**Practical 1**

A]

clc()

*//define complex number*

a1=2;

b1=1;

a2=1;

b2=2;

z1=complex(a1,b1);

z2=complex(a2,b2);

*//Mathematical operation*

z3=z1+z2; *//z3=z1-z2,z3=z1\*z2,z3=z2/z3//*

a3=real(z3);

b3=imag(z3);

*//plot*

figure(0)

clf()

hf=gcf()

hf.background=-2

ha=gca()

ha.data\_bounds=[-5,-5;5,5];

xgrid();

plot([0 a1],[0 b1],'b','LineWidth',3)

plot([0 a2],[0 b2],'r','LineWidth',3)

plot([0 a3],[0 b3],'g','LineWidth',3)

xlabel('Real axis(Re)','FontSize',2)

ylabel('Imaginary axis(Im)','FontSize',2)

legend('$\Large {z\_{1}}$','$\Large {z\_{2}}$','$\Large {z\_{3}}$')

plot(0,0,'sk')

plot(a1,b1,'sk')

plot(a2,b2,'sk')

plot(a3,b3,'sk')

xstring(a1,b1,'$\Large{z\_{1}=2+i}$')

xstring(a2,b2,'$\Large{z\_{2}=1+2i}$')

xstring(a3,b3,'$\Large{z\_{3}=3+3i}$')

*//calculate exp form*

r1=sqrt(a1^2+b1^2);

r2=sqrt(a2^2+b2^2);

r3=sqrt(a3^2+b3^2);

phi1=atan(b1/a1)\*180/%pi;

phi2=atan(b2/a2)\*180/%pi;

phi3=atan(b3/a3)\*180/%pi;

*//Display polynomial and polar parameters*

mprintf('%s\t%s\t%s\t%s\n','a','b','r','phi')

mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a1,b1,r1,phi1)

mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a2,b2,r2,phi2)

mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f',a3,b3,r3,phi2)

B]

clc();

a1=1;b1=2;

a2=1;b2=-1;

a3=1;b3=2;

a4=1;b4=1;

z1=complex(a1,b1);

z2=complex(a2,b2);

z3=complex(a3,b3);

z4=complex(a4,b4);

n1=2;

n2=1;

n3=-1;

n4=1;

z11=z1^n1;

z22=z2^n2;

z33=z3^n3;

z44=z4^n4;

z=(z11\*z22)/(z33\*z44);

disp(z11);

disp(z22);

disp(z33);

disp(z44);

disp(z);

a=real(z);

b=imag(z);

r1=sqrt(a1^2+b1^2);

r2=sqrt(a2^2+b2^2);

r3=sqrt(a3^2+b3^2);

r4=sqrt(a4^2+b4^2);

r=sqrt(a^2+b^2);

phi1=atan(b1/a1);

phi2=atan(b2/a2);

phi3=atan(b3/a3);

phi4=atan(b4/a4);

phi=atan(b/a)

r11=r1^n1

r22=r2^n2

r33=r3^n3

r44=r4^n4

phi11=phi1\*n1

phi22=phi2\*n2

phi33=phi3\*n3

phi44=phi4\*n4

p11=r11\*(cos(phi11)+%i\*sin(phi11))

p22=r22\*(cos(phi22)+%i\*sin(phi22))

p33=r33\*(cos(phi33)+%i\*sin(phi33))

p44=r44\*(cos(phi44)+%i\*sin(phi44))

p=(p11\*p22)/(p33/p44);

disp(p)

mprintf('%s\t%s\t%s\t%s\n','a','b','r','phi')

mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a1,b1,r1,phi1)

