**XJTLU Entrepreneur College (Taicang) Cover Sheet**

|  |  |  |
| --- | --- | --- |
| Module code and Title | **DTS104TC Numerical Methods** | |
| School Title | **School of Artificial Intelligence and Advanced Computing** | |
| Assignment Title | **Assignment 1** | |
| Submission Deadline | **June 10, 2021. 5pm (GMT+8)** | |
| Final Word Count | **-** | |
| If you agree to let the university use your work anonymously for teaching and learning purposes, please type **“yes”** here. | |  |

I certify that I have read and understood the University’s Policy for dealing with Plagiarism, Collusion and the Fabrication of Data (available on Learning Mall Online). With reference to this policy I certify that:

* My work does not contain any instances of plagiarism and/or collusion.  
  My work does not contain any fabricated data.

**By uploading my assignment onto Learning Mall Online, I formally declare that all of the above information is true to the best of my knowledge and belief.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Scoring – For Tutor Use** | | | | | | |
| **Student ID** | | | |  | | |
|  | | | | | | |
| **Stage of Marking** | | **Marker**  **Code** | **Learning Outcomes Achieved （F/P/M/D）**  **(please modify as appropriate)** | | | **Final**  **Score** |
| **A** | **B** | **C** |
| 1st Marker – red pen | |  |  |  |  |  |
| Moderation  – green pen | | **IM**  **Initials** | The original mark has been accepted by the moderator (please circle as appropriate): | | | Y / N |
|  | Data entry and score calculation have been checked by another tutor (please circle): | | | Y |
| 2nd Marker if needed – green pen | |  |  |  |  |  |
| **For Academic Office Use** | | | **Possible Academic Infringement (please tick as appropriate)** | | | |
| **Date**  **Received** | **Days late** | **Late Penalty** | **☐ Category A** | | Total Academic Infringement Penalty (A,B, C, D, E, Please modify where necessary) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  |  |  | **☐ Category B** | |
| **☐ Category C** | |
| **☐ Category D** | |
| **☐ Category E** | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Module** | **EXAMINER** | **DEPARTMENT** | **Email** |
| DTS104TC | Long Huang | School of Intelligent Manufacturing Ecosystem | Long.Huang@xjtlu.edu.cn |

**2nd SEMESTER 2020/21 Assignment**

**Undergraduate – Year 2**

**DTS104TC Numerical Methods**

**Submission Deadline: June 10, 2021. 5pm (GMT+8)**

**INSTRUCTIONS**

1. **The weighting of this assignment is 80% of the final mark.**
2. **The marking criteria sheet is provided as a supplementary document.**
3. **Your submission should only be in English.**
4. **Where required, Matlab code should be attached as .m files.** 
   1. **State the relevant .m file name in answer sheet to each question.**
   2. **It is allowed to make use of multiple .m files as functions and inputs.**
   3. **All relevant functions of all questions should be submitted in a single folder. The final answers to each question should be displayed in an executable .m file named after each question.**
5. **Answers to questions should be typed on separate A4 pages as Word files. The assignment must be submitted in Word via Learning Mall Online to the correct drop box. Only electronic submissions are accepted and no hard copy submissions are permitted.**
6. **All students must download their file and check that it is viewable after submission. Documents may become corrupted during the uploading process (e.g. due to slow internet connections). However, students themselves are responsible for submitting a functional and correct file for their assessments.**

|  |  |
| --- | --- |
| **Student Name** | **Student ID** |
|  |  |
| **DEPARTMENT** | **Email** |
|  |  |

***Question – 1 (20/100)***

Consider the following equation: Equation (1)

|  |  |
| --- | --- |
|  | (1) |

Given a value of y, Equation (1) can be used to solve for x using the bisection method.

1. Assume the initial guesses of x are set as 273.15 and 313.15. While solving for x based on the initial guesses and using the bisection method, find out how many iterations are required to determine the value of x to an absolute error of 0.01. (5 marks)

According to the question(a), the absolute magnitude of the error is,

and b-a, namely 313.15-273.15=40,

k11.97=12

Therefore, the iterations are 12

1. Implement and test the bisection method using Matlab. (5 marks)

Use the initial guesses in question 1(a) and include a check if the initial guesses bracket the roots in the Matlab code. (5 marks)

Solve x for the following cases: y=8; y=10; and y=13 for an absolute error of 0.01. (5 marks)

Submission requirements:

* Fill out Table-1 with your answers.
* The final answers should be executed and obtained in Matlab, using a file named AnswerOne.m.
* All relevant Matlab code should be copy & pasted in the section below.
* Attach your Matlab code as .m files in your submission.

