	3	4	5	6	7	8	9	Total
Points:	16	8	6	6	6	6	8	6	8	70
Score:										

Instructions

- You have 150 minutes to write this exam.
- This exam contains 5 pages including this cover page. Ensure you have all the pages.
- Some parts of this exam are multiple choice: for these you do not need to show any calculations unless specifically asked.
- For other problems, please write your answers on an exam booklet.
- For short answer questions, write solutions to the questions in the space provided below each question. If you require more space, label clearly where you have written your solution.
- No aids are allowed. No notes, books, programmable calculators etc. Simple calculators are acceptable, but should not be needed.

16 marks 1. (a) True False A singular matrix has a zero condition number.

True False A *composite quadrature method* is a method that adaptively chooses the best grid to approximate an integral.

True False The Barycentric Lagrange formula is one possible technique for evaluating an interpolating polynomial.

True False Radial basis function methods approximate a function $u(\vec{x})$ as a weighted sum of radial functions: $u(\vec{x}) = \sum_i^N w_i \, \phi(||\vec{x} - \vec{x}_i||_2)$.

True False Even if the grid points $x_0, x_1, \ldots, x_n$ are not distinct, the interpolating polynomial of degree n is still unique.

True False When doing spectral methods, the second derivative can be computed in the "frequency domain" by $\widehat{u_{xx}} = k^2 \hat{u}$.

(b) A numerical method for a differential equation is ____________________ if the global error approaches zero in the limit as $h \to 0$.

(c) A numerical method is *consistent* with a differential equation if the ____________________ goes to zero as $h \to 0$.

(d) A well- ____________________ *problem* is a problem where small perturbations do not have a large effect on the exact solution.

(e) An ____________________ numerical method is one that is highly sensitive to perturbation.

(f) State (or derive if you wish) the *variational problem* corresponding to the Poisson problem $-u_{xx} = f(x)$ on the domain $x \in [a, b]$.

(g) Sketch a "hat function" appropriate for use as one of the *basis functions* in a finite element method based on piecewise linear test functions. Label your diagram.

(h) Give the cardinal polynomial $L_{n,k}(x)$

8 marks 2. The following matrix problems approximate the differential equation $u_{xx} = f$. The first and last row of each matrix is missing; fill in to approximate the indicated boundary conditions. You may need to add values on the right-hand side as well.

Here $\vec{u}$ is a vector of point-wise values of u sampled on the interval $[a, b]$ (the grid is not specified and could be different in each part).

(a) **Periodic** with $u(a) = u(b)$:

$$\frac{1}{h^2}\begin{bmatrix} & & & & & & \\ \hdashline 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & 1 & -2 & 1 & \\ & & & & 1 & -2 & 1 \\ \hdashline & & & & & & \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} \\ \hdashline f_1 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} \\ \hdashline \\ \end{bmatrix}$$

(b) **Dirichlet** with $u(a) = 0$ and $u(b) = 6$:

$$\frac{1}{h^2}\begin{bmatrix} & & & & & & \\ \hdashline 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & 1 & -2 & 1 & \\ & & & & 1 & -2 & 1 \\ \hdashline & & & & & & \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} \\ \hdashline f_1 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} \\ \hdashline \\ \end{bmatrix}$$

(c) **Neumann and. . . an integral!** where $u'(a) = 0$ and $\int_a^b u(x)dx = 7$:

$$\frac{1}{h^2}\begin{bmatrix} & & & & & & \\ \hdashline 1 & -2 & 1 & & & & \\ & 1 & -2 & 1 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & 1 & -2 & 1 & \\ & & & & 1 & -2 & 1 \\ \hdashline & & & & & & \end{bmatrix}\begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \\ u_N \end{bmatrix} = \begin{bmatrix} \\ \hdashline f_1 \\ f_2 \\ \vdots \\ f_{N-2} \\ f_{N-1} \\ \hdashline \\ \end{bmatrix}$$

Solutions for the following problems should be written in exam booklets.

6 marks 3. Let P (a number) be the solution to some problem. Suppose we have a numerical method $M(h)$ which approximates P using a small numerical parameter h (such as a grid spacing). Furthermore, we know an error analysis: $M(h) = P + Ch^2 + O(h^3)$, where C is an unknown fixed constant independent of h.