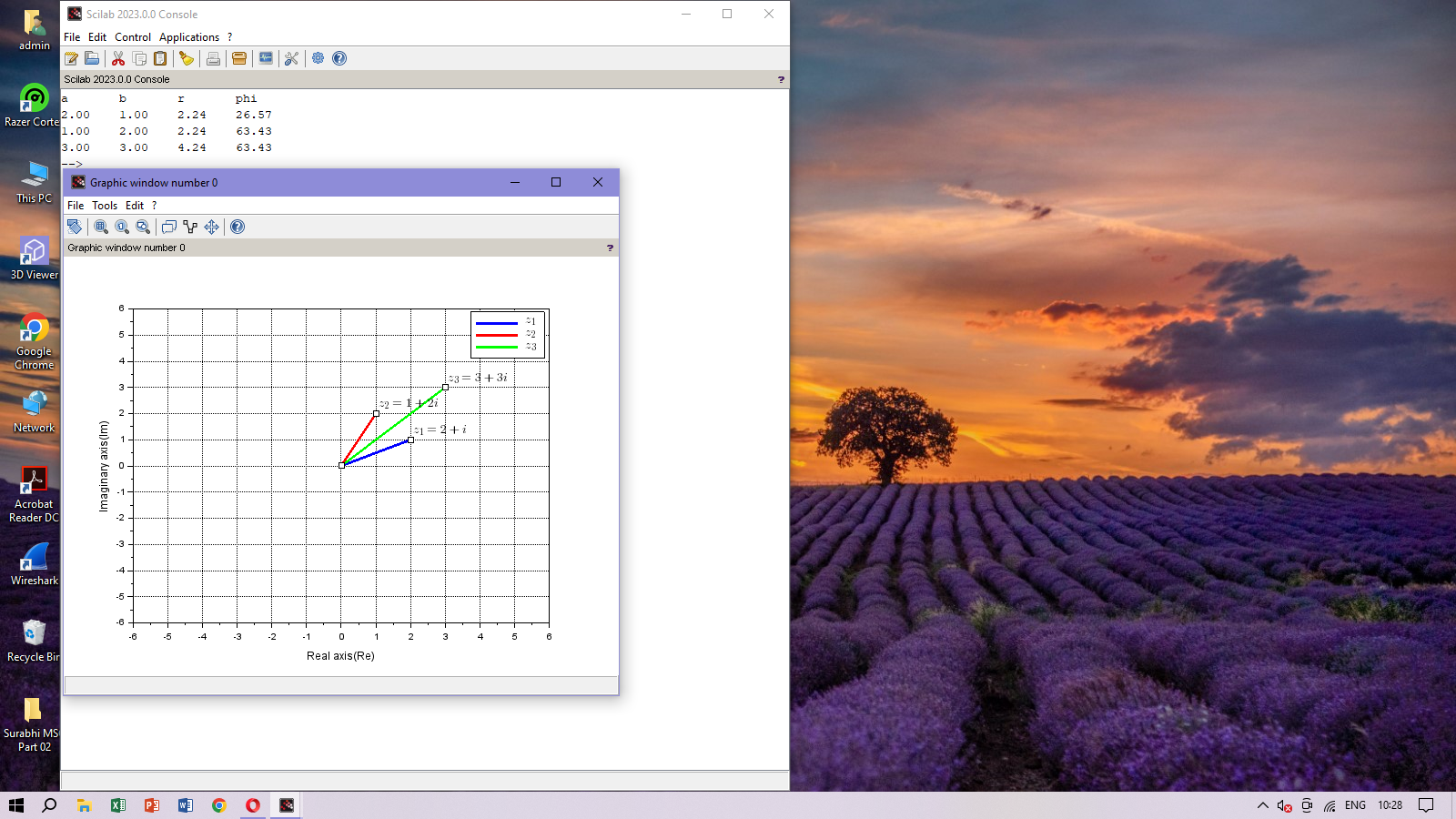
mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a2,b2,r2,phi2)

mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a3,b3,r3,phi3)

