AMATH 301 Final Exam

Name: Skyler Halliman

Student NetID: 1732227

Question	Points	Score
1	15	
2	10	
3	15	
4	18	
5	16	
6	16	
7	15	
8	18	
9	11	
10	16	
Total:	150	

Please write your name on every page you want us to grade.

Many of these questions ask you to write short segments of code. You should try your best to write functioning Matlab code. That means proper use of arithmetic symbols, parentheses, brackets, builtin function names, semicolons, etc.

0 0.01 0.02 0.03

Name: Skyler Hollinan
NetID: [illegible]

dt = 0.01 time = 0:dt:1

1. (15 points) Write Matlab code to solve the initial value problem $\dot{x} = x^2 + 1$ from $t = 0$ to $t = 1$ with a time step of $\Delta t = 0.01$ and the initial condition $x(0) = -1$. Use the forward Euler method. (You should write your own solver, not use a builtin function.) Your answer should be a row vector named x.

```
dxdt = @(x) (x^2 +1)
x0 = -1
dt = 0.01
x = zeros(1, 101)
x(1) = x0
 N = 100
 for k=1:N
x(k+1) = x(k) + dt * dxdt(k)
end
```

2. (10 points) Suppose that A is a 5×7 matrix. Write a single line of Matlab code using the svd command to find the reduced singular value decomposition of A. (You can assume that A is already defined.) What are the dimensions of the matrices returned by svd? Write another line of Matlab code to reconstruct A (exactly) using the resulting matrices.

[U,S,V] = svd(A, 'econ')

Dimensions: S is 5x5, U and V are 5x7

A = U*S*V'

1×2

2×1 $\quad -1 \; 2$ $\quad \begin{matrix} 1 & 2 \\ 2 & 3 \end{matrix}$

Name: Skyler Halliman
NetID: 1732227

3. (15 points) Let 3×1 vector $\quad 1\times 3$ vector

$$\mathbf{u} = \begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}, \; \mathbf{v} = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}, \; \mathbf{x} = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix} \text{ and } \mathbf{y} = \begin{pmatrix} 2 \\ -1 \\ 1 \end{pmatrix}.$$

Decide if each of the following expressions makes sense. If it does, calculate the result by hand and write a *single line* of Matlab code to compute the result. (You can assume that u, v, x and y have already been defined in Matlab.) If the result is a matrix, say what the rank of the matrix is.

(a) $\mathbf{u}^T\mathbf{v} + \mathbf{x}^T\mathbf{y}$

$(1\times 3)\times(3\times 1) + (1\times 3)\times(3\times 1)$ Adding inner products to give value but

$1\times 1 + 1\times 1 \rightarrow$ Works

$$\begin{bmatrix} -1 & 1 & -1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} + \begin{bmatrix} 2 & 1 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$$

$$\rightarrow \left[-1(1) + 1(0) - 1(2)\right] + \left[2(2) + -1(1) + 1(3)\right] \rightarrow [-3] + [6] \rightarrow [3]$$

$\rightarrow 3$

Matlab: u.'*v + x.'*y

(b) $\mathbf{u}\mathbf{v}^T + \mathbf{x}\mathbf{y}^T$

$(3\times 1)(1\times 3) + (3\times 1)(1\times 3)$

$(3\times 3) + (3\times 3) \rightarrow$ Makes sense. Rank 2 Matrix

$$\begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 2 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} \times \begin{bmatrix} 2 & -1 & 1 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} -1 & 0 & -2 \\ 1 & 0 & 2 \\ -1 & 0 & -2 \end{bmatrix} + \begin{bmatrix} 4 & -2 & 2 \\ 2 & -1 & 1 \\ 6 & -3 & 3 \end{bmatrix}$$

$$= \begin{bmatrix} 3 & -2 & 0 \\ 3 & -1 & 3 \\ 5 & -3 & 1 \end{bmatrix}$$

Matlab: u*v.' + x*y.'

Name: Skyler Hellman
NetID: 1732227

4. (18 points) The following code is meant to implement the section search to find the maximum of the function $f(x)$ given below. It is easy to check that the maximum is at $x = 1$. This code is written in two different files, called `section.m` and `fun.m`. These files are saved in the same directory. The code has several errors, which we will fix in turn.

$$f(x) = \begin{cases} -x^2 & \text{if } x < -2 \\ 2x & \text{if } -2 \le x \le 1 \\ 3 - x^3 & \text{if } x > 1. \end{cases}$$

```
FileName: section.m
1     clear all; close all; clc;
2
3     a = -4; b = 4;
4     c = 0.6;
5     tolerance = 1e-8;
6
7     for k = 1:100
8         x = c*a + (1 - c)*b;
9         y = (1 - c)*a + c*b;
10
11        if fun(x) < fun(y)
12            b = y;
13        else
14            a = x;
15        if (b - a) < tolerance
16            break
17        end
18    end
```

$x = 0.6 * a + (1-$

```
Filename: fun.m
1     function y = fun(x)
3     if x < -2
4         y = -x^2;
5     elseif (x >= -2) && (x <= 1)
6         y = 2x;
7     else
8         y = 3 - x^3;
9     end
10    end
```

The numbers on the left are just line numbers, not code. For each of the following parts (starting on the following page), clearly indicate any lines that you would need to change or add and what changes you would make. Be sure to indicate which file(s) need changes. Only fix the problem from the given part, not all problems that you see.