Table - 1

|  |  |  |  |
| --- | --- | --- | --- |
| y | 8 | 10 | 13 |
| x (4 decimal places) | 299.9322 | 288.5357 | 277.4615 |

------------------------------------------------------------------------------------------

%start of matlab code for Question 1(b)

%The solution for question 1

%Initiation

close all;

clc;

clear all;

format long ... Scaled fixed point format with 15 digits.

%the function between different situations

f=@(x) -139.34411+1.575701\*10^5/x-6.642308\*10^7/(x^2)+1.243800\*10^10/(x^3)-8.621949\*10^11/(x^4)-log(8);

h=@(x) -139.34411+1.575701\*10^5/x-6.642308\*10^7/(x^2)+1.243800\*10^10/(x^3)-8.621949\*10^11/(x^4)-log(10);

g=@(x) -139.34411+1.575701\*10^5/x-6.642308\*10^7/(x^2)+1.243800\*10^10/(x^3)-8.621949\*10^11/(x^4)-log(13);

%the solution when y=8

fprintf('When y=8,')

bisection(f)

%the solution when y=10

fprintf('When y=10,')

bisection(h)

%the solution when y=13

fprintf('When y=13,')

bisection(g)

function bisection(f)

% initial guesses

a=273.15;

b=313.15;

% check if the initial guesses bracket the roots

if f(a)\*f(b)>0

disp('the initial guesses do not bracket the roots')

else

p = (a + b)/2; %the midpoint

err = abs(a-b); %the distance

while err > 0.01

if f(a)\*f(p)<0 %root lies in the lower interval,

b = p;

else

a = p; %root lies in the upper interval

end

p = (a + b)/2;

err = abs(a-b);

end

fprintf('the solution of the equation is ')

fprintf('%5.4f \n',p)

end

end

%end of matlab code for Question 1(b)

------------------------------------------------------------------------------------------

***Question – 2 (20/100)***

Implement and test the Gauss-Seidel method using Matlab. (10 marks)

Include a check in the Matlab code to determine if the coefficient matrix is diagonally dominant. (3 marks)

Compute the solutions for Equation-2 for a tolerance of 1e-3. (7 marks)

|  |  |
| --- | --- |
|  | (2) |

Submission requirements:

* Fill out Table-2 with your answers.
* The final answers should be executed and obtained in Matlab file named as AnswerTwo.m.
* All relevant Matlab code should be copy & pasted in the section below.
* Attach Matlab code as .m files in your submission.

Table-2

|  |  |
| --- | --- |
|  | Value (4 decimal places) |
| x1 | 0.9997 |
| x2 | 0.9998 |
| x3 | 1.0000 |
| x4 | 1.0001 |
| x5 | 1.0002 |
| x6 | 1.0001 |
| Number of Iterations | 9 |

------------------------------------------------------------------------------------------

%start of matlab code for Question 2

%The solution for question 2

%Initiation

close all;

clc;

clear all;

format long ... Scaled fixed point format with 15 digits.

A=[3 -1 0 0 0 1/2;

-1 3 -1 0 1/2 0;

0 -1 3 -1 0 0;

0 0 -1 3 -1 0;

0 1/2 0 -1 3 -1;

1/2 0 0 0 -1 3];

b=[5/2 3/2 1 1 3/2 5/2]';

%tolerance

tol=1e-3;

[M, N] = size(A);

%check: square matrix

if M~=N

error ('A is not a square matrix');

end

%check for diagnally dominant

for m= 1: M

row = abs(A(m,:));

d=sum(row)-row(m);

if row(m)<=d

error('A is not diagnally dominant');

end

end

%initial guess

x=zeros(M,1);

err=Inf;

itr=0;

while err>tol

x\_old=x;

for i=1:M

sigma=0;

for j=1:i-1

sigma=sigma+A(i,j)\*x(j);

end

for j=i+1:M

sigma=sigma+A(i,j)\*x\_old(j);

end

x(i)=(1/A(i,i))\*(b(i)-sigma);

end

itr=itr+1;

err=norm(x\_old-x);

end

fprintf('Solution of the system is :');

fprintf('\n%5.4f',x);

fprintf('\n');

disp(['total number of iterations= ' num2str(itr) ]);

%end of matlab code for Question 2

------------------------------------------------------------------------------------------

Question – 3 (20/100)

Implement and use the power method to determine the highest eigenvalue of Matrix A in Matlab.