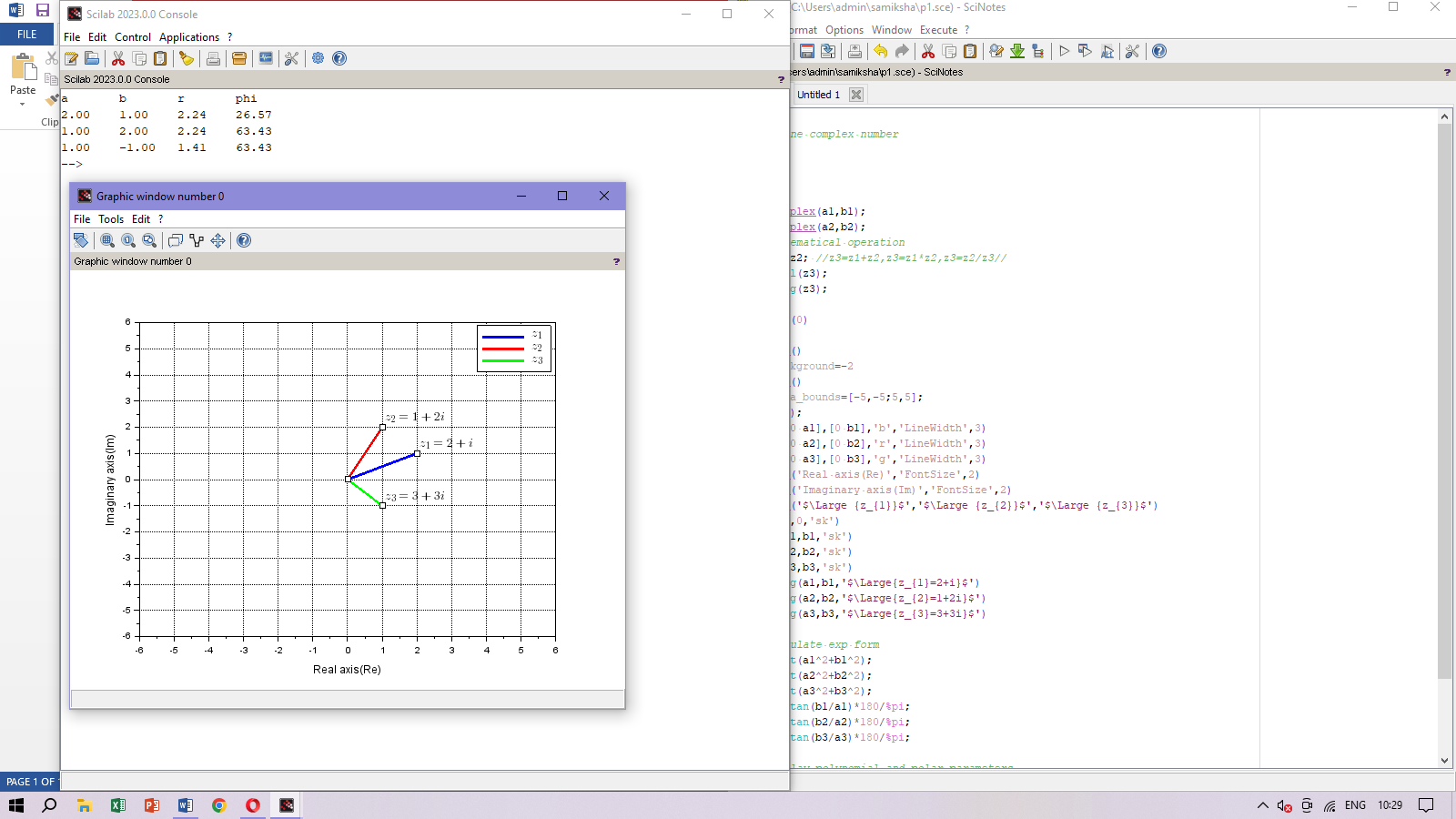
mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a4,b4,r4,phi4)

mprintf('%4.2f\t%4.2f\t%4.2f\t%4.2f\n',a,b,r,phi);

Addition output:

