
Table of Contents

.....	1
Solution - Problem 1 (do not modify this line)	1
Solution - Problem 2 (do not modify this line)	1
Solution - Problem 3 (do not modify this line)	2
Solution - Problem 4 (do not modify this line)	3
Solution - Problem 5 (do not modify this line)	3

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
MATLAB HW 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% - whenever you write code, it is a good practice to
%   add comments that describe the code.
% - if you use 'two % marks' (%%), we can separate sections
%   in a script file.
% - good luck!
```

Solution - Problem 1 (do not modify this line)

```
rho = 700;
v = 5; % your turn
diam_ratio = [0.9, 0.8, 0.7, 0.6, 0.5, 0.2];

deltaP = 0.5*(1-diam_ratio.^2).^2*rho*v^2 % your turn

deltaP =

    1.0e+03 *

Columns 1 through 3

    0.3158750000000000    1.1339999999999999    2.2758750000000000

Columns 4 through 6

    3.5840000000000001    4.9218750000000000    8.0640000000000000
```

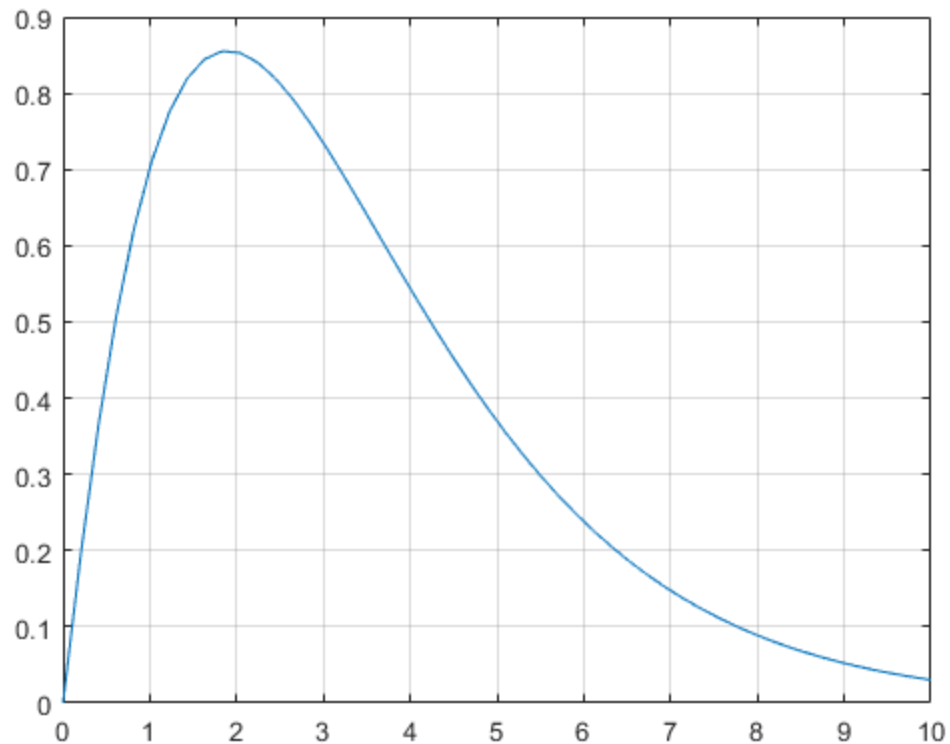
Solution - Problem 2 (do not modify this line)

```
func2 = @(x)(x.*exp(-0.7.*x).*sqrt(1+x));
func2(8)
x = linspace(0,10,50); % your turn
y = func2(x);
```

