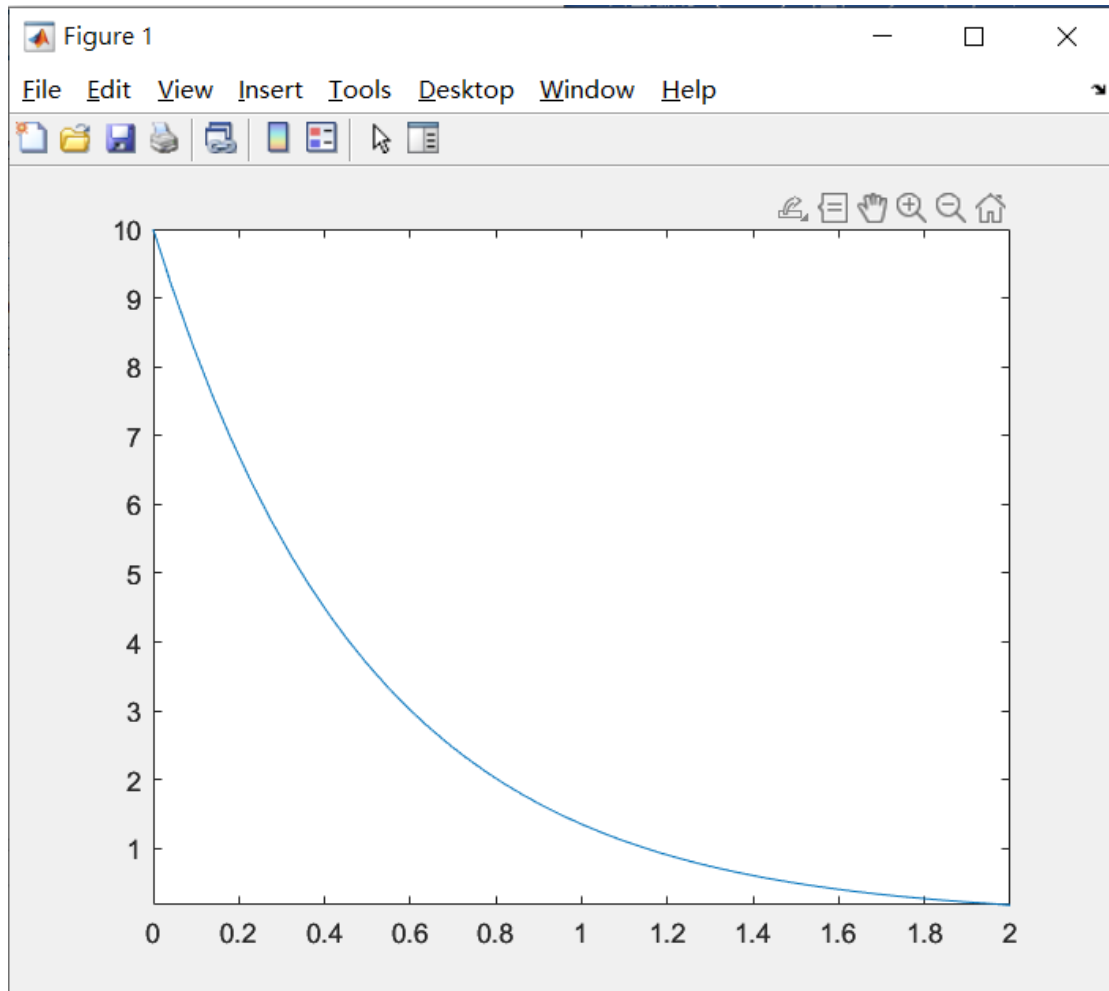


1.

