FAMILY NAME: ....
OTHER NAME(S): ....
STUDENT NUMBER: ....
SIGNATURE: ....

THE UNIVERSITY OF NEW SOUTH WALES

SCHOOL OF MATHEMATICS AND STATISTICS

Semester 1, 2013

MATH2089
Numerical Methods and Statistics

(1) TIME ALLOWED – 3 Hours

(2) TOTAL NUMBER OF QUESTIONS – 6

(3) ANSWER ALL QUESTIONS

(4) THE QUESTIONS ARE OF EQUAL VALUE

(5) THIS PAPER MAY **NOT** BE RETAINED BY THE CANDIDATE

(6) **ONLY** CALCULATORS WITH AN AFFIXED "UNSW APPROVED" STICKER MAY BE USED

(7) STATISTICAL FORMULAE ARE ATTACHED AT END OF PAPER
STATISTICAL TABLES ARE ATTACHED AT END OF PAPER

Part A – Numerical Methods consists of questions 1 – 3

Part B – Statistics consists of questions 4 – 6

Both parts must be answered

All answers must be written in ink. Except where they are expressly required pencils may only be used for drawing, sketching or graphical work.

Part A – Numerical Methods

1. Answer in a separate book marked Question 1

a) What are the values produced by the following MATLAB expressions (pay attention to the semi-colon at the end of the expressions):

i)
```
x = -1e+20;
ans1 = log(exp(x))
```
ii)
```
v = [2:-2:-2]
ans2 = v.^(1/2)./v
```
iii)
```
y = 0.01;
h = 1e-16;
ans3 = y+h == y
```

b) The computational complexity of some common operations with n by n matrices are given in Table 1.1

Operation	Flops
Matrix multiplication	$2n^3$
LU factorization	$\frac{2n^3}{3} + O(n^2)$
Cholesky factorization	$\frac{n^3}{3} + O(n^2)$
Back/forward substitution	$n^2 + O(n)$
Tridiagonal solve	$8n + O(1)$

Table 1.1: Flops for some operations with n by n matrices

i) Estimate the size of the largest linear system that can be solved in one hour on a 2.5 GHz quad core computer, where each core can do two floating point operations per clock cycle. Assume that the coefficient matrix has no special structure.

ii) If multiplying two $n \times n$ matrices takes 30 seconds, estimate how long it will take to solve an $n \times n$ symmetric positive definite linear system for 100 different right-hand-side vectors.

c) Consider a linear system $A\boldsymbol{x} = \boldsymbol{b}$ where you know A exactly, and you have the following output from some MATLAB code

```
nm = norm(A-A')
nm =
   9.8026e+01
rc = rcond(A)
rc =
   4.8677e-05
emin = min(eig(A))
emin =
   6.1427e-01
emax = max(eig(A))
emax =
   9.9990e+03
```

i) Is A symmetric?

ii) Obtain an estimate of the condition number $\kappa(A)$ using an appropriate norm. State which norm you used.

iii) If you want to have 6 significant figures of accuracy in a computed solution to $A\boldsymbol{x} = \boldsymbol{b}$, how accurate must the right-hand-side vector $\boldsymbol{b}$ be in terms of the relative error?

d) Consider the function $f(x) = e^x \sin(x) - 100$.

i) Formulate Newton's method for solving $f(x) = 0$.

ii) You are **given** that $f(x)$ has a simple zero at $x^* \approx 6.443$. If you use a starting value x_1 near x^*, what is the expected order of convergence for Newton's method?

iii) If the initial error $e_1 = |x_1 - x^*|$ is around 0.1, estimate the number of iterations k required to achieve the accuracy $e_{k+1} = |x_{k+1} - x^*| \leq 10^{-14}$.

