Laplace Transformation

Q1: Find the Laplace transform of $g(t) = \mathrm{te}^{-3t}$.

Sol^n :

The solution can be obtained by *translation and change of scale properties* of Laplace transformation.

The first translation or shifting property: If $\mathrm{L}\langle\{F(t)\} = \int_0^\infty e^{-\mathrm{st}}F(t)\mathrm{dt} = f(s)$**, then we have**

$\mathrm{L}\langle\{e^{\mathrm{at}}F(t)\} = f(s-a).$

The code of Laplace transformation for given function is below:

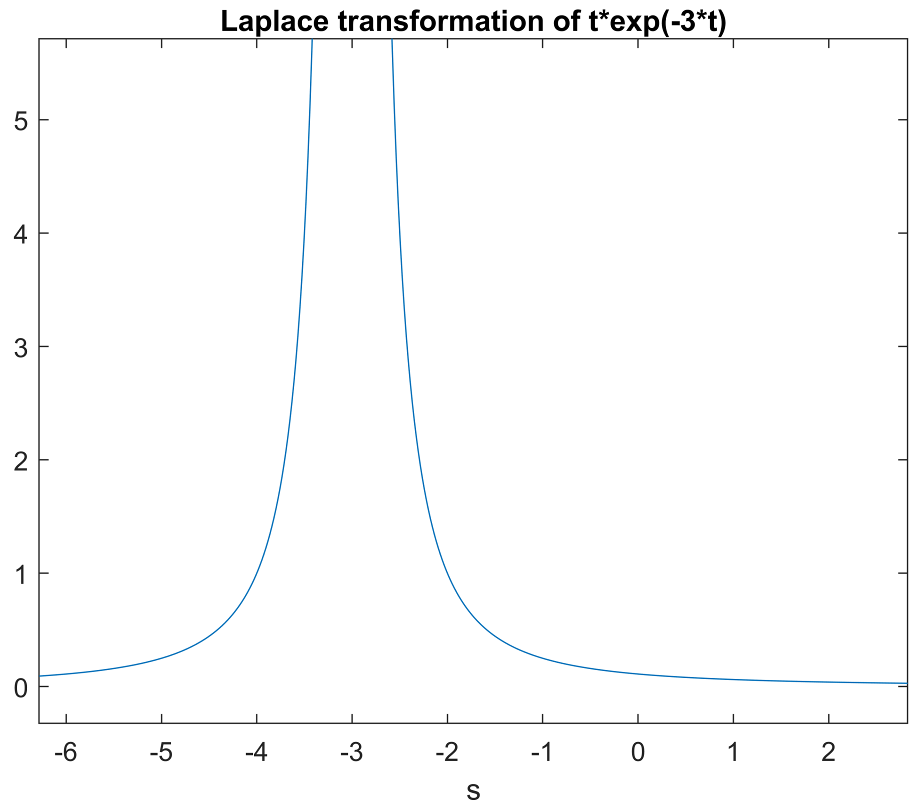
```
clear all;

% define the function
syms t;
g(t) = t * exp(-3*t);
lap = laplace(g); % Laplace transformation
fprintf('Laplace tranform: ');
```

```
Laplace tranform:
```

```
disp(lap)
```

$$\frac{1}{(s+3)^2}$$

```

% ploting the solution
ezplot(lap);
title('Laplace transformation of t*exp(-3*t)');
```



Q2: Find the Laplace transform of $f(t) = 8\sin(5t) - e^{-t}\cos(2t)$

Sol^n :

The solution can be obtained by the *linear property* and *translation properties* of Laplace transformation respectively.

The first translation or shifting property: If $\mathrm{L}\{F(t)\} = \int_0^\infty e^{-\mathrm{st}}F(t)\mathrm{dt} = f(s)$**, then we have**

$\mathrm{L}\{e^{\mathrm{at}}F(t)\} = f(s-a)$.

and the linear property: If c_1 **and** c_2 **are any constants while** $F_1(t)$ **and** $F_1(t)$ **are functions with Laplace transforms** $f_1(s)$ **and** $f_2(s)$ **respectively, then**

$\mathrm{L}\{c_1F_1(t) + c_2F_2(t)\} = c_1\mathrm{L}\{F_1(t)\} + c_2\mathrm{L}\{F_2(t)\} = c_1f_1(s) + c_2f_2(s)$.

