```
%EGM3601 stoplight project

clc, clear all, close all;

printf("EGM3601 Solid Mechanics stoplight Design Project\n")
printf("Name: Caleb Gibson\nNID: ca727627")
disp(date())
printf("Due date: Friday April 22, 2022\n")

%{
%declare and define constants
F_g = 164.59; %Newton [N]
rho_w = 76518; %Newton per cubic meter [N/m^3] material weight density
ro = 0.177; %meters [mm]
%R_y = 3*F_g + w*L; %Newton
E = 200e9; %Newtons per m^2 [N/m^2]
L = 14; %meters [m]
v = 0.1;  %meters [m] deflection
w = @(ri) rho_w*pi*(ro^2 - ri^2);
I = @(ri) (pi/2)*(ro^4 - ri^2);

%declare and define functions
v_tip = @(ri) (1/(E*I(ri)))*((-w(ri)*(14^4))/24+((14^3)*(-3*F_g+w(ri)*L))/6 ...
                                        +((14^2)*(89*F_g+98*w(ri)))/2) - 0.1;

%solve for given params
ri_guess1 = 0.17;
ri_guess2 = 0.18;
tol = 1e-5;
max_iters = 20;
ri_actual = secant(v_tip,ri_guess1,ri_guess2,max_iters,tol);
printf("The inner radius for the deflection to be 0.1m is %.6f.\n",ri_actual)

%}

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

%AFTER ATTENDING DR. YAVAS'S REVIEW SESSION: It has been discovered that the max
% deflection for a point load at ANY POINT along the length "L" is easily given
%by D_MAX = ((P*a^2)/(6*E*I)) * (3*L - a), where "a" is the length from the
%origin to the point of the application of the point load AND "P" is the POINT
%LOAD.
% The max deflection for a distributed load ALONG THE WHOLE LENGTH of a
%CANTILEVERED beam of length "L" and weight distribution "w" is given by:
%D_MAX = (w*(L^4))/(8*E*I). IN EACH CASE, "E" is the MODULUS OF ELASTICITY, "I"
%is the MOMENT OF INERTIA ABOUT THE Z OR Y AXIS OF THE CROSS-SECTIONAL AREA OF
%THE BEAM, and the MAX DEFLECTIONS CAN BE ADDED TOGETHER to get the TOTAL MAX
%DEFLECTION!!!
% THEREFORE,  (**side note: "UDL" is UNIFORM DISTRIBUTED LOAD**)
L = 14; %m
E = 207.5e9;  %N/m^2; according to https://matmatch.com/materials/minfm52552-astm-a595-grade-a-carbon-steel [1], average Modulus of Elasticity of Grade A Structural A595 carbon steel. This material was chosen because it has been consistently seen as being used by different Departments of Transportation (DOTs) as one of their materials of choice for the construction of traffic light pole arms.
P = 16.7829*9.8; %N; according to https://www.hillsboroughcounty.org/en/newsroom/2019/10/31/traffic-signals [2], average 3-light aluminum traffic signals weigh 37 lbs, which equates to 16.7829 "kilograms" according to Google's built in unit converter (these are actually Newtons, as when we measure the weight of something we are NOT measuring its MASS -- we are measuring its WEIGHT, which is a FORCE, not a MASS, and therefore we must measure it in terms of NEWTONS, NOT KILOGRAMS)
rho_w = 7850*9.8;  %N/m^3; according to [1], average density of A525 steel in kilograms per cubic meter, multiplied by 9.81 m/s^2 - the acceleration due to gravity - in order to convert it to Newtons per cubic meter.