2. Answer in a separate book marked Question 2

a) Let $f : [0, 5] \to \mathbb{R}$ be defined by

$$f(x) = \frac{8}{25\pi}\sqrt{x(5-x)}.$$

The function f is a probability density function so

$$I(f) = \int_0^5 f(x)\,\mathrm{d}x = 1.$$

Approximations to $I(f)$ were calculated using the Trapezoidal rule, Simpson's rule and the Gauss-Legendre rule, giving the following table of errors $E_N(f) = I(f) - Q_N(f)$:

N	E_N Trapezoidal	E_N Simpson	E_N Gauss-Legendre
32	5.8377e-03	2.2902e-03	-1.5288e-05
64	2.0659e-03	8.0865e-04	-1.9545e-06
128	7.3076e-04	2.8571e-04	-2.4714e-07
256	2.5842e-04	1.0098e-04	-3.1072e-08
512	9.1377e-05	3.5695e-05	-3.8953e-09
1024	3.2309e-05	1.2619e-05	-4.8762e-10

i) Find a linear transformation $x = \alpha + \beta z$ that maps $z \in [-1, 1]$ to $x \in [0, 5]$.

ii) **Given** the nodes $z_j, j = 1, \ldots, N$ and weights $w_j, j = 1, \ldots, N$ for the Gauss-Legendre rule for the interval $[-1, 1]$, how can you approximate $I(f)$?

iii) The error for Simpson's rule satisfies

$$E_N^{\mathrm{Simp}}(f) = O(N^{-4}), \tag{2.1}$$

provided that $f \in C^4([0, 5])$. You do **not** need to prove this.

A) Use (2.1) to estimate the ratio

$$\frac{E_N^{\mathrm{Simp}}(f)}{E_{2N}^{\mathrm{Simp}}(f)}. \tag{2.2}$$

B) Use the table of errors to estimate the ratio (2.2) when $N = 512$.

C) Is the table of errors consistent with the theoretical error estimate in (2.1)?

iv) You are **given** that for $x \in (0, 5)$,

$$f'(x) = \frac{4}{25\pi}\frac{5-2x}{\sqrt{x(5-x)}}.$$

Explain why $I(f)$ is difficult to approximate.

b) The motion of a damped mass-spring system is modelled by the initial value problem

$$my'' + cy' + ky = 0, \qquad y(1) = 2, \quad y'(1) = -1,$$

where $y(t)$ is the displacement of the block at time t, m is the mass of the block, c is the damping coefficient, and k is the spring constant. Consider here the case

$$m = 2, \quad c = 1, \quad \text{and} \quad k = 1.$$

i) What is the order of the differential equation?

ii) Convert this ordinary differential equation into a system

$$\boldsymbol{x}' = \boldsymbol{f}(t, \boldsymbol{x}), \quad \text{for } t > t_0,$$

of first order differential equations.

iii) What is the initial condition $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$?

iv) Write

- **EITHER** a MATLAB anonymous function `myode`
- **OR** a MATLAB function M-file `myode.m`

to evaluate the vector valued function $\boldsymbol{f}(t, \boldsymbol{x})$.

v) Use Euler's method with a step of $h = 0.2$ to estimate $\boldsymbol{x}(1.2)$.

3. Answer in a separate book marked Question 3

Fick's second law predicts how diffusion causes the concentration $u(x, y)$ of a chemical to change with position $(x, y) \in \Omega$. The steady state version of Fick's second law (without interior sources of the chemical) is Laplace's equation

$$\frac{\partial^2 u(x,y)}{\partial x^2} + \frac{\partial^2 u(x,y)}{\partial y^2} = 0. \tag{3.1}$$

Consider the rectangular domain

$$\Omega = \{(x, y) \in \mathbb{R}^2 \; : \; 0 \le x \le 2, \; 0 \le y \le 1\},$$

and discretize it using $h = 1/n$ and

$$\begin{cases} x_i = ih & \text{for } i = 0, 1, \ldots, 2n, \\ y_j = jh & \text{for } j = 0, 1, \ldots, n. \end{cases}$$

This is illustrated in Figure 3.1 for $n = 5$.