```
plot(x,y); % your turn
grid on;
```

```
ans =
```

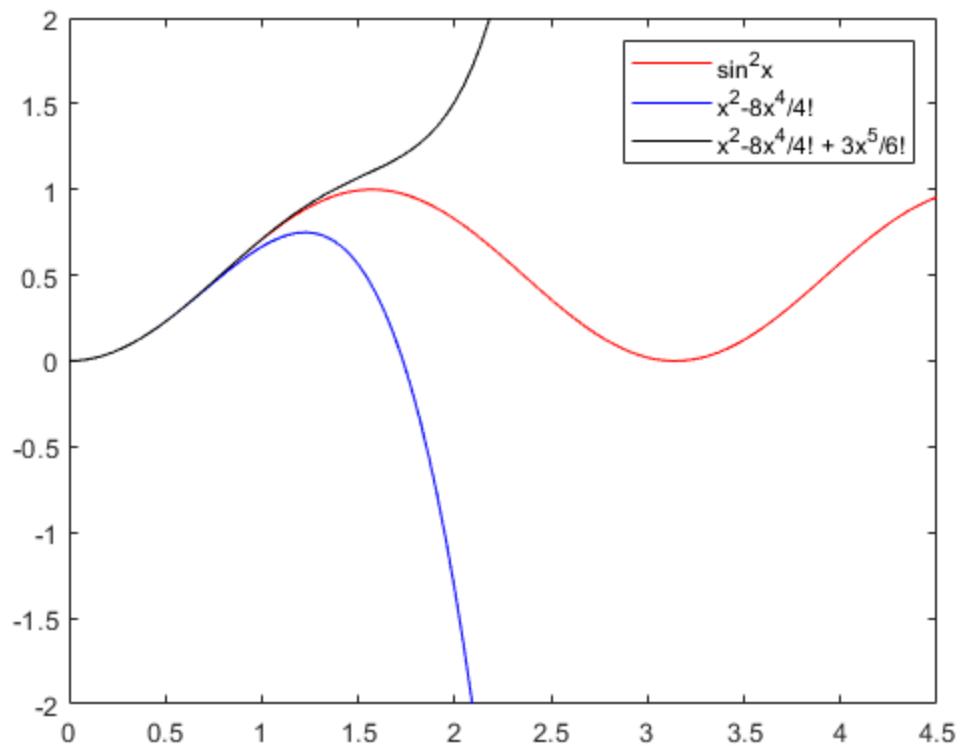
```
0.088748729195590
```



Solution - Problem 3 (do not modify this line)

```
x = linspace(0,4.5,100);
y1 = sin(x).^2;
y2 = x.^2-2^3*x.^4/factorial(4);
y3 = x.^2-2^3*x.^4/factorial(4)+ 2^5*x.^6/factorial(6);

plot(x,y1,'r'); hold on;
plot(x,y2,'b');
plot(x,y3,'k'); hold off;
ylim([-2 2]);
legend('sin^2x', 'x^2-8x^4/4!', 'x^2-8x^4/4! + 3x^5/6!')
```



Solution - Problem 4 (do not modify this line)

```
A = [];
rows = 4; cols = 6;

for i = 1:rows
    for j = 1:cols
        A(i,j) = 2*i-3*j;
    end
end
A
```

A =

-1	-4	-7	-10	-13	-16
1	-2	-5	-8	-11	-14
3	0	-3	-6	-9	-12
5	2	-1	-4	-7	-10

Solution - Problem 5 (do not modify this line)

```
% part b(i)
```

```
myfunc = @(x)(x^3);
x0 = 0.6; h = x0/100;
dfdx2pt = twoptderi(myfunc,x0,h);
fprintf('Approximate derivative of x^3 at x= 0.6 is %4.4f \n',
    dfdx2pt)
% part b(ii)

myfunc = @(x)(x^3*exp(2*x));
x0 = 2.5; h = x0/100;
dfdx2pt = twoptderi(myfunc,x0,h);
fprintf('Approximate derivative of x^3exp(2x) at x= 2.5 is %4.6f \n',
    dfdx2pt)
```

```
function dfdx2pt = twoptderi(myfunc,x0,h)
% dfdx2pt approximates the derivative using
% the two-point central difference formula
% x0 is the point where the derivative is calculated
% h is a number relative to x0
dfdx2pt = (myfunc(x0+h)-myfunc(x0-h))/(2*h);
end
```

```
Approximate derivative of x^3 at x= 0.6 is 1.0800
Approximate derivative of x^3exp(2x) at x= 2.5 is 7427.554645
```

Published with MATLAB® R2018a