

Math 273 Midterm I

2001 October 4

Carry out the *solution* of each problem: show steps of any required calculations; state reasons that justify any conclusions. Mere oracular *answers* will receive no credit.

1. Give a MATLAB command that creates a 2×128 matrix `Dice` of random integers from one to six (inclusive), representing 128 throws of a pair of dice.

Given the array `Dice` described above, what does the following MATLAB fragment do?

```
eights = 0;
for i = 1:128
    if (Dice(1,i)+Dice(2,i) == 8)
        eights = eights + 1;
    end
end
```

Give a one-line MATLAB command that calculates the quantity `eights` without a `for` loop.

2. Using *only* the information in the following table, give three different ways to approximate the derivative $\Gamma'(1)$. Which approximation do you expect to be the most accurate?

x	$\Gamma(x)$
$\frac{1}{2}$	$\sqrt{\pi}$
1	1
$\frac{3}{2}$	$\frac{1}{2}\sqrt{\pi}$
2	1

3. Find the Newton form of the cubic polynomial that interpolates the data in the table below.

x	1	2	3	4
y	1	1	1	25

[Exam continues on Reverse]

4. Both function M-files shown here, downloaded from Van Loan's web site, compute the matrix product $y=A*x$. Although they obviously use different algorithms, the descriptive comments are identical! Edit the comments so that they correctly describe the algorithms used.

<pre> function y = MatVecR0(A,x) % y = MatVecR0(A,x) % Computes the matrix-vector % product y = A*x (via saxpys) % where A is an m-by-n matrix % and x is a column n-vector. [m,n] = size(A); y = zeros(m,1); for i=1:m y(i) = A(i,:)*x; end </pre>	<pre> function y = MatVecC0(A,x) % y = MatVecC0(A,x) % This computes the matrix-vector % product y = A*x (via saxpys) % where A is an m-by-n matrix % and x is a column n-vector. [m,n] = size(A); y = zeros(m,1); for j=1:n y = y + A(:,j)*x(j); end </pre>
--	---

5. On pages 25–26 Van Loan gives four ways to set up an $n \times m$ matrix A with $A(k,j) = \sin(jx_k)$. The script M-file `SumOfSines.m` in the middle of page 25 handles the case $(n,m) = (200,4)$ without a for loop. Why would that scheme be awkward to program if $m = 25$?

In §5.3 we learned that MATLAB's built-in functions accept matrix arguments. Give a one-line MATLAB command that computes A as the sine of an outer product matrix.

[Exam begins on Obverse]