function [ ] = lamina\_failure( )

%lamina\_failure takes a string excelfilename and opens and reads the data

% in the excel file. Compares data using the Maximum Stress and Maximum

% Strain criteria to determine failure.

% get some info from the file being read

status = xlsfinfo('HW6.xlsx');

% Check the status variable to make sure the file is an Excel sheet

% If the status isn't the same as the string given

if strcmp(status,'Microsoft Excel Spreadsheet') == 0

% display an error message

disp('Error: File not an Excel sheet.')

% and return nothing

return

% If the string does match, continue the function

else

% Read data from the excel file

data = xlsread('HW6.xlsx');

% Extract individual components from the data matrix

stress\_xy = data(1:3,1);

strengths = data(:,2);

mat\_props = data(1:4,3);

theta = data(1,4);

E1 = mat\_props(1);

E2 = mat\_props(2);

G12 = mat\_props(3);

nu12 = mat\_props(4);

% Calculate nu\_21 from nu\_12

nu21 = nu12 \* E2 / E1;

% Calculate sin and cos

s = sind(theta);

c = cosd(theta);

% Transformation matrix

T = [ c^2 s^2 2\*s\*c ;...

s^2 c^2 -2\*s\*c ;...

-c\*s c\*s c^2-s^2 ];

% Calculate stress state

stress\_12 = T \* stress\_xy;

%% Maximum Stress criteria

fprintf('\nMaximum Stress Criteria:\n')

% Check the Stresses in each direction

% Check against positive and negative 1-direction

if stress\_12(1) < -strengths(2)

% Print a message and the relevant limits

fprintf(' Fails in 1-direction compression\n')

show = [ -strengths(2) stress\_12(1) strengths(1) ];

disp(show)

elseif stress\_12(1) > strengths(1)

% Print a message and the relevant limits

fprintf(' Fails in 1-direction tension\n')

show = [ -strengths(2) stress\_12(1) strengths(1) ];

disp(show)

end

% Check against positive and negative 2-direction

if stress\_12(2) < -strengths(4)

% Print a message and the relevant limits

fprintf(' Fails in 2-direction compression\n')

show = [ -strengths(4) stress\_12(2) strengths(3) ];

disp(show)

elseif stress\_12(2) > strengths(3)

% Print a message and the relevant limits

fprintf(' Fails in 2-direction tension\n')

show = [ -strengths(4) stress\_12(2) strengths(3) ];

disp(show)

end

% Check in 1-2-direction

if abs(stress\_12(3)) > strengths(5)

% Print a message and the relevant limits

fprintf(' Fails in shear\n')

show = [ -strengths(5) stress\_12(3) strengths(5) ];

disp(show)

end

% If it passes all the others, it must pass through this to be within

% the failure envelope

if stress\_12(1) >= -strengths(2) && stress\_12(1) <= strengths(1) &&...

stress\_12(2) >= -strengths(4) && stress\_12(2) <= strengths(3) &&...

abs(stress\_12(3)) <= strengths(5)

% Print a message and the relevant limits

fprintf(' Within failure envelope\n')

show = [ [ -strengths(2); -strengths(4); -strengths(5) ] stress\_12 [ strengths(1); strengths(3); strengths(5) ] ];

disp(show)

end

%% Maximum Strain criteria

% Calculate strains

strain\_12 = [ E1^-1 -nu21/E2 0; -nu12/E1 E2^-1 0; 0 0 G12^-1 ] \* stress\_12;

% Calculate max strains

elp = strengths(1)/E1;

eln = strengths(2)/E1;

etp = strengths(3)/E2;

etn = strengths(4)/E2;

elt = strengths(5)/G12;

% Check the Strains in each direction

fprintf('\nMaximum Strain Criteria:\n')

% Check against positive and negative 1-direction

if strain\_12(1) < -eln

% Print a message and the relevant limits

fprintf(' Fails in 1-direction compression\n')

show = [ -eln strain\_12(1) elp ];

disp(show)

elseif strain\_12(1) > elp

% Print a message and the relevant limits

fprintf(' Fails in 1-direction tension\n')

show = [ -eln strain\_12(1) elp ];

disp(show)

end

% Check against positive and negative 2-direction

if strain\_12(2) < -etn

% Print a message and the relevant limits

fprintf(' Fails in 2-direction compression\n')

show = [ -etn strain\_12(2) etp ];

disp(show)

elseif strain\_12(2) > etp

% Print a message and the relevant limits

fprintf(' Fails in 2-direction tension\n')

show = [ -etn strain\_12(2) etp ];

disp(show)

end

% Check in 1-2-direction

if abs(strain\_12(3)) > elt

% Print a message and the relevant limits

fprintf(' Fails in shear\n')

show = [ -elt strain\_12(3) elt ];

disp(show)

end

% If it passes all the others, it must pass through this to be within

% the failure envelope

if strain\_12(1) >= -eln && strain\_12(1) <= elp &&...

strain\_12(2) >= -etn && strain\_12(2) <= etp &&...

abs(strain\_12(3)) <= elt

% Print a message and the relevant limits

fprintf(' Within failure envelope\n')

show2 = [ [ -eln; -etn; -elt ] strain\_12 [ elp; etp; elt ] ];

disp(show2)

end

end