$w = \rho(\pi d^2/4)(\pi D) N_e$

ME 360 – P02 Name _____

USE THIS SHEET AS THE FIRST PAGE OF YOUR PROJECT SUBMISSION

3 inequality constraints

1) Use MATLAB "fminsearch" to design a torsion spring based on the following specifications.
Attach hard copy of your code and MATLAB output.
minimum weight
moment for θ = 90° must be greater than 8 in.lbf
must operate over 0.400 inch DIA shaft at θ = 90°
factor of safety at θ = 90° must be greater than 1.0 to prevent yield
coil direction = LH, number of coils in body of spring N_b = 4.25, leg length = 2.0 inch
round steel music wire
starting design – d = 0.090 in, OD = 0.800 in, w = 0.024 lbf
include Norton equations 14.27 and 14.30 in your analysis

$M_{WORK} = k_\theta (\frac{1}{4} turn)$

$ID = D - d$

$\tau < S_{yB}$

find d and D

2) Select a spring that most closely matches your optimal design from McMaster-Carr.

	Optimal design	**McMaster-Carr**
wire diameter d [in]	_____	_____
coil OD [in]	_____	_____
number of coils in body N_b	4.25	_____
leg length [in]	2.0	_____
weight [lbf]	_____	_____
coil ID unloaded [in]	_____	_____
spring rate k_T [in.lbf/rad]	_____	_____
moment at θ = 90° [in.lbf]	_____	_____
coil ID at θ = 90° [in]	_____	_____
N_{FS} at θ = 90°	_____	_____
McMaster-Carr part number		_____
cost each		_____

Extra credit – Explore the sensitivity of your design to number of coils and leg length.

ME 360 – P02 Name _____

```
% t_fminsearch.m - test fminsearch
% HJSIII, 14.10.16

% initial guess
x_start = [ 1 1 ]';

% call
[ x_solution, min_val ] = fminsearch( 'biquad', x_start )

% bottom - t_fminsearch
```

d_start = 0.090;
D_start = 0.800;
x_start = [d_start ; D_start];

('my_spring')

++

w my_spring

```
function z = biquad( x )
% biquadratic test function for fminsearch
% HJSIII, 14.10.16

% minimum = 3 at x(1)=2 and x(2)=5
z = ( x(1)^2 - 4*x(1) + 4 ) + ( x(2)^2 - 10*x(2) + 25 ) + 3;

% penalty function to provide inequality constraint
% constrained minimum = 3.8 at x(1)=2.4 and x(2)=4.2
%t = 0.5 * x(1) + 3;
%if x(2) > t,
%   z = z + 100;
%end

% bottom - biquad
```

return

d = x(1);
D = x(2);
kth =
Mwork =
ID = D - d;
Sut =
Syb =
tau =
w =

++

```
» t_fminsearch

x_solution =

    2.0000
    5.0000

min_val =

    3.0000
```

if Mwork < 8,
  w = w + 10;
end
if ID < 0.4,
  w = w + 20;
end

if tau > Syb,
  w = w + 40;
end
return