The code of Laplace transformation for the given function is given below:

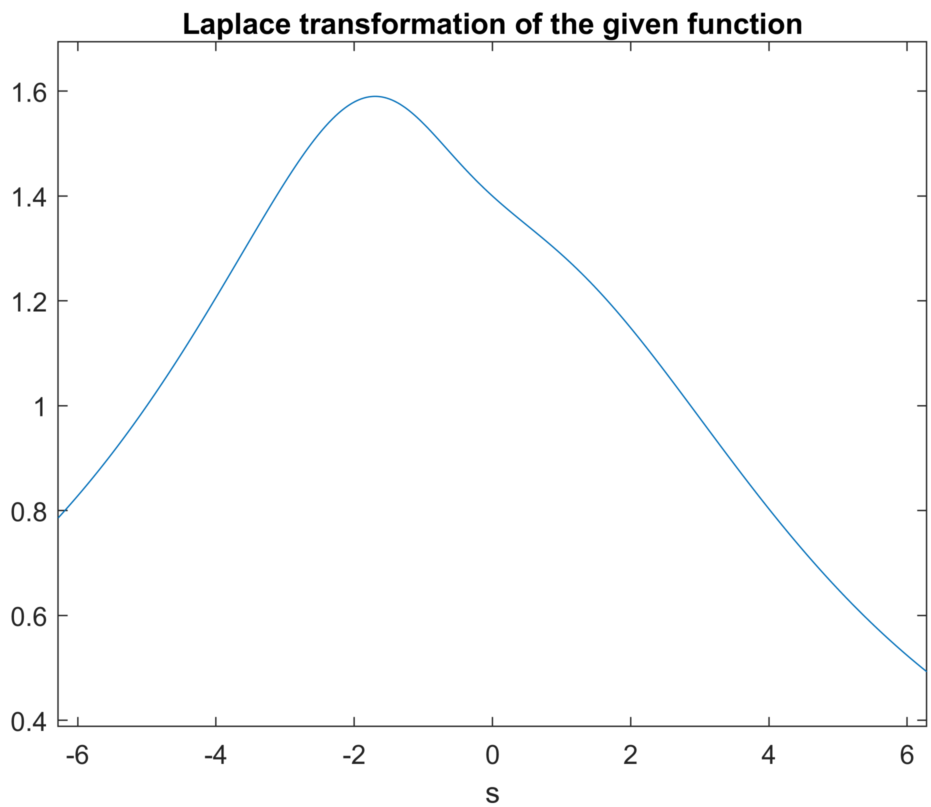
```
clear all;

% define function
syms t;
f(t) = 8 * sin(5*t) - exp(-t) * cos(2*t);
lap = laplace(f); % Laplace transformation
fprintf('Laplace tranform: ');
```

Laplace tranform:

```
disp(lap)
```

$$\frac{40}{s^2+25} - \frac{s+1}{(s+1)^2+4}$$

```
% ploting the solution
ezplot(lap);
title('Laplace transformation of the given function');
```



Q3: What is the inverse Laplace transform of $f(s) = \frac{1}{s} - \frac{2s^2+3}{s^2+9}$.

Sol^n :

The inverse Laplace transform of this function can be obtained by *Linearity property* and then making a partial fraction of the second term. So, we get