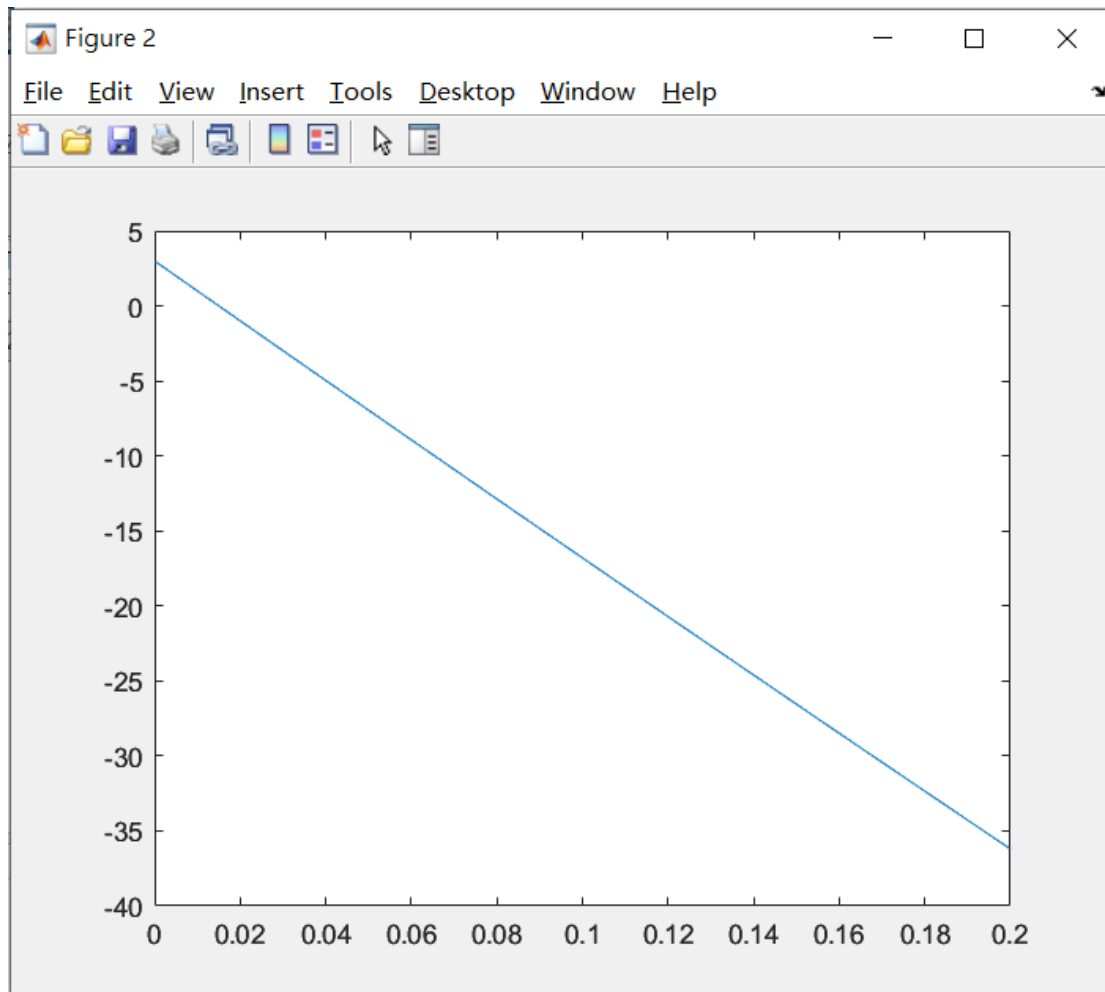
% (a)

```
y = @(x) 10*exp(-2*x) ;
```

```
fplot(y , [0 2])
```



```
% (b)
figure
y = @(x) 20*x.^2-200*x+3 ;
x = 0:0.01:0.2 ;
plot(x , y(x))
[t , index] = min(abs(y(x))) ;
fprintf("f(x)=0 is approximated at x=%f\n" , x(index))
```



```
f(x)=0 is approximated at x=0.020000
```

```
%(c)
```

```
fzero(y , 0)
```

```
fzero(y , 100)
```

```
>> d1166506_mid_1
```

```
ans =
```

```
0.0150
```

```
ans =
```

```
9.9850
```