%FROM PAGE 6 OF THE NYDOT TRAFFIC DESIGN MANUAL [3], IT IS KNOWN THAT THE DIA-
%-METER OF THE BASE OF THE MAST ARM IS 1.75 M FOR A STOPLIGHT OF THIS LENGTH (14
% m =~ 45.9318 feet), SO THE TOTAL DIAMETER OF THE POLE ARM can be taken as
%being HALF this base diameter, which is 1.75/2 m = *0.875 METERS.*
```

```
%[1]https://matmatch.com/materials/minfm52552-astm-a595-grade-a-carbon-steel
%[2] https://www.hillsboroughcounty.org/en/newsroom/2019/10/31/traffic-signals
%[3] https://www.dot.ny.gov/portal/pls/portal/mexis_app.pa_ei_eb_admin_app.show_pdf?id=13512

d_O = 0.875;  %m
rO = d_O/2; %m



%SINCE we are calculating for the geometry such that the MAX DEFLECTION <= 100
%MM, ALSO KNOWN AS 0.1 M, CAN MAKE the WHOLE DEFLECTION FUNCTION a FUNCTION of
%INNER RADIUS "ri" and REMOVE THE EI FROM THE DENOMINATOR TO SIMPLIFY THE
%EQUATION. (**side note: "rO" = outer radius)
% D_MAX_TOTAL = 0.1 = D_MAX_UDL + D_MAX_P1 + D_MAX_P2 + D_MAX_P3, THEREFORE,
% 0 = D_MAX_UDL + D_MAX_P1 + D_MAX_P2 + D_MAX_P3 - 0.1, THEREFORE
%ri_calculatn = 0 = 3wL^4 + 4PL1^2(3L-L1) + 4PL2^2(3L-L2) + 4PL3^2(3L-L3) -
% 0.1EI , WHERE "I" = (PI/4)*(rO^4 - ri^4) and "w" = rho_wpi(rO^2-ri^2)
%**SPECIAL NOTE TO SELF: here, "I" refers to the second PLANAR moment of area,
%NOT the second polar moment of area (also known as the PLANAR MOMENT OF
%INERTIA, not the polar moment of inertia). This can be confusing, since dif-
%-ferent professions and sub-specializations use "moment of inertia" and "second
% moment of area" to refer to different moments. However, THE MOMENT OF INERTIA
%FOR CALCULATING DEFLECTION IS THE *PLANAR* MOMENT OF INERTIA, NOT the polar
%moment of inertia. [2]

%[2] https://www.engineeringtoolbox.com/area-moment-inertia-d_1328.html#:~:text=Area%20Moment%20of%20Inertia%20or,bending%20and%20stress%20in%20beams.

%I_y = @(ri) (pi/4)*(rO^4-ri^4);

w = @(ri) rho_w*pi*(rO^2-ri^2);
ri_calculatn = @(ri) ((3*w(ri)*L^4) + ((4*P*(5^2))*(3*L-5)) + ((4*P*L*(9^2))*(3*P-9)) + ((4*P*L*(13^2))*(3*P-13))) - (24*0.1*E*(pi/4)*(rO^4 - ri^4));

%ri_calculatn = @(ri) (w(ri)*L^4)/(8*E*I_y(ri)) + ((4*P*(5^2))*(3*L-5))/(6*E*I_y(ri)) + ((4*P*L*(9^2))*(3*P-9))/(6*E*I_y(ri)) + ((4*P*L*(13^2))*(3*P-13))/(6*E*I_y(ri)) - 0.1;

%(**special note: L1=5m, L2=9m, L3=13m**) **"24" is LCD of equatn.
%NOW, HAVING the EQUATION WE NEED, we CAN USE OUR "secant" SOLVER FUNCTION TO
%SOLVE FOR THE ROOTS OF THE "D_MAX_TOTAL" EQUATION SUCH THAT THE MAX DEFLECTION
%IS <=100MM ALSO KNOWN AS 0.1M.

tol = 1e-3;
max_iters = 1000;
ri_guess=0.4;  %m
ri_guess2=0.3;  %m
ri_actual = secant(ri_calculatn,ri_guess,ri_guess2,max_iters,tol);

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

%NOW, we can CALCULATE *STEPS 4 AND 5 AND 6* of the project. Using what we know
%now, we can calculate the force acting along on the beam due to a wind gust
%with a velocity of 50 m/s. The total DRAG FORCE "D" due to the wind gust is the dynamic pressure
% dynamic pressure due to the air being forced against it, given by
%p = CD*(0.5*rho_air*v_air^2), where "CD" is the drag coefficient of the shape
%(in our case, a cylinder), "rho_air" is air's density at standard temperature
%and pressure (STP) which is 1.225 kg/m^3, using the International Standard
%Atmosphere (ISA) values of temperature = 15 degrees Celsius and pressure
%= 1 atm = 101325 Pa at sea level (according to macinstruments.com [^1]), all
%that TIMES the PLATFORM AKA PROJECTED AREA of the SURFACE IT IS ACTING UPON
%(in our case, the HORIZONTAL of the stoplight arm, MINUS THE END [so
%OUTER DIAMETER TIMES TOTAL LENGTH, or "D_OUTER" times "L"]).