1. Present four iterations of the calculation displaying the results with 4 decimal places. (15 Marks)
2. Determine if the solution is converging by comparing it with the eigenvalue obtained using eig() function. (5 Marks)

|  |  |
| --- | --- |
|  | (3) |

Submission requirements:

* Each iteration should be presented as typed matrices and equations.
* Fill Table-3 with your answers.
* The final answers should be executed and obtained in Matlab file named as AnswerThree.m.
* All relevant Matlab code should be copy & pasted in the section below.
* Attach your Matlab code as .m files in your submission.

Table-3

|  |  |
| --- | --- |
| Iteration 1 | |
| Estimated Eigenvector (normalized) | Estimated Eigenvalue |
| 0.5698 | 24.0000 |
| 0.4558 |
| 0.6838 |
| Iteration 2 | |
| Estimated Eigenvector (normalized) | Estimated Eigenvalue |
| 0.5658 | 20.6667 |
| 0.4548 |
| 0.6878 |
| Iteration 3 | |
| Estimated Eigenvector (normalized) | Estimated Eigenvalue |
| 0.5669 | 20.5323 |
| 0.4541 |
| 0.6873 |
| Iteration 4 | |
| Estimated Eigenvector (normalized) | Estimated Eigenvalue |
| 0.5665 | 20.5515 |
| 0.4543 |
| 0.6875 |

------------------------------------------------------------------------------------------

%start of matlab code for Question 3

%Initiation

close all;

clc;

clear all;

format long ... Scaled fixed point format with 15 digits.

A=[2 8 10;8 3 5;10 5 9]; %Write your matrix here.

u= [1;1;1]; % The initial choice of eigenvector.

n=length(u); % Size of initial eigenvector.

v=zeros(n,1);

% m1=0;m2=0;

itr=4;

for i=1:itr

v=A\*u;

m2=max(abs(v));

u=v/m2;

m1=m2;

fprintf('\n\nWhen the iteration %d:',i);

fprintf('\nThe greatest eigenvalue is %5.4f\n',m1);

disp('The corresponding eigenvector is:');

norm\_u = u/norm(u);

fprintf('\n%5.4f',norm\_u);

end

e=eig(A)

%end of matlab code for Question 3

------------------------------------------------------------------------------------------

***Question – 4 (20/100)***

Consider the model below to simulate the growth of y over time (t).

|  |  |
| --- | --- |
|  | (4) |

1. Implement Euler and 4th-Order Runge-Kutta Methods, conduct the simulation for t= [0, 10, 20, 30, 40, 50]. The initial condition of y0(t=0)=2555 is known. (15 Marks)
2. Compare the error based on the data presented in Table-4a. Plot the true errors (%) obtained with the two methods. (5 Marks)

Table-4a

|  |  |
| --- | --- |
| t | y-actual |
| 0 | 2555 |
| 10 | 3040 |
| 20 | 3710 |
| 30 | 4455 |
| 40 | 5275 |
| 50 | 6080 |

* Fill out Table-4b with your answers.
* Plot the errors in Matlab and paste them here in the answer sheet.
* Final answers should be executed and obtained in Matlab file named as AnswerFour.m.
* All relevant Matlab codes should be copy and pasted in the section below.
* Attach Matlab code as .m files in submission.

