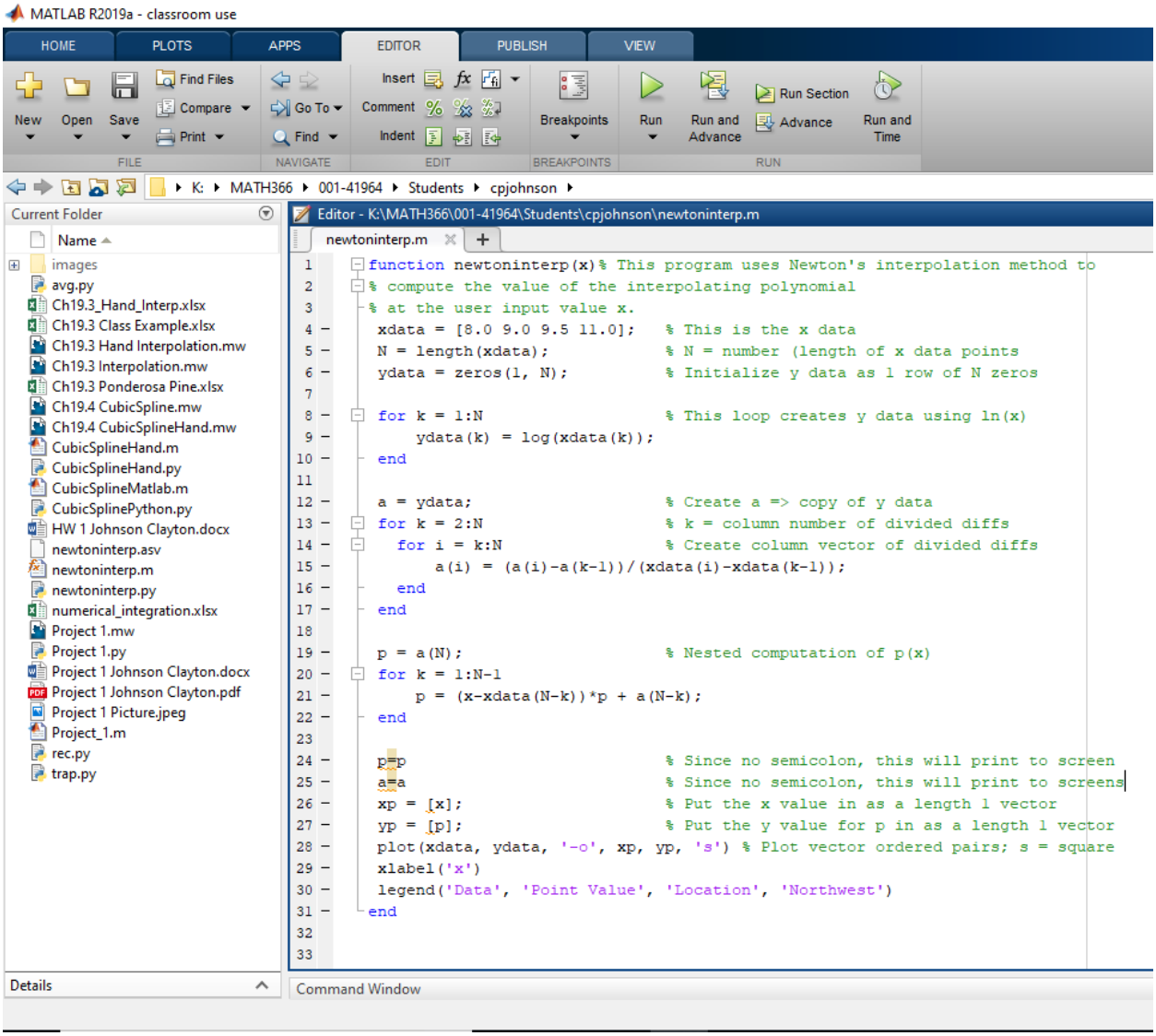
MATH 366 Methods of Applied Mathematics II

Clayton Johnson

HW 1: Ch19.3 Interpolation with Matlab

Overview



Program Code

Editor - K:\MATH366\001-41964\Students\cpjohnson\newtoninterp.m

newtoninterp.m

```
function newtoninterp(x)% This program uses Newton's interpolation method to
% compute the value of the interpolating polynomial
% at the user input value x.
 xdata = [8.0 9.0 9.5 11.0];    % This is the x data
 N = length(xdata);             % N = number (length of x data points
 ydata = zeros(1, N);           % Initialize y data as 1 row of N zeros

 for k = 1:N                    % This loop creates y data using ln(x)
     ydata(k) = log(xdata(k));
 end

 a = ydata;                     % Create a => copy of y data
 for k = 2:N                    % k = column number of divided diffs
   for i = k:N                  % Create column vector of divided diffs
       a(i) = (a(i)-a(k-1))/(xdata(i)-xdata(k-1));
   end
 end

 p = a(N);                      % Nested computation of p(x)
 for k = 1:N-1
     p = (x-xdata(N-k))*p + a(N-k);
 end

 p=p                            % Since no semicolon, this will print to screen
 a=a                            % Since no semicolon, this will print to screens
 xp = [x];                      % Put the x value in as a length 1 vector
 yp = [p];                      % Put the y value for p in as a length 1 vector
 plot(xdata, ydata, '-o', xp, yp, 's') % Plot vector ordered pairs; s = square
 xlabel('x')
 legend('Data', 'Point Value', 'Location', 'Northwest')
end
```

Command, Output and Figure

Editor - K:\MATH366\001-41964\Students\cpjohnson\newtoninterp.m

Command Window

```
>> newtoninterp(9.75)

p =

    2.2773


a =

    2.0794    0.1178   -0.0064    0.0004

fx >>
```