Name: Skyler Halliman
NetID: 1732227

(a) The code from the previous page does not work. When you run section.m, Matlab produces the following error message:

```
Error: File: section.m Line: 7 Column: 1
At least one END is missing: the statement may begin here.
```

What is causing this error, and what would you have to change in order to fix it?

section.m ↳ There is a missing end inside the for loop that starts at line 7 and ends at 18. There needs to be an end inserted at the end of the if else statement inside the for loop.

(b) After correcting the above error, your code still does not work. When you run section.m, Matlab produces the following error message:

```
Error: File: fun.m Line: 6 Column: 10
Unexpected MATLAB expression.

Error in section (line 11)
    if fun(x) < fun(y)
```

What is causing this error, and what would you have to change in order to fix it?

There is an error in the function file fun.m on line 6. There is an error because they have $y = 2x$ and they are missing the asterisk to indicate multiplication, should be $y = 2*x;$ instead.

(c) After correcting both of the above errors, your code appears to run successfully and stops after 41 steps. However, the final value of x is extremely close to −4, which is not the correct maximum value. What is causing this problem, and what would you have to change in order to fix it?

The initial bounds a and b have negative function evaluations when they are called. This means that using a and b would not allow for a proper convergence towards a minimum. Change initial a and b values. Or, instead

Name: Skyler Hallinan

NetID: 1732227

Root mean Square

5. (16 points) For the rest of this problem, assume that you have a file called data.mat in the same directory as the rest of your code. This file contains two vectors x and y of the same length. We will try find parameters a, b and c such that the curve $y = \cos(a * x) + \sin(b * x) + c$ best fits the data.

Sum of

(a) First, complete the following function that will compute the root-mean-square error between the curve $y = \cos(a * x) + \sin(b * x) + c$ and the data given in data.mat. Note that a, b and c are arguments in the function but x and y are not.

```
function E = rms_error(a, b, c)
import(data, mat)
y-calc = cos(a * x) + sin(b * x) + c;
E = sqrt(sum(1/((y-calc - y).^2)))
end
```

local minimum

(b) Now write code to calculate the coefficients a, b and c that minimize the root-mean-square error from part (a). Use an initial guess of $a = -1$, $b = 1$ and $c = 0$. You should name your results a, b and c. You can assume that the function from part (a) is saved as rms_error.m in the same directory as this code and the file data.mat.

```
guess = [-1, 1, 0];
wrap = @(v)(rms_error(v(1), v(2), v(3)))
[x, fval] = fminsearch(wrap, guess)
a = x(1);
b = x(2);
c = x(3);
```

9, 3, 4, 6

Name: Skyler Hallinan
NetID: 1732227

6. (16 points) Consider the second order initial value problem

$$\ddot{x} - 5(1-x^2)\dot{x} + x = 0,$$

$\ddot{x} = 5\dot{x}(1-x^2) + x$

where $x(0) = 1$ and $\dot{x}(0) = 0$. $\ddot{x} = 0$

(a) Rewrite this equation (by hand) as a system of first order equations. Be sure to rewrite the initial conditions as well. Show your work!

At $t=0$, $\ddot{x} - 5(1-1)0 + 1 = 0$

$\ddot{x} = -1$

$-5(1-x^2)\dot{x} + x = -1$

-1 $\ddot{x} = 5(1-x^2)\dot{x} - x$

$\ddot{x} = -5(0)\dot{x} + 1$

$\dot{x} = \frac{-1-x}{5(1-x^2)}$

(b) This equation is known to *not* be stiff. Write Matlab code using an appropriate builtin solver to solve this system of differential equations from $t = 0$ to $t = 5$. Find the value of x at time $t = 5$ and name your answer x5.

```
guess = [1 0]
V = @(x)([5*(1 - x(1)^2) x(1);(-1-x(1)) /(5*(1 - x(1)^2))])
T = 5;
[t_out, x_out] = ode45(V, [0 T], guess)
x5 = x_out(end)
```

$Ux = y$

$LU = PA$ $LU\ x = Pb$

$Ly = Pb$ ~~[illegible]~~

Name: [illegible]
NetID: 1732227

7. (15 points) Consider the system of equations $(A+I)\mathbf{x} = \mathbf{b}$, where

$$A = \frac{1}{(\Delta t)^2}\begin{pmatrix} -2 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 1 & -2 \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} 3/(\Delta t)^2 \\ 0 \\ 0 \\ 0 \\ -1/(\Delta t)^2 \end{pmatrix},$$

$\Delta t = 0.2$ and I is the 5×5 identity matrix.

