Table of Contents

 1
Basic Level 1
Medium Level 2
Advanced 3


```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Ian Landwehr, Sam Alvares, and Sam Ridgley
% March 30rd, 2021
%
% ME480: Machine Component Design
% Dr. Constans
% Spring 2021
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Basic Level

```
clc
close all
clear variables

% Geometry
Aloc = [1 -24 36];
COMloc = [38 -36 36];
MomentArm = COMloc-Aloc;

% Reaction At A
ExtForces = [0 30 0]; % Shear Force on A
ExtMoment = cross(MomentArm, ExtForces);

IntForces = -1*ExtForces;
IntMoment = -1*ExtMoment;


% Beam Shear due to Fy
T = 1/4 ; %     The cross sectional wall thickness area.
Do = 1 ;%       The outer Diameter of the lower pole
Di = 1-1/4;%    The inner Diameter of the lower pole
alpha = 90; %   Degrees (half-circle)
Q = 2*sind(alpha)/3*((Do/2)^3-(Di/2)^3);
I = pi/64*(Do^4-Di^4);
V = IntForces(2);
Tau = V * Q /I/T;

% Bending Moment due to Mz
Mz = IntMoment(3);
sigma = IntMoment * Do / 2 / I ;
```

Medium Level

```
clc
close all
clear variables

% Geometry
Bloc = [0 -18 1];
COMloc = [38 -36 36];
MomentArm = COMloc-Bloc;

% Reaction At B
ExtForces = [0 30 0]; % Shear Force on B
ExtMoment = cross(MomentArm, ExtForces);
IntForces = -1*ExtForces;
IntMoment = -1*ExtMoment;


% Beam Shear due to Fy
Do = 2 ;%       The outer Diameter of the lower pole
Di = 1.5;%    The inner Diameter of the lower pole
alpha = 90; %   Degrees (half-circle)
Q = 2*sind(alpha)/3*((Do/2)^3-(Di/2)^3);
I = pi/64*(Do^4-Di^4);
T = 0.5;
V = IntForces(2);
TauBeam = V * Q /I/T;

% Torsion due to Mz
T = IntMoment(3);
J = pi/32*(Do^4-Di^4);
TauTorsion = abs((T*Do)/(2*J));

% Bending Moment due to Mx
Mx = IntMoment(1);
sigmaMx = Mx * Do / 2 / I ;

% Enter stress matricies
sigma1 = [         0, 0,TauTorsion;
                   0, 0,         0;
          TauTorsion, 0,   sigmaMx];

sigma2 = [0,                     0,                      0;
          0,                     0, -(TauTorsion+TauBeam);
          0, -(TauTorsion+TauBeam),                      0];

sigma3 = [         0, 0,-TauTorsion;
                   0, 0,          0;
          -TauTorsion, 0,  -sigmaMx];

sigma4 = [0,                     0,                     0;
          0,                     0, (TauTorsion+TauBeam);
          0,  (TauTorsion+TauBeam),                     0];
```

```
% Find Principle stresses
[V1,D1] = eigs(sigma1);
[V2,D2] = eigs(sigma2);
[V3,D3] = eigs(sigma3);
[V4,D4] = eigs(sigma4);
```

Advanced

```
clc
clear all
close all
clc
close all
clear variables

% Geometry
Bloc = [0 -18 1];
COMloc = [38 -36 36];
MomentArm = COMloc-Bloc;

% Find Reactions at A
ForceVector = 15*[2 1 3]*1/3.7417;
ExtForces = [0 30 0]-ForceVector;
ExtMoment = cross(MomentArm, ExtForces);
%Internal Reactions
IntForces = -1*ExtForces;
IntMoment = -1*ExtMoment;


% Beam Shear due to Fy
Do = 2 ;%       The outer Diameter of the lower pole
Di = 1.5;%    The inner Diameter of the lower pole
A = pi*(Do^2-Di^2)/4;
alpha = 90; %   Degrees (half-circle)
Q = 2*sind(alpha)/3*((Do/2)^3-(Di/2)^3);
I = pi/64*(Do^4-Di^4);
T = 0.5;
V = IntForces(2);
TauBeamY = abs(V * Q /I/T);
TauBeamX = abs(IntForces(1)* Q /I/T);
TauBeamVec = [TauBeamX TauBeamY];
TauBeamMag = sqrt(TauBeamVec(1)^2+TauBeamVec(2)^2);
%TauBeamAngle =


%Tension Stress
sigmaFz = abs(IntForces(3)/A);
% Torsion due to Mz
T = abs(ExtMoment(3));
J = pi/32*(Do^4-Di^4);
TauTorsion = (T*Do)/(2*J);
```

```
% Bending Moment due to Mx
Mx = abs(IntMoment(1));
sigmaMx = Mx * Do / 2 / I ;
% Bending Moment due to My
My = abs(IntMoment(2));
sigmaMy = My * Do / 2 / I ;

sigma1 = [          0,             0,           (TauTorsion+TauBeamX);
                    0,             0,                               0;
          (TauTorsion+TauBeamX), 0,   (sigmaFz+sigmaMx)];

sigma2 = [0,                     0,                          0;
          0,                     0, -(TauTorsion+TauBeamY);
          0, -(TauTorsion+TauBeamY),    (sigmaFz+sigmaMy)];

sigma3 = [          0, 0,(TauBeamX-TauTorsion);
                    0, 0,                    0;
          (TauBeamX-TauTorsion), 0,  (sigmaFz-sigmaMx)];

sigma4 = [0,                     0,                          0;
          0,                     0, (TauTorsion-TauBeamY);
          0,  (TauTorsion-TauBeamY),   (sigmaFz-sigmaMy)];
% Find Principle stresses
[V1,D1] = eigs(sigma1);
[V2,D2] = eigs(sigma2);
[V3,D3] = eigs(sigma3);
[V4,D4] = eigs(sigma4);

% Find point of max principle stress
thetaR = atand(IntMoment(2)/IntMoment(1));
theta5 = thetaR+90;
MR = sqrt(IntMoment(2)^2+IntMoment(1)^2);
sigma5 = [ 0 0 TauTorsion;
           0 0 0;
           TauTorsion 0 (sigmaFz+sqrt(sigmaMy^2+sigmaMx^2))];
[V5,D5] = eigs(sigma5);
```

Published with MATLAB® R2020a