$$f(s) = \frac{1}{s} - \frac{2s^2}{s^2+9} + \frac{3}{s^2+9}.$$

Code of inverse Laplace for the given function is below:

```
clear all;
```

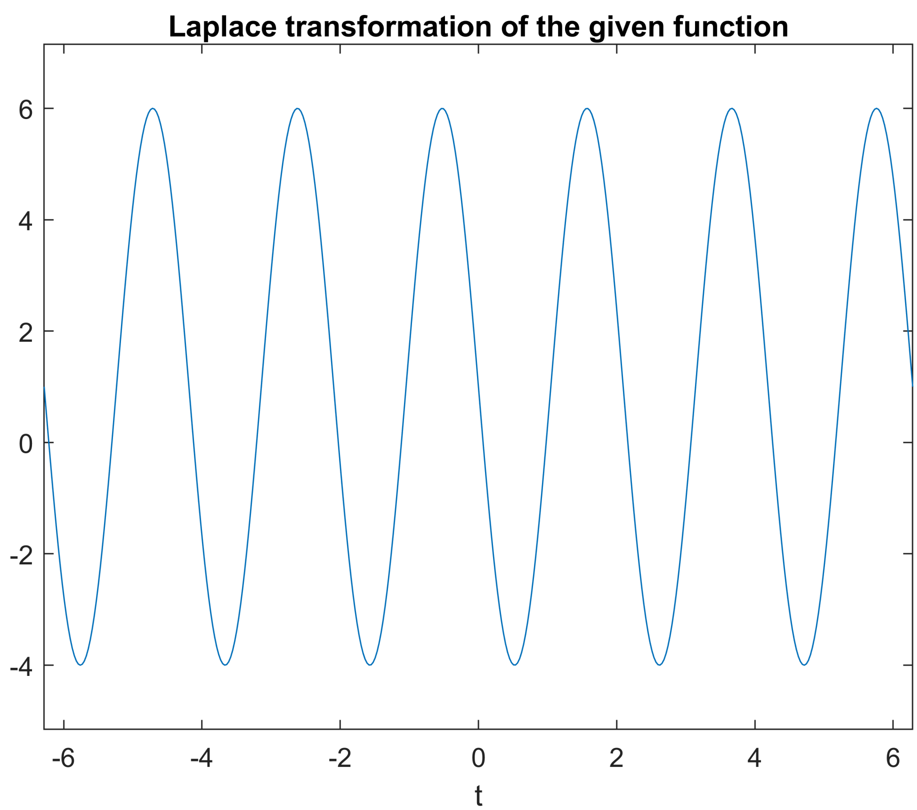
```
syms s t;
f1(s) = (1/s);
f2(s) = (2*s^2 / (s^2+9));
f3(s) = (3 / (s^2+9));
ilf1 = ilaplace(f1);
ilf2 = ilaplace(f2);
ilf3 = ilaplace(f3);
fprintf('The inverse Laplace of f(s): ');
```

```
The inverse Laplace of f(s):
```

```
f(t) = ilf1 + ilf2 + ilf3;
disp(f)
```

$2\,\delta(t) - 5\sin(3\,t) + 1$

```
% ploting the solution
ezplot(f);
title('Laplace transformation of the given function');
```



Q4: What is the inverse Laplace transform of $f(s) = \dfrac{s}{s^2 - 49} - \dfrac{3}{s^2 - 9}$.

Sol^n :

The code of inverse Laplace transfor if the given function is below:

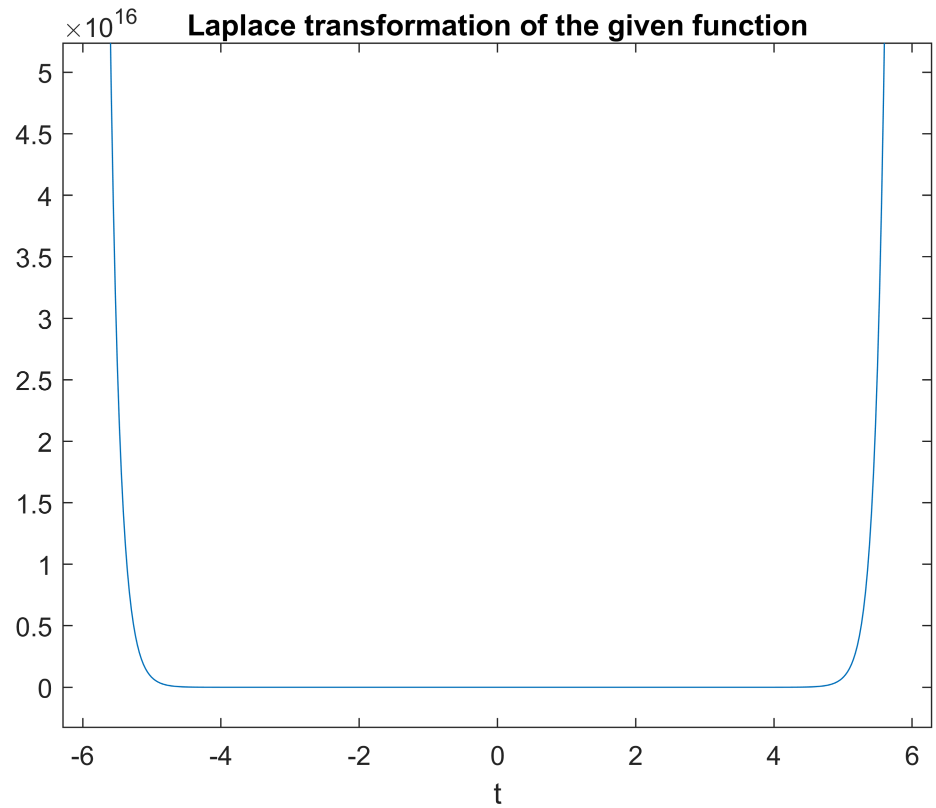
```
clear all;

% define the function
syms s;
f(s) = (s / (s^2-49)) - (3 / (s^2 - 9));
ilf = ilaplace(f); % performing Laplace transformation
fprintf('The inverse Laplace of f(s): ');
```

```
The inverse Laplace of f(s):
```

```
disp(ilf)
```