Using only the values of $M(h)$ and $M(h/3)$, determine a new more-accurate method for find P. Give the error term (in Big Oh notation).

Suppose that $M(1) = 2$ and $M(1/3) = 3$. What is your best guess for P?

6 marks 4. Let $A = \begin{bmatrix} 12 & 3 & 4 \\ 3 & 5 & 10 \\ 4 & 10 & 10 \end{bmatrix}$. In this problem, "$\times$" indicates values that we do not care about.

(a) Find an orthogonal matrix H such that $HA = \begin{bmatrix} \alpha & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \end{bmatrix}$. What is α?

(b) Find an orthogonal matrix Q such that $QA = \begin{bmatrix} 12 & 3 & 4 \\ \beta & \times & \times \\ 0 & \times & \times \end{bmatrix}$. What is β?

(c) How can we get a matrix $\begin{bmatrix} 12 & \times & \times \\ \beta & \times & \times \\ 0 & \times & \times \end{bmatrix}$ with the same eigenvalues as A?

6 marks 5. Consider the factorization of a matrix into $A = LL^T$, where L is a lower-triangular matrix. This is the *Cholesky factorization*; every *positive definite* matrix has one.

(a) Assuming we already have the Cholesky factorization, describe how to solve $Ax = b$ for x. Give the operation count (in "Big Oh" notation) for each step.

(b) *Colin's Bad Theorem: every positive definite matrix has only unit eigenvalues.*

This theorem ain't true; find the error in the following "proof":

> "Consider the Cholesky factorization $A = LL^T$. Now $A = LIL^T = LIL^{-1}$ where I is the identity matrix. Thus A is similar to the identity matrix. Therefore A has one eigenvalue, $\lambda = 1$ (of multiplicity n)."

6 marks 6. Give a finite difference discretization of each of these differential equations. Assume there is a uniform grid in x, given by x_i with grid spacing h and a uniform grid in t, given by t_n with grid spacing k.

(a) $u_{tt} + u_t = (u_{xx})^2$

(b) $(a(x)u_x)_x = f(x)$

(c) $u_{xt} = g(x)$, using forward differences in t and centered in x

8 marks 7. Consider the ODE $u_t = f(t,u)$ with initial condition $u(0) = u_0$. Which of the following methods are *consistent* and which are *zero-stable*?

(a) $u_{n+1} = u_n$ ("Colin's Method", three easy payments of \$39.99)

(b) $$\begin{cases} y = u_n + \frac{1}{2}kf(t_n, u_n) \\ u_{n+1} = y + \frac{1}{2}kf(t_n + \frac{1}{2}k, y) \end{cases}$$

(c) $u_{n+1} = 3u_n + 4kf(t_n, u_n)$

(d) $u_{n+1} = u_n + \frac{3}{2}kf(t_n, u_n) - \frac{1}{2}kf(t_{n-1}, u_{n-1})$

6 marks 8. For part (c) of the previous question, perform an absolute stability analysis. Assume that λ (in the test problem) is real (so that $z = k\lambda$ is real). Determine any restrictions on z for absolute stability.

For part (d) of the previous question, begin an absolute stability analysis and give two equations which must be satisfied for absolute stability. Note you *do not need to* find the restrictions on z.

8 marks 9. (a) State Newton's Method for solving the scalar algebraic equation $F(y) = 0$.

(b) Give the implicit Backward Euler method for numerically solving the scalar ODE $u' = f(u)$ with solution $u(t)$ with initial condition $u(0) = u_0$ using a stepsize h.

(c) We want to solve $u' = f(u)$ using the Backward Euler method. Suppose we (only) have two codes `myfcn(u)` and `myderiv(u)` which compute $f(u)$ and its derivative $f'(u)$ for a given input. We do not know anything else about $f(u)$.

Write an algorithm (e.g., in Matlab-esque or Pythonese "pseudocode") for taking a single step of Backward Euler (to advance from u_n to u_{n+1}).

Hint: your algorithm does not need *exactly* implement Backward Euler; it is enough to approximate u_{n+1} to within a tolerance of 10^{-12} (after all, u_{n+1} is only an approximation of the exact solution $u(t)$).