Table-4b

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data | | Euler | | 4th-Order Runge-Kutta | |
| t | y-actual | Estimation (4 significant figures) | True Error (%) with 2 decimal places | Estimation (4 significant figures) | True Error (%) with 2 decimal places |
| 0 | 2555 | 2555 | 0.00% | 2555 | 0.00% |
| 10 | 3040 | 3076 | 1.18% | 3114 | 2.43% |
| 20 | 3710 | 3668 | 1.13% | 3747 | 1.00% |
| 30 | 4455 | 4328 | 2.85% | 4445 | 0.22% |
| 40 | 5275 | 5045 | 4.36% | 5191 | 1.59% |
| 50 | 6080 | 5802 | 4.57% | 5963 | 1.92% |

------------------------------------------------------------------------------------------

%start of matlab code for Question 4

%Initiation

close all;

clc;

clear all;

format long ... Scaled fixed point format with 15 digits.

f=@(t,y) 0.0259\*(1-y/12000)\*y;

t0=0;

y0=2555;

tn=50;

h=10;

fprintf('The result produced by Euler method:');

while t0<=tn

fprintf('\n%d %5.0f ',t0,y0); %values of t and y

y1=y0+h\*f(t0,y0);

t1=t0+h;

t0=t1;

y0=y1;

end

fprintf('\nThe result produced by 4th-Order Runge-Kutta method:');

ty=0:h:50;

yy = zeros(1,length(ty));

yy(1) = 2555;

for i=1:(length(ty)) % calculation loop

k\_1 = f(ty(i),yy(i));

k\_2 = f(ty(i)+0.5\*h,yy(i)+0.5\*h\*k\_1);

k\_3 = f((ty(i)+0.5\*h),(yy(i)+0.5\*h\*k\_2));

k\_4 = f((ty(i)+h),(yy(i)+k\_3\*h));

yy(i+1) = yy(i) + (1/6)\*(k\_1+2\*k\_2+2\*k\_3+k\_4)\*h; % main equation

fprintf('\n%d %5.0f ',ty(i),yy(i));

end

%end of matlab code for Question 4

------------------------------------------------------------------------------------------