$$\frac{e^{-3t}}{2} - \frac{e^{3t}}{2} + \frac{e^{-7t}}{2} + \frac{e^{7t}}{2}$$

```

% ploting the solution
ezplot(ilf);
title('Laplace transformation of the given function');
```



Q5: Solve the differential equation:

$$\frac{dy}{dt} + y = 2U(t)$$

where $U(t)$ is the heaviside function. Let all initial conditions be zero. What is the forced response?

Sol^n :

Given, $\frac{\mathrm{dy}}{\mathrm{dt}} + y = 2U(t)$

$\Rightarrow y' + y = 2U(t)$

$\Rightarrow L^{-1}\{y'\} + L^{-1}\{y\} = 2\,L^{-1}\{U_{a}(t)\}$

$\Rightarrow \mathrm{sY}(s) + Y(s) = 2\,.\frac{e^{-\mathrm{as}}}{s}$

$\Rightarrow Y(s)(s+1) = 2.\frac{e^{-\mathrm{as}}}{s}$

$\Rightarrow Y(s) = \frac{2e^{-\mathrm{as}}}{s(s+1)}$

$\Rightarrow Y(s) = 2F(s)e^{-\mathrm{as}}$ **where** $F(s) = \frac{1}{s(s+1)}$

$\Rightarrow L^{-1}\{Y(S)\} = L^{-1}\{2F(s)e^{-\mathrm{as}}\}$

Now the code for solving the following function Y(s) is:

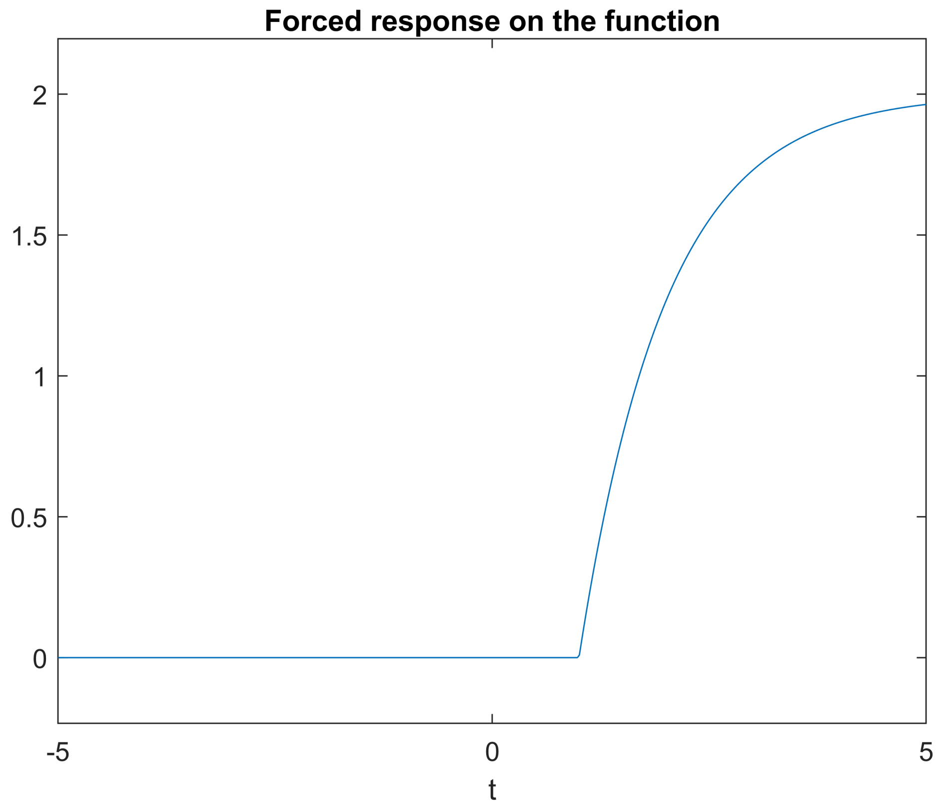
```
clear all;

% define the function
syms s;
a = 1;
y(s) = (2*exp(-a*s)) / (s*(s+1));

% performing inverse Laplace tranform
lt = ilaplace(y);

fprintf('The solution: ');
```

```
The solution:
```

```
disp(lt)
```