%[^3] https://macinstruments.com/blog/what-is-the-density-of-air-at-stp/

%*Note to self: the reason we only use "outer diameter" instead of "pi*diamtr/2"
% is because the "projected area" is not the area if we were to unwrap the
```

```
%surface the dynamic fluid is blowing against, but rather simply our view of it
%from the singular direction (in this case) that the dynamic fluid is flowing
%from, which in this case would be the same no matter which direction it came at
% the cylinder at, as long as it is still normal to points on the wall - not the
% bases - of the cylinder. Henceforth, OUTER DIAMETER WILL BE REFERRED TO AS
% "d_O".
%"CD" ~= 0.5 for a cykinder.
%THEREFORE, "D" = 0.5*(0.5*1.225*50^2)*(dO*L) = 0.25*1.225*2500*(0.875*14). The
%DISTRIBUTED FORCE DUE TO THE WIND, "w_DRAG", is this drag force "D" per unit
%length, in essence "w_DRAG" = "D" divided by the length "L", in another word,
%"w_DRAG" = D/L = (0.5*(0.5*1.225*50^2)*(dO*L))/L, so = 0.5*(0.5*1.225*50^2)*dO,
% and since dO = 2rO = 2*0.127 = 0.254 [m],"w_DRAG" = 0.25*1.225*2500*0.254

w_DRAG = 0.25*1.225*2500*d_O; %(kg/m^3)(m^2/s^2)(m)=[kg/s^2],since [N]=[kgm/s^2], then [kg/s^2]=[(kgm/s^2)/m]=[(N)/m]
printf("==================================================================\n")
printf("NOW APPLYING A WIND GUST OF 50 M/S PARALLEL TO THE Y-AXIS, NORMAL \n\
TO THE Z AND X AXES ACTING ALONG THE ENTIRETY OF THE BEAM:\n")
printf("The total drag force acting on the beam is %.4f N.\n",w_DRAG*L)
printf("The drag force per unit length acting on the beam is %.4f N/m.\n",w_DRAG)
printf("The dynamic pressure due to drag is %.4f Pa.\n",(w_DRAG/d_O))

%the total moment acting in the "y" direction, normal to both the mast arm as
%well as the vertical upright pole it is attached to (acts as the "z" axis), is
%equivalent to the magnitude force times the magnitude (in other words the
%length) of the moment which extends out to the point upon which it acts, which,
% since it is a distributed force, can be modeled as having its total force -
%which is "D" - as acting on the centroid of its (w_DRAG's) application to the
%beam. THEREFORE, half the length of its (the UDL's) application distance, which
% since it acts across the whole length of the beam is L over 2 {L/2}.
%=> D(L/2). MOMENT ALONG THE *X* AXIS is given by the weight of the beam acting
% on its centroid (the distance to which is the moment arm), plus each point
% load times its respective moment arm.

M_z = -(w_DRAG*L)*L/2; % <- this is actually M in y, but the PowerPoint has it notated as M in z, so to be consistent I have notated it as M in z as well; but according to our reference system, it is actually M in y
M_y = -(w(ri_actual)*L/2 + P*27);  %since M_y = w*L/2 + P*5 + P*9 + P*13, the "P" can just be factored out of the latter 3/4ths of the equation, to reduce math operations (MOPS), thereby reducing overall computational complexity (albeit only minimally)

%it is worth noting that although traditionally, the sign of every moment in the M_x equation would be positive, since each one causes the arm to rotate counterclockwise, here we take clockwise to be our positive convention, since we denote to the left of "x" as positive (in another word, LEFT OF ORIGIN is our POSITIVE direction along the x axis by our convention, so we must use clockwise along x as our positive moment convention along x; our upwards and outwards [z and y] conventions are the same as traditional however, so moment need not be converted to negative counterclockwise positive clockwise for these 2 cardinal directions)
%now to find the combined stress-state at the critical cross-section of the
%pole, i.e. where the moment is maximized, which is at the base of the beam
%(just before the reaction forces act where it attaches to the rigid vertical light pole). Therefore, calculate the
%bending moment at different points (which are essentially infinitesimally small
%areas on the cross-section) of the cross-section. In order to calculate the max
% stresses experienced at these points due to the bending moments (which are dif
%-ferent along the y and z axes, since the bending moment in z is only due to the wind force blowing against the mast arm whereas the benfding moment in x is due to the weight of the mast arm acting along its centrodi of its applicatioin as well as the weights of each stoplight acting at a moment arm of the length from the base at which they act).