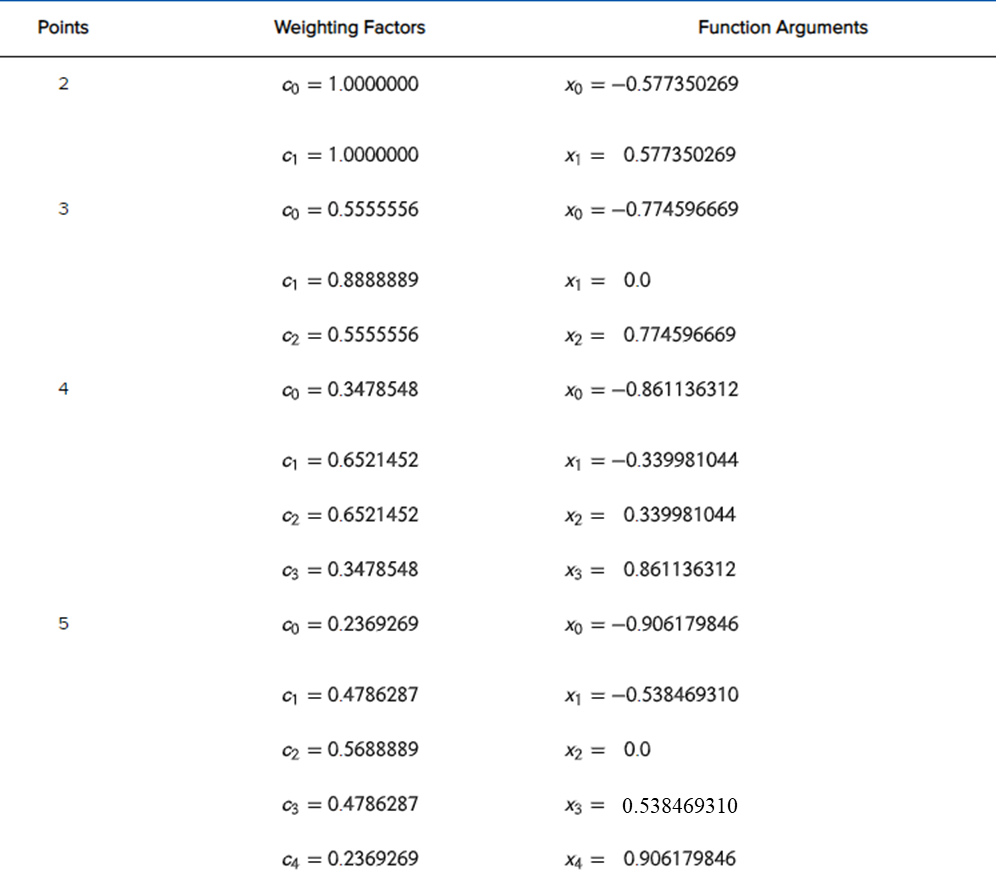
***Question – 5 (20/100)***

1. Table-5 provides the Gaussian Quadrature coefficients. In Matlab, calculate the approximate solution of Equation (5) using Gaussian Quadrature Method with n=3, n=4 and n=5; (10 marks)

|  |  |
| --- | --- |
|  | (5) |

1. Consider that the actual integral is 0.397610357. Compared to the accuracy attained using Gaussian Quadrature with n=5, how many segments are required by the Trapezoidal rule to attain a better accuracy? Test this in the Matlab environment. (10 marks)

Table-5



Submission Requirements:

* Fill out Table-5b with the answers.
* Final answers should be executed and obtained in Matlab file named as AnswerFive.m.
* All relevant Matlab codes should be copy pasted in the section below.
* Attach Matlab code as .m files in submission.

Table 5-b

|  |  |
| --- | --- |
| n | Approximation (9 decimal places) |
| 3 | 0.400020482 |
| 4 | 0.397441959 |
| 5 | 0.397613249 |

------------------------------------------------------------------------------------------

%start of matlab code for Question 5

%Initiation

close all;

clc;

clear all;

format long ... Scaled fixed point format with 15 digits.

a=1;

b=4;

% n=3

c0=0.5555556;

c1=0.8888889;

c2=0.5555556;

x0=-0.774596669;

x1=0.0;

x2=0.774596669;

f=@(x) exp(-0.5\*x^2);

z0=(b-a)/2.0\*x0+(b+a)/2.0;

z1=(b-a)/2.0\*x1+(b+a)/2.0;

z2=(b-a)/2.0\*x2+(b+a)/2.0;

I=(b-a)/2.0\*(c0\*f(z0)+c1\*f(z1)+c2\*f(z2));

fprintf('Integration of given function is=%5.9f, when n=3\n',I);

% n=4

c0=0.3478548;

c1=0.6521452;

c2=0.6521452;

c3=0.3478548;

x0=-0.861136312;

x1=-0.339981044;

x2=0.339981044;

x3=0.861136312;

f=@(x) exp(-0.5\*x^2);

z0=(b-a)/2.0\*x0+(b+a)/2.0;

z1=(b-a)/2.0\*x1+(b+a)/2.0;

z2=(b-a)/2.0\*x2+(b+a)/2.0;

z3=(b-a)/2.0\*x3+(b+a)/2.0;

I=(b-a)/2.0\*(c0\*f(z0)+c1\*f(z1)+c2\*f(z2)+c3\*f(z3));

fprintf('Integration of given function is=%5.9f, when n=4\n',I);

% n=5

c0=0.2369269;

c1=0.4786287;

c2=0.5688889;

c3=0.4786287;

c4=0.2369269;

x0=-0.906179846;

x1=-0.538469310;

x2=0.0;

x3=0.538469310;

x4=0.906179846;

f=@(x) exp(-0.5\*x^2);

z0=(b-a)/2.0\*x0+(b+a)/2.0;

z1=(b-a)/2.0\*x1+(b+a)/2.0;

z2=(b-a)/2.0\*x2+(b+a)/2.0;

z3=(b-a)/2.0\*x3+(b+a)/2.0;

z4=(b-a)/2.0\*x4+(b+a)/2.0;

I=(b-a)/2.0\*(c0\*f(z0)+c1\*f(z1)+c2\*f(z2)+c3\*f(z3)+c4\*f(z4));

fprintf('Integration of given function is=%5.9f, when n=5\n',I);

%end of matlab code for Question 5

------------------------------------------------------------------------------------------

END OF QUESTIONS

***This page is intentionally left blank.***

**LEARNING OUTCOMES**

This assessment tests your ability to:

A. Apply numerical methods in a number of different contexts.

B. Solve systems of linear and nonlinear algebraic equations to specified precision.

C. Compute eigenvalues and eigenvectors by the power method.

D. Solve boundary value and initial problems to finite precision.

E. Develop quadrature methods for numerical integration.

**MARKING CRITERIA**

The following table indicates what is expected for each classification category, highlighting generic marking criteria that bring together expectations in performance for each percentage (or alphabetical) band and the criteria that need to be satisfied.