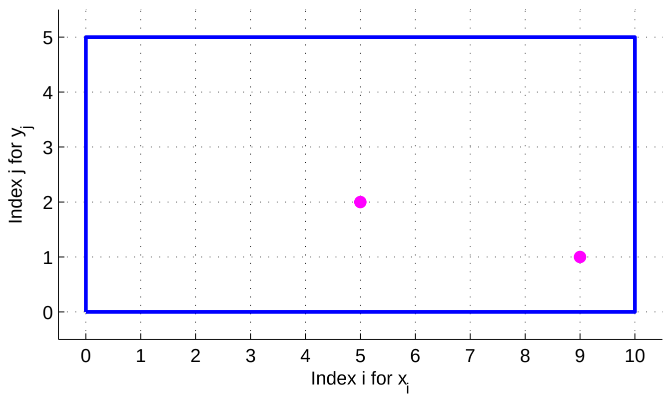

Figure 3.1: Discretization of domain for $n = 5$ and grid points for part d)

a) What extra information is needed to completely specify this problem?

b) You are **given** the following standard finite difference approximations for a function f of **one** variable:

$$\begin{aligned} f'(x) &= \frac{f(x+h) - f(x)}{h} + O(h), \\ f'(x) &= \frac{f(x+h) - f(x-h)}{2h} + O(h^2), \\ f''(x) &= \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2). \end{aligned}$$

Let $u_{i,j}$ denote the approximation to the value $u(x_i, y_j)$ of concentration at the grid point (x_i, y_j). Give central difference approximations of accuracy $O(h^2)$ to the following derivatives at the point (x_i, y_j)

i) $\dfrac{\partial^2 u(x,y)}{\partial x^2}$ ii) $\dfrac{\partial^2 u(x,y)}{\partial y^2}$

c) Using the finite difference approximations from the previous part, show that the equation (3.1) can be written as

$$\beta u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1} = 0, \tag{3.2}$$

and determine the value of β.

d) Given that, in appropriate units,

$$\begin{cases} u(x,0) = u(x,1) = 10x & \text{for } 0 \le x \le 2, \\ u(0,y) = 0 & \text{for } 0 \le y \le 1, \\ u(2,y) = 20 & \text{for } 0 \le y \le 1, \end{cases}$$

write down the equation (3.2) for a discretization with $n = 5$ at the grid points (marked in Figure 3.1)

i) (x_5, y_2) ii) (x_9, y_1)

e) Information about the coefficient matrix A using a row-ordering of the variables $u_{i,j}$ is given in Figure 3.2.

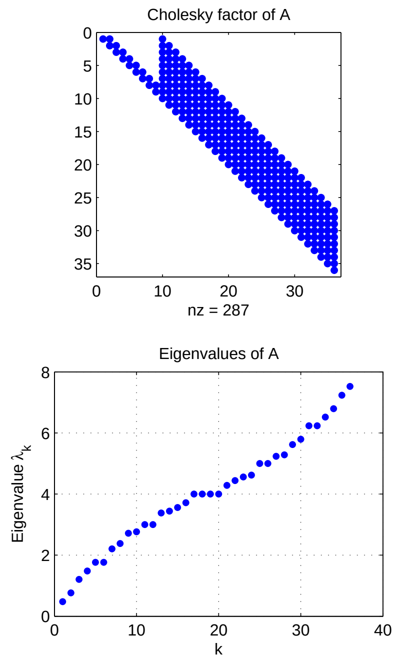

Figure 3.2: Spy plots of A, Cholesky factor R, inverse A^{-1}, and eigenvalues of A

i) Calculate the sparsity of A.

ii) Why is calculating A^{-1} not a good idea?

iii) From Figure 3.2, give **two** reasons why A is positive definite?

iv) Explain how to use the Cholesky factorization $A = R^{\mathrm{T}}R$ to solve the linear system $A\boldsymbol{u} = \boldsymbol{b}$?

v) What other structure does A have that could make solving a linear system $A\boldsymbol{u} = \boldsymbol{b}$ more efficient?