$-2\,\mathrm{heaviside}(t-1)\;(\mathrm{e}^{1-t}-1)$

```
% ploting the solution
ezplot(lt, [-5 5]);
title('Forced response on the function');
```



Fourier Transformation

Q1: Find the fourier transfor of x^2.

Sol^n :

The code for the following problem is below:

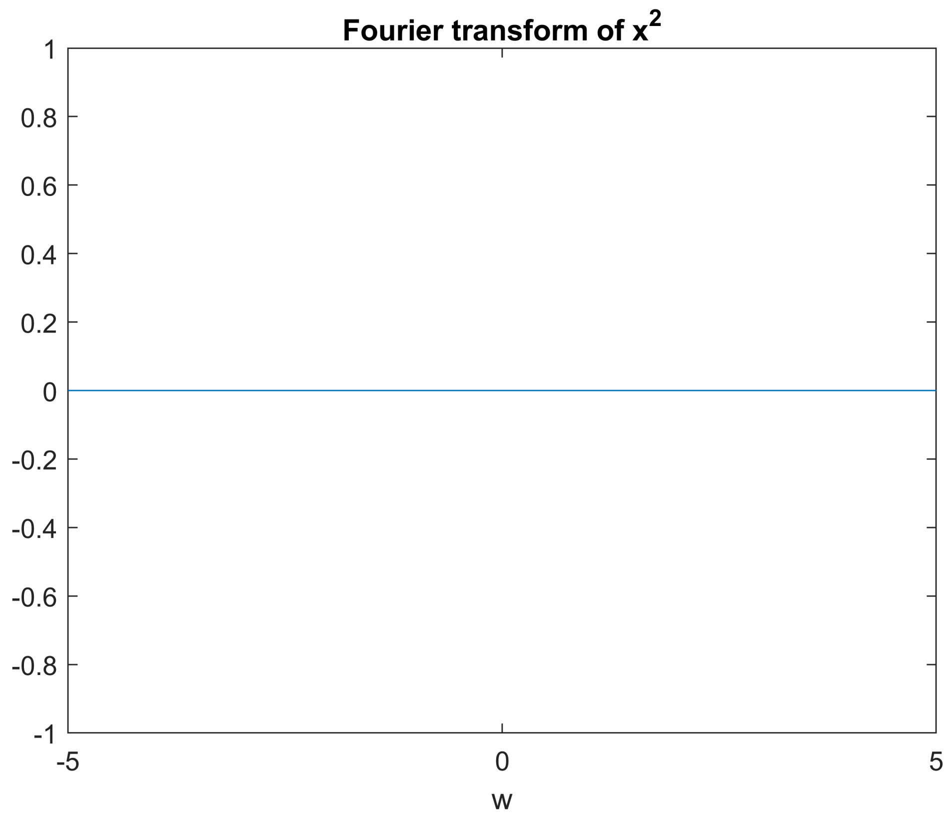
```
clear all;

% define the function
syms x;
f = x^2;

% perform fourier transform
ft = fourier(f);
fprintf('The Fourier transform of the f(x): ');
```

```
The Fourier transform of the f(x):
```

```
disp(ft);
```

$-2\,\pi\,\delta''(w)$

```
% plot the solution
ezplot(ft, [-5 5]);
title('Fourier transform of x^2');
```



Q2: Find the fourier transfor of $x \cos x$.