2.

```
wage = [5 5.5 6.5 6 6.25] ;  
hour = [40 43 37 50 45] ;  
product = [1000 1100 1000 1200 1100] ;
```

% (a)

```
salary = wage.*hour
```

```
salary =  
  
    200.0000    236.5000    240.5000    300.0000    281.2500
```

% (b)

```
fprintf("Total salary paid: ") ;
```

```
disp(sum(salary))
```

```
Total salary paid:    1.2582e+03
```

% (c)

```
units = sum(product)
```

```
units =  
  
    5400
```

% (d)

```
fprintf("Average cost per unit: ") ;
```

```
disp(sum(salary)/sum(units))
```

```
Average cost per unit:    0.2330
```

% (e)

```
fprintf("Average hours per unit: ") ;
```

```
disp(sum(hour)/sum(units))
```

```
Average hours per unit:    0.0398
```

```
% (f)
efficient = product./hour ;
fprintf("Most efficient: %d\n" , find(efficient==max(efficient))) ;
fprintf("Most inefficient: %d\n" , find(efficient==min(efficient))) ;
Most efficient: 3
Most inefficient: 4
```

3.

```
A = 2000 ;  
L = @(R) (A-0.5*pi*(R.^2))./(2.*R) ;  
cost = @(R) 2*L(R)*40 + 2*R*40 + pi*R*50 ;  
R = fminsearch(cost , 0) ;  
fprintf("Radius: %f\n" , R) ;  
fprintf("Length: %f\n" , L(R)) ;  
fprintf("Cost: %f\n" , cost(R)) ;
```

```
>> d1166506_mid_3  
Radius: 21.427000  
Length: 29.841363  
Cost: 7467.214305
```

4.

calculate.m

```
function y = calculate(a , n)
    y = a^(n+1)/factorial(n) ;
end
```

```
a = 2 ;
disp(" n   x")
for n=0:20
    fprintf("%2d %f\n" , n , calculate(a , n)) ;
end
```

```
>> d1166506_mid_4
```

n	x
0	2.000000
1	4.000000
2	4.000000
3	2.666667
4	1.333333
5	0.533333
6	0.177778
7	0.050794
8	0.012698
9	0.002822
10	0.000564
11	0.000103
12	0.000017
13	0.000003
14	0.000000
15	0.000000
16	0.000000
17	0.000000
18	0.000000
19	0.000000
20	0.000000

5.

$A = \begin{bmatrix} 2 & -3 & 6 & 2 & 5 \\ -2 & 3 & -3 & -3 & -4 \\ 4 & -6 & 9 & 5 & 9 \\ -2 & 3 & 3 & -4 & 1 \end{bmatrix}$;

% (a)

$B = A(:, 3:5)$

B =

6	2	5
-3	-3	-4
9	5	9
3	-4	1

% (b)

$C = A(2:4, :)$

C =

-2	3	-3	-3	-4
4	-6	9	5	9
-2	3	3	-4	1

% (c)

$D = A(1:2, 3:5)$

D =

6	2	5
-3	-3	-4

6.

```
str = 'ABABA' ;
newstr = lower(str) ;
result = pal1(str , 1 , length(newstr)) ;
if result==1
    fprintf("%s is palindrome.\n" , str) ;
else
    fprintf("%s isn't palindrome.\n" , str) ;
end
```

```
function is_pal = pal1(str , l , r)
    if (l>r)
        is_pal = 1 ;
    elseif(str(l , l)==str(r , r))
        is_pal = pal1(str , l+1 , r-1) ;
    else
        is_pal = 0 ;
    end
end
```

```
>> d1166506_mid_6
ABBA is palindrome.
>> d1166506_mid_6
ABBAA isn't palindrome.
>> d1166506_mid_6
ABABA is palindrome.
```