**Generic Marking Criteria**

|  |  |  |  |
| --- | --- | --- | --- |
| **Grade** | **Point Scale** |  | **Criteria to be satisfied** |
| A | 81+ | First | * Outstanding work that is at the upper limit of performance. * Work would be worthy of dissemination under appropriate conditions. * Mastery of advanced methods and techniques at a level beyond that explicitly taught. * Ability to synthesise and employ in an original way ideas from across the subject. * In group work, there is evidence of an outstanding individual contribution. * Excellent presentation. * Outstanding command of critical analysis and judgment. |
| B | 70 - 80 | First | * Excellent range and depth of attainment of intended learning outcomes. * Mastery of a wide range of methods and techniques. * Evidence of study and originality clearly beyond the bounds of what has been taught. * In group work, there is evidence of an excellent individual contribution. * Excellent presentation. * Able to display a command of critical thinking, analysis and judgment. |
| C | 60 - 69 | Upper Second | * Attained all the intended learning outcomes for a module or assessment. * Able to use well a range of methods and techniques to come to conclusions. * Evidence of study, comprehension, and synthesis beyond the bounds of what has been explicitly taught. * Very good presentation of material. * Able to employ critical analysis and judgement. * Where group work is involved there is evidence of a productive individual contribution |
| D | 50- 59 | Lower Second | * Some limitations in attainment of learning objectives but has managed to grasp most of them. * Able to use most of the methods and techniques taught. * Evidence of study and comprehension of what has been taught * Adequate presentation of material. * Some grasp of issues and concepts underlying the techniques and material taught. * Where group work is involved there is evidence of a positive individual contribution. |
| E | 40 - 49 | Third | * Limited attainment of intended learning outcomes. * Able to use a proportion of the basic methods and techniques taught. * Evidence of study and comprehension of what has been taught, but grasp insecure. * Poorly presented. * Some grasp of the issues and concepts underlying the techniques and material taught, but weak and incomplete. |
| F | 0 - 39 | Fail | * Attainment of only a minority of the learning outcomes. * Able to demonstrate a clear but limited use of some of the basic methods and techniques taught. * Weak and incomplete grasp of what has been taught. * Deficient understanding of the issues and concepts underlying the techniques and material taught. * Attainment of nearly all the intended learning outcomes deficient. * Lack of ability to use at all or the right methods and techniques taught. * Inadequately and incoherently presented. * Wholly deficient grasp of what has been taught. * Lack of understanding of the issues and concepts underlying the techniques and material taught. * Incoherence in presentation of information that hinders understanding. |
| G | 0 | Fail | * No significant assessable material, absent, or assessment missing a “must pass” component. |

**Marking Criteria For Questions and Sub Question with Matlab Code on the basis of 100%.**

|  |  |  |
| --- | --- | --- |
| Area | Basis of marking | Marks(%) |
| Input | * Inputs are appropriately implemented   Mark   * Outstanding: 7 – 10 * Appropriate: 4 - 6 * Need improvement: 1 – 3 * No implementation or missing section: 0 | 10 |
| Correct implementation of each steps in the code | * Evidence of understanding of methods implementing   Mark   * Outstanding: 45 - 60 * Appropriate: 30 - 44 * Need improvement: 10 - 29 * No implementation or missing major sections: 0-9 | 60 |
| Post-process and output of results in supporting the answers | * Use of output methods and showing the answers with required significant digits, tables and figures.   Mark   * Outstanding: 8 - 10 * Appropriate: 5 - 7 * Need improvement: 1 - 4 * No display: 0 | 20 |
| Comments | * Use of appropriate comments in the code showing clear understanding of each step   Mark   * Outstanding: 7 – 10 * Appropriate: 4 - 6 * Need improvement: 1 – 3 * No implementation or missing section: 0 | 10 |
| Overall Mark (%) |  | 100 |