Sol^n :

The code for the following problem is below:

```
clear all;

% define the function
syms x;
f = x * cos(x);

% performing fourier transform
ft = fourier(f);
fprintf('The Fourier transform of the f(x): ');
```

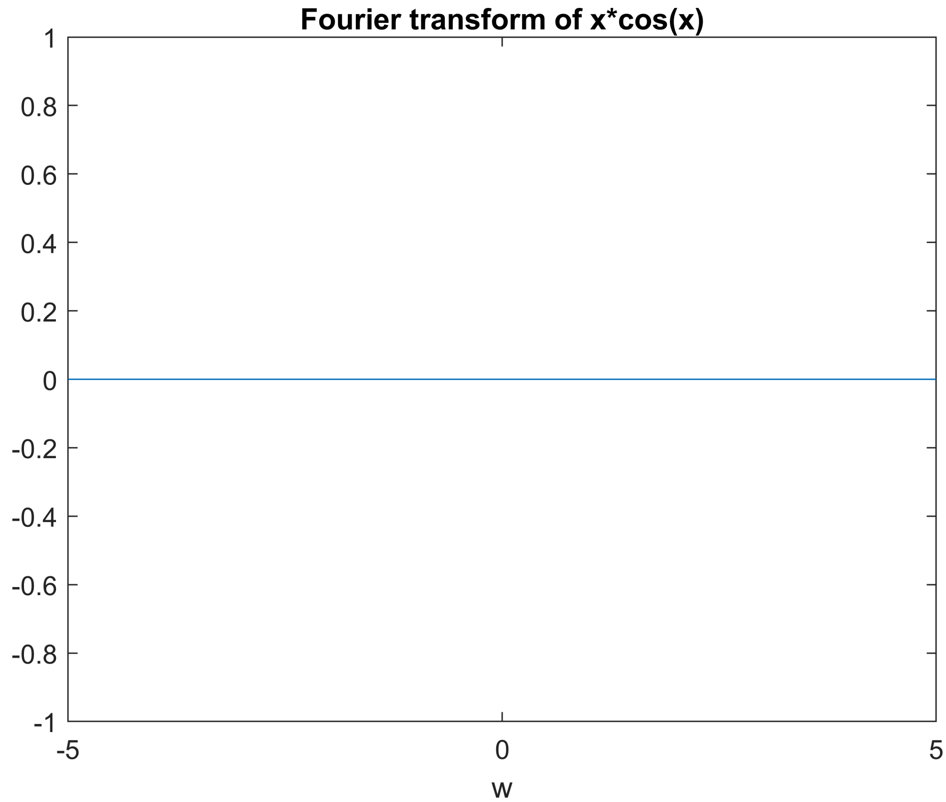
```
The Fourier transform of the f(x):
```

```
disp(ft);
```

$\pi\ (\delta'(w-1)+\delta'(w+1))\,\mathrm{i}$

```

% ploting the solution
```

```
ezplot(ft, [-5 5]);
title('Fourier transform of x*cos(x)');
```



Q3: Find the inverse Fourier transform of $\dfrac{1}{1+\imath\omega}$.

Sol^n :

The code for the solution of the given problem is below:

