Nicholas Bush

Math 50

Assignment 5

Question 1

clear all

clc

fid = fopen('codemonkey.txt','r');

fid\_out = fopen('codemonkey\_out.txt','w');

j = 1;

while 1

tline = fgetl(fid);

if ~ischar(tline)

break

end

disp(tline)

fprintf(fid\_out, '%s\n', tline);

j = j + 1;

end

fclose(fid);

fclose(fid\_out);

Question 2

clear all

clc

fid\_out = fopen('survey.txt','w');

aline='Applied Math';

fprintf(fid\_out,'%s\n',aline);

bline = 'sophomore';

fprintf(fid\_out,'%s\n',bline);

cline = 7;

fprintf(fid\_out,'%u\n',cline);

dline = 7;

fprintf(fid\_out,'%u\n',dline);

fclose(fid\_out);

question 3

%i put all of the separate parts of this question into different files, error, fprimec, fprimel, fprimer

%fprimec means the center f prime

%fprimel means left f prime

%fprimer means right f prime

function [ fprimel ] = fprimel( f, a, b, n )

h = (b-a)/n;

fprimel = zeros(1,n);

x = linspace(a,b,n); %makes n points between a and b

for j=2:n

fprimel(j) = f(x(j)) - f(x(j-1));

end

fprimel = fprimel./h;

end

function [ fprimer ] = fprimer( f, a, b, n )

h = (b-a)/n;

fprimer=zeros(1,n);

x = linspace(a,b,n); %makes n points between a and b

for j=1:n-1

fprimer(j) = f(x(j+1)) - f(x(j));

end

fprimer = fprimer./h;

end

function [ fprimec ] = fprimec( f, a, b, n )

h = (b-a)/n;

fprimec = zeros(1,n);

x = linspace(a,b,n); %makes n points between a and b

for j = 2:n-1 %calculate for all but boundary values

% Finite Difference Scheme (centered)

fprimec(j) = f( x(j+1) ) - f( x(j-1) );

end

fprimec(1) = -3\*f( x(1) ) + 4\*f( x(2) ) - f( x(3) );

fprimec(n) = 3\*f( x(j) ) - 4\*f( x(j-1) )+ f( x(j-2) );

fprimec = fprimec ./ (2\*h);

end

function [ errs\_fprimer, errs\_fprimec, errs\_fprimel ] = error( f, a, b, n, set\_fprime )

% Nic

% 2-27-2014

% Math 50

% Lecturer: Derek

%

% INPUTS: function f, interval [a,b], number of grid points n known f'

% OUTPUT: errors

%

% will use this to compare left right and central estimates for deriv

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% initialization

h = (b-a)/n;

fprime = zeros(1,n);

errs\_fprimec = zeros(1,n);

errs\_fprimel = zeros(1,n);

errs\_fprimer = zeros(1,n);

for j=1:n

fprime(j) = set\_fprime(j\*h);

end

% error would be (predicted - actual)

errs\_fprimec =(fprimec(f,a,b,n)- fprime);

%this provides a detailed matrix of the errors of fprimec and the next two

%do the same thing for fprimel and fprimer

errs\_fprimel =(fprimel(f,a,b,n)- fprime);

errs\_fprimel(1)=0;%must reset this to zero as fprimel has no value here

errs\_fprimer= (fprimer(f,a,b,n)- fprime);

errs\_fprimer(n) = 0; %must reset this to zero as fprimer has no value here

errorc = norm(errs\_fprimec)%this prints the error that i will use to

%compare all three of these

errorl = norm (errs\_fprimel)

errorr = norm(errs\_fprimer)

% for some reason when using this on a sin function error r was smaller

% than the others and l was the largest, this is probaly because i used it

% between 0 and pi/3

%when i plugged in x.^2 and 2\*x both l and r were the same value and c was

%smaller than both of them

end