```
clear; close all;

% SFD and BMD

L = 1255; %length of board
n = 1200; %deiscretize by 1mm
P = 400; %total loading
u = 0; %uniform load
x = linspace(0, 1200, n+1); %x-axis

wheelLoc = [52 228 392 568 732 908]; %wheel location of train
wheelLoad = [-1 -1 -1 -1 -1 -1] * P/6; %load on each wheel

numStart = 241; % all 241 starting locations (0 - 240)
SFDi = zeros(numStart, n + 1);
BMDi = zeros(numStart, n + 1);

for i = 1:numStart
    reactionB = -(sum(wheelLoc .* wheelLoad) + 1200 * u * 600)/1200;
    reactionA = P - 1200 * u - reactionB;
    weight = zeros(1, n + 1); %weight function
    weight(1) = reactionA; % first element is reactionA
    weight(1201) = reactionB; % last element is reactionB
    for j = 1:length(wheelLoc)
        weight(wheelLoc(j)) = wheelLoad(j); % add applied load to weight
 function
    end
    SFDi(i, 1) = weight(1);
    for j = 2:n+1
        SFDi(i, j) = SFDi(i, j - 1) + weight(j) + u; % shear force of every
 increment = last SFD value + this weight value + UDL
    end
    for j = 2:n+1
        BMDi(i, j) = BMDi(i, j - 1) + SFDi(i, j); % bending moment of every
 increment = last BMD value + current SFD value
    end
    wheelLoc = wheelLoc + 240/(numStart - 1); % change starting location
end

SFD = max(abs(SFDi)); %SFD envelope
BMD = max(BMDi); %BMD envelope

% Material and Sectional Properties OG

E = 4000;
mu = 0.2;
stressTens = 30;
stressComp = 6;
shearMax = 4;
shearGlueMax = 2;
dimensions = [1.27 72.96 72.96 1.27 1.27 3.81; 65 1.27 1.27 6.27 6.27 86.4];
```

```
totHeight = dimensions(1,1) + dimensions(1,2) + dimensions(1,4) +
 dimensions(1,6);

ybarLocal = [dimensions(1,1)/2, dimensions(1,2)/2 + dimensions(1,1),
 dimensions(1,3)/2 + dimensions(1,1), dimensions(1,4)/2 + dimensions(1,2)
 + dimensions(1,1), dimensions(1,5)/2 + dimensions(1,3) + dimensions(1,1),
 dimensions(1,6)/2 + dimensions(1,4) + dimensions(1,2) + dimensions(1,1)];
ybarNumerator = dot(dimensions(1,:) .* dimensions(2,:), ybarLocal);
ybarDenominator = dot(dimensions(1,:), dimensions(2,:));
ybar = ybarNumerator/ybarDenominator;

I = sum(dimensions(1,:).^3 .* dimensions(2,:) / 12 + dimensions(1,:) .*
 dimensions(2,:) .* ((ybarLocal - ybar).^2));
Qmax = dimensions(1,1) * dimensions(2,1) * (ybar - ybarLocal(1,1)) + 2 * (ybar
 - dimensions(1,1)) * dimensions(2,2) * (ybar - ((ybar - dimensions(1,1))/2 +
 dimensions(1,1)));
Qglue = dimensions(1,6) * dimensions(2,6) * (totHeight - ybar -
 dimensions(1,6)/2);

stressTop = BMD(1, 400:400) * (totHeight - ybar) / I; %compression
stressBot = BMD(1, 400:400) * (ybar) / I; %tension
shearCent = SFD(1, 400:400) * Qmax / (I * (dimensions(2, 2) + dimensions(2,
 3))); %maxShear
shearGlue = SFD(1, 400:400) * Qglue / (I * (dimensions(2, 4) +
 dimensions(2,5))); %shearGlue00
stressTop = max(stressTop);
stressBot = max(stressBot);
shearCent = max(shearCent);
shearGlue = max(shearGlue);

% Thin plate buckling
flexBuck1 = (4 * pi^2 * E) / (12 * (1 - mu^2)) * (dimensions(1, 6) /
 (dimensions(2,1) - dimensions(2,2)/2 - dimensions(2,3)/2))^2;
flexBuck2 = (0.425 * pi^2 * E) / (12 * (1 - mu^2)) * (dimensions(1, 6) /
 ((dimensions(2,6) - dimensions(2,1))/2 + dimensions(2,2)/2))^2;
flexBuck3 = (6 * pi^2 * E) / (12 * (1 - mu^2)) * (dimensions(2, 2) /
 ((dimensions(1,1) + dimensions(1,2) + dimensions(1,4)/2) - ybar))^2;
shearBuck = (5 * pi^2 * E) / (12 * (1 - mu^2)) * ((dimensions(2,2)/
(dimensions(1,1) + dimensions(1,2)))^2 + (dimensions(2,2)/(420))^2);

%FOS
FOS_tens = stressTens/stressBot;
FOS_comp = stressComp/stressTop;
FOS_shear = shearMax/shearCent;
FOS_glue = shearGlueMax/shearGlue;
FOS_buck1 = flexBuck1/stressTop;
FOS_buck2 = flexBuck2/stressTop;
FOS_buck3 = flexBuck3/stressTop;
FOS_buckV = shearBuck/shearCent;
```

*Published with MATLAB® R2022b*