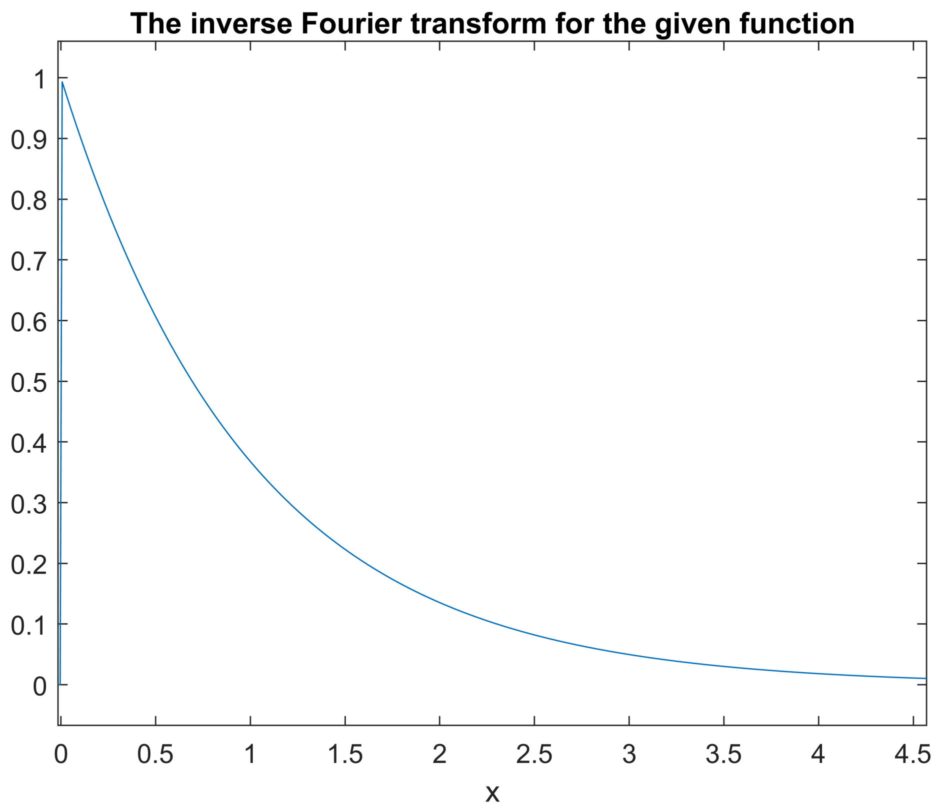
```
clear all;

% define the function
syms w;
f = 1 / (1 + i*w);

% performing inverse fourier transform
inv_f = ifourier(f);

fprintf('The solution is: ');
```

```
The solution is:
```

```
disp(inv_f);
```

$$\frac{\mathrm{e}^{-x}\,(\mathrm{sign}(x)+1)}{2}$$

```
% ploting the solution
ezplot(inv_f);
title('The inverse Fourier transform for the given function');
```



Q4: What is the fast Fourier transform of a = [2, 4, -1, 2].

Sol^n :

The code for the fast Fourier transform of a is below:

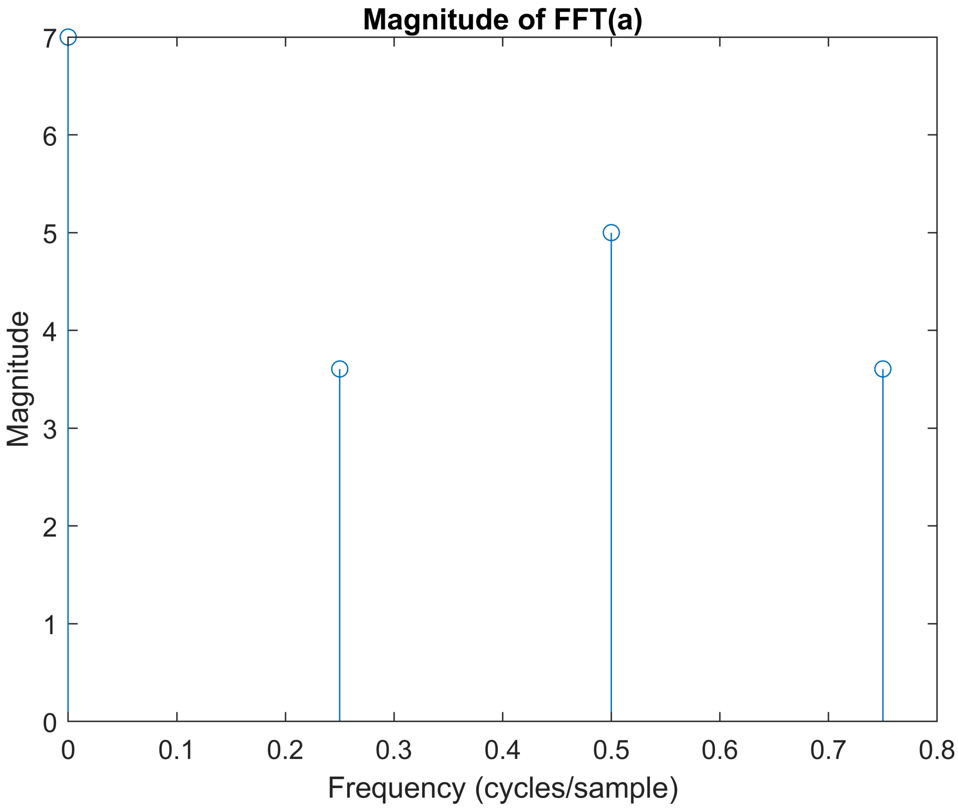
```
clear all;

% define the signal
a = [2, 4, -1, 2];

% calculating the fft of the signal
ft = fft(a);

fprintf('The fast Fourier Transform of a: ');
```

```
The fast Fourier Transform of a:
```

```
disp(ft);
```

```
    7.0000 + 0.0000i   3.0000 - 2.0000i  -5.0000 + 0.0000i   3.0000 + 2.0000i
```

```
%ploting the figure
f = (0:length(ft)-1)/length(ft);
stem(f, abs(ft));
xlabel('Frequency (cycles/sample)');
ylabel('Magnitude');
title('Magnitude of FFT(a)');
```