%Therefore, the bending stresses and shear stresses at the outermost points along the wall (of the mast arm) must be calculated, and each one respectively must be treated as most critical or not so by comparing its stresses with the others and then seeing which one is the mallest (since this is the most critical stress, since it will be able to resist only this amount of stress, which is smaller than all the other stresses). Then we have to calculate which stress is the greatest and whether or not it will break the material (because it exceeds the ultimate tensile strength of the material, which in our case is Structural A992 steel, which has an yield strength of 345 MPa and an ultimate strength of 450 MPa.
%equation for max normal bending stress in y:bsy=-(M_z*c)/Iz; in z:bsz=-(M_y*c)/Iz
```

```
%equation for max shear stress in y:tauy=(V*Qy)/(Iy*t); in z:tauz=(V*Qz)/(Iz*t)
%Iz = (pi/2)*(rO^4-ri^4) = Iy = I; Qz = Qy = A*distance to centroid normal to axis; the geometric center (centroid) for a half-circle is (pi*(1/3)*(rO^3-rO^2)/(pi*(rO^2-ri^2)/2)=(2/3)*((rO^3-ri^3)/(rO^2-ri^2)), therefore since Q_max = Q for the whole half-circle area, then the centroid will be:
%half_circ_centroid = (2/3)*((rO^3-ri_actual^3)/(rO^2-ri_actual^2));
%And therefore, Q will be this times the area of the half circle, (pi/2)*(rO^2-ri^2), => Q = (pi/3)*(rO^3-ri_actual^3)

Q = (2/3)*(rO^3-ri_actual^3);
%shear force in z direction is wL+3P; shear force in y direction is total drag force; thickness "t" is rO-ri; so t = rO-ri_actual:

t = 2*(rO-ri_actual);

%Therefore, we have

I = (pi/4)*(rO^4-ri_actual^4);
bsy = (M_z*rO)/I;
bsz = (M_y*rO)/I;
total_flexure = abs(bsy) + abs(bsz);
tauy = ((w(ri_actual)*L+3*P)*Q)/(I*t);
tauz = ((w_DRAG*L)*Q)/(I*t);
printf("==================\n")
printf("The maximum bending stress in the y direction of the cross-sectional \
area is %.8f Pa.\n",bsy)
printf("The maximum bending stress in the z direction of the cross-sectional \
area is %.8f Pa.\n",bsz)
printf("The magnitude of the total flexure of the cross-sectional area is %.8f \
 Pa.\n",total_flexure)
printf("The maximum shear stress in the y direction of the cross-sectional area\
 is %.8f Pa.\n",tauy)
printf("The maximum shear stress in the z direction of the cross-sectional area\
 is %.8f Pa.\n",tauz)

stresses = [bsy;bsz;total_flexure;tauy;tauz];
max_stress=0;

for i = 1:length(stresses)
  if abs(stresses(i)) > abs(max_stress)
    max_stress=stresses(i);
  else
    continue
  endif
endfor
min_stress = max_stress;
for j=1:length(stresses)
  if abs(stresses(j)) < abs(min_stress)
    min_stress = stresses(j);
  else
    continue
  endif
endfor

printf("\nThe MOST CRITICAL STRESS is %.8f Pa.\n",max_stress)

%printf("The most critical of these stresses is %s.\n",min_stress)

%Now, it must be determined whether this most critical stress exceeds the ultimate tnesile strength of the material selected, Structural A992 steel.

uts_A595_grade_a = 450e6; %N/m^2 aka Pa
ys_A595_grade_a = 380e6;  %N/m^2 aka Pa
if abs(uts_A595_grade_a) > abs(max_stress)
  printf("The material will NOT FAIL due to the stress.\n")
  failure = 0;
else
  printf("FAILURE!! The material will fail at this stress load.\n")
  failure = 1;
endif
```

```
if failure == 0
  FS = abs(uts_A595_grade_a/max_stress); % Factor of Safety, equal to the 'Ultimate Tensile Strength' divided by the 'Critical stress'
  printf("\nThe Factor of Safety (F.S.) of the material is %.6f.\n", FS)
else
  printf("The factor of safety is less than 1, therefore either the pole needs to be re-designed or the selected material needs to be stronger.\n")
endif
```