Write Matlab code to solve this system using LU decomposition. You should explicitly keep track of the permutation matrix and you should not assume that any Matlab variables have already been defined. Your final answer should be named x.

```
A=(1/0.04).*[-2,1,0,0,0;1,-2,1,0,0;0,1,-2,1,0;0,0,1,-2,1;0,0,0,1,-2]
A= A+eyes(5);
[L,U,P] = lu(A);
b = [3/0.04;0;0;0;-1/0.04]
y= L\P*b;
x= U\y;
```

Name: Skyler Hollines
NetID: 1732227

8. (18 points) Your friend just told you that the following difference scheme is an approximation for $f'(x)$:

$$f'(x) \approx \frac{-f(x-2\Delta x) - 3f(x) + 4f(x+\Delta x)}{6\Delta x}$$

Confirm that this scheme really does approximate $f'(x)$ as Δx goes to zero. Find the leading error term and the order of accuracy of this method. (Write the order using big-oh notation.) Show your work!

Taylor series: $f(x) + \frac{\Delta x}{2!} f'(x) - \frac{\Delta x^2}{3!} f$

$$f'(x) \approx \frac{1}{6\Delta x}\left(\left(f(x) - \frac{2\Delta x}{2!} f'(x) + \frac{\Delta x^2}{3!} f\right) - 3f(x) + 4\left(f(x) + \frac{\Delta x}{2!} f'(x) - \frac{\Delta x^2}{3!} f\right)\right)$$

Error Leading Term: $f'(x) - \frac{5\Delta x^3}{6} f''(x)$

Order of Accuracy: $\Delta(O^3)$

Name: [illegible]
NetID: 1732227

9. (11 points) The Jacobi elliptic functions $sn(u,m)$, $cn(u,m)$ and $dn(u,m)$ are defined in terms of the integral

$$u = \int_0^{\phi} \frac{d\theta}{\sqrt{1 - m\sin^2\theta}}.$$

The Jacobi elliptic functions are

$$sn(u,m) = \sin(\phi(u,m)), \quad cn(u,m) = \cos(\phi(u,m)) \text{ and}$$
$$dn(u,m) = \sqrt{1 - m\sin^2(\phi(u,m))}.$$

Matlab has a builtin command called `ellipj` that efficiently evaluates these functions. You probably haven't seen this function before, but with access to the help file you should be able to use it in Matlab code. Below is a portion of the help file for `ellipj`.

```
ellipj
Jacobi elliptic functions

Syntax
[SN, CN, DN] = ellipj(U, M)
[SN, CN, DN] = ellipj(U, M, tol)

Description
[SN, CN, DN] = ellipj(U, M) returns the Jacobi elliptic functions sn, cn
and dn evaluated at the corresponding elements of arguments U and M.
Inputs U and M must be the same size, or either U or M must be scalar.

[SN, CN, DN] = ellipj(U, M, tol) computes the Jacobi elliptic functions
to accuracy tol.  The default value of tol is eps.  Increase tol for a less
accurate but more quickly computed answer.
```

Use the function `ellipj` to find $cn(0.75, 0.3)$ using the default tolerance. Name your final answer y.

```
[SN, CN, DN] = ellipj(0.75, 0.3)
    y = CN;
```

$\frac{f(x+\Delta x)-f(x-\Delta x)}{2\Delta x}$

Name: Skyler Ullman
NetID: 17 32227

10. (16 points) Suppose that x and y are two 100 × 1 vectors representing points on the function $y(x)$, where the entries of x are strictly increasing and evenly spaced dx apart. You can assume that dx is small. Throughout the problem, you can assume that x, y and dx are already defined, but you should define any other variables you need.

   (a) Write Matlab code to approximate the derivative y' at each point of x. Your answer should be in a vector named yprime. Use a central difference scheme whenever possible. You can use any order of accuracy.

```
yprime = zeros (100 , 1)
yprime(1) = (y(1) - y( 2 ))/dx
yprime(100) = (y(100) ) -y( 99 ))/dx;
  for  k = 2:98
    yprime (k)= (y (k+1) - y(k-1)) /(2 * dx)
  end
```

   (b) Write a *single line* of Matlab code to approximate the integral of $y(x)$ (over the entire x interval) using the trapz command. Save your answer as TRAP.

```
TRAP= trapz(x, y)

% Save ("TRAP.dat", "TRAP", "-ascii");
```

   (c) Write Matlab code to approximate the integral of $y(x)$ (over the entire x interval) using the right hand rule. Save you answer as LHR.

```
LHR = 0;
  for  k = 1:100
LHR=LHR+y(k)*dt;
  end

% LHR
```