Q5: Let $x(t) = \sin(\pi t) + 2\sin(4\pi t)$. Add some noise to the signal using randn(size(t)). What is the highest power content of the signal and at what frequency?

Sol^n :

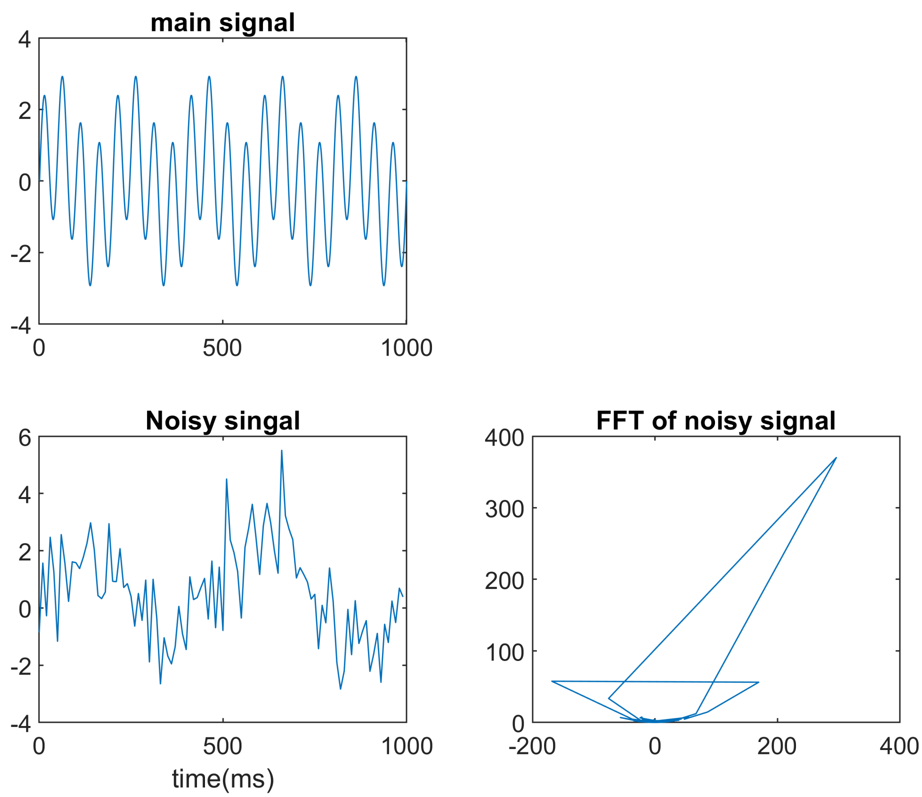
The solution code is given below:

```
clear all;

t = 0:0.01:10; % time matrix
x = sin(pi*t) + 2*sin(4*pi*t);
subplot(2, 2, 1);
plot(x); title('main signal'); % ploting the main signal
x_noisy = x + randn(size(t)); % adding noise
subplot(2, 2, 3);
plot(1000*t(1:100), x_noisy(1:100)), xlabel('time(ms)'), title('Noisy singal');

% fast Fourier transform of the noisy signal
FT = fft(x_noisy, 512);
p = FT.*conj(FT)/512;
```

```
subplot(2, 2, 4);
plot(FT, p);
```

```
Warning: Imaginary parts of complex X and/or Y arguments ignored.
```

```
title('FFT of noisy signal');
```




```
% finding the highest value of power of the signal
[~, idx] = max(FT(1:end/2));
max_freq = (idx-1) / (length(FT)/2) * (1/0.001);
fprintf('The highest power signal is at %0.3f Hz\n', max_freq);
```

```
The highest power signal is at 39.063 Hz
```