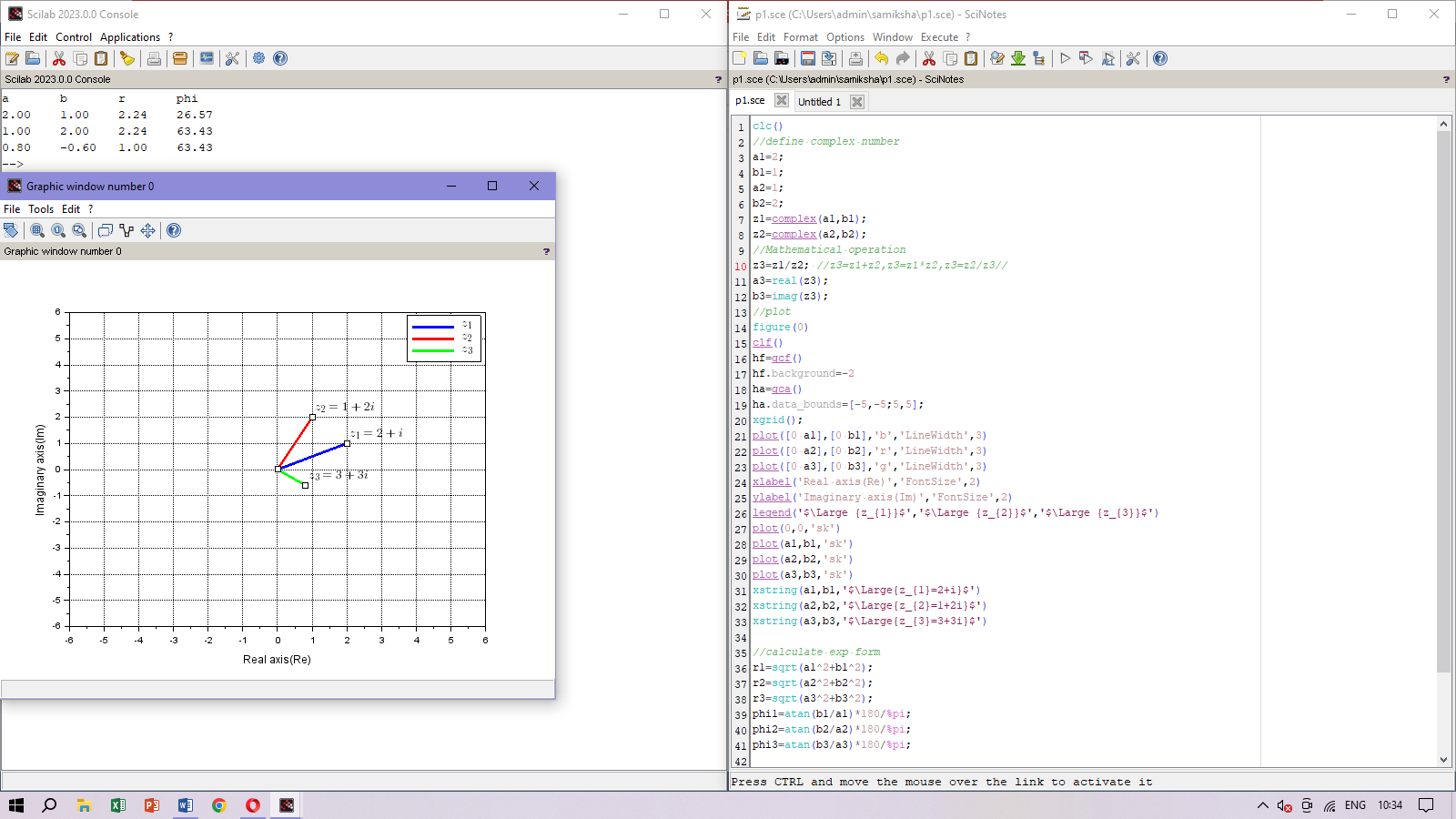


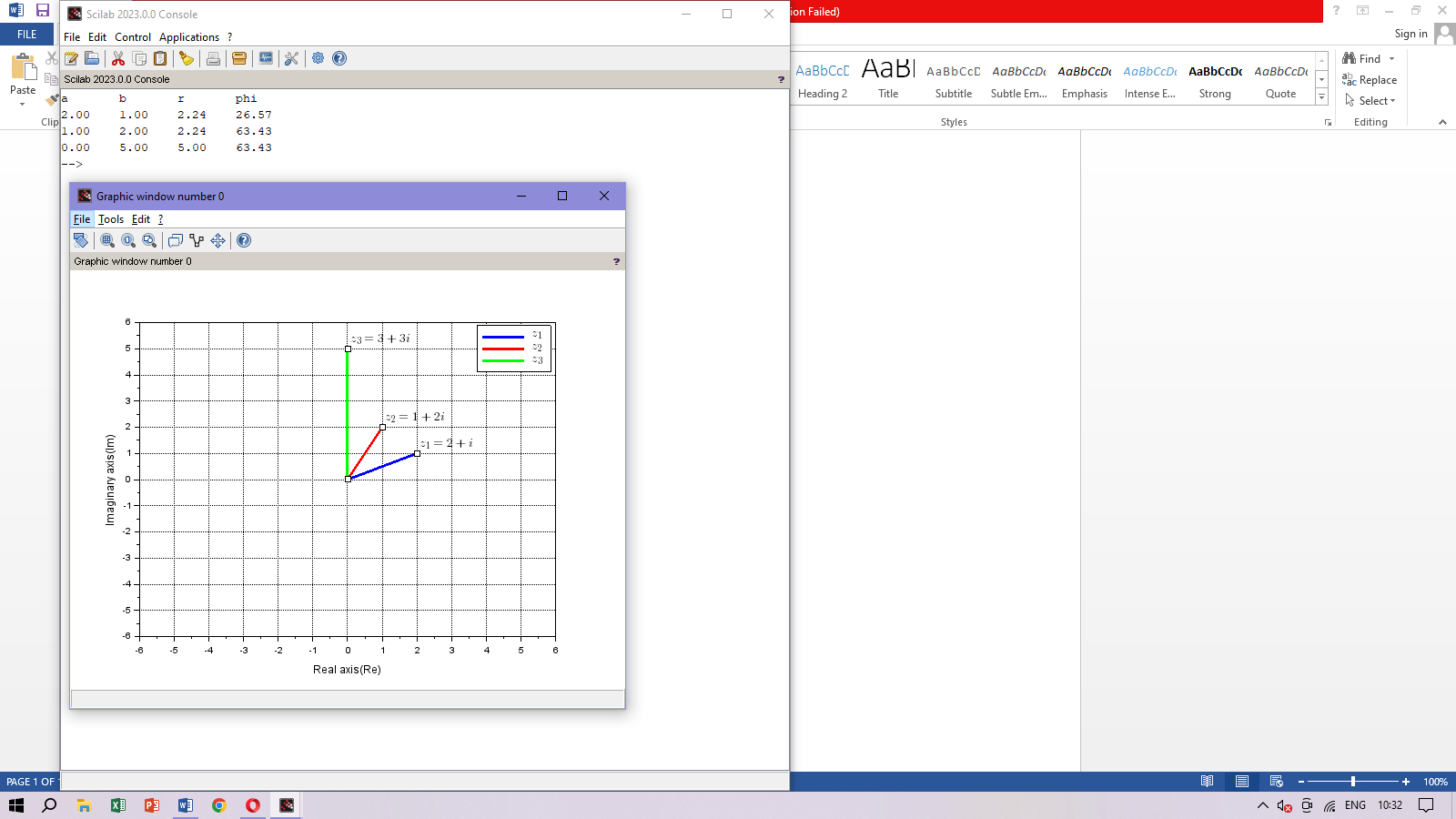
Subtraction output:

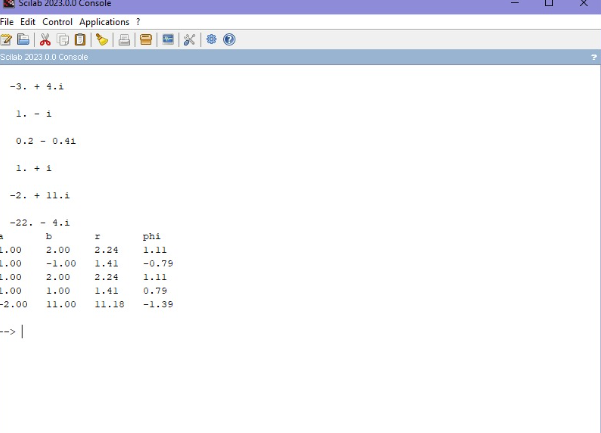


Multiplication output:

Division output:



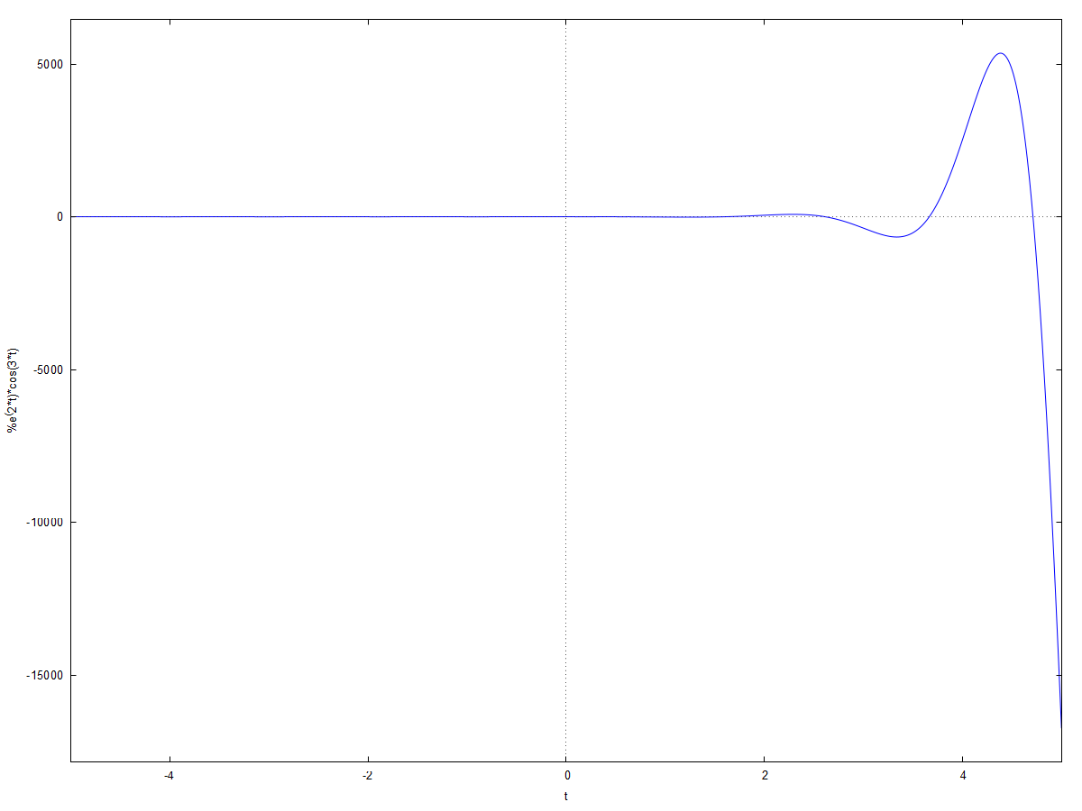


Practical1b:

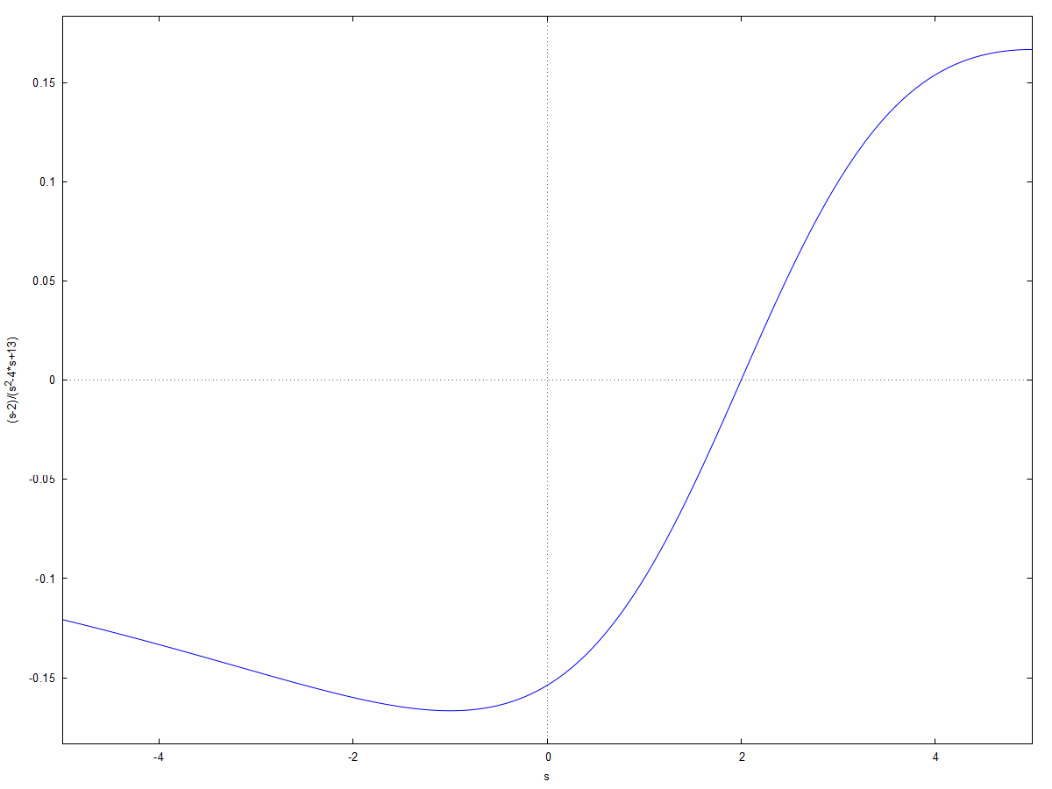
**Practical 2A**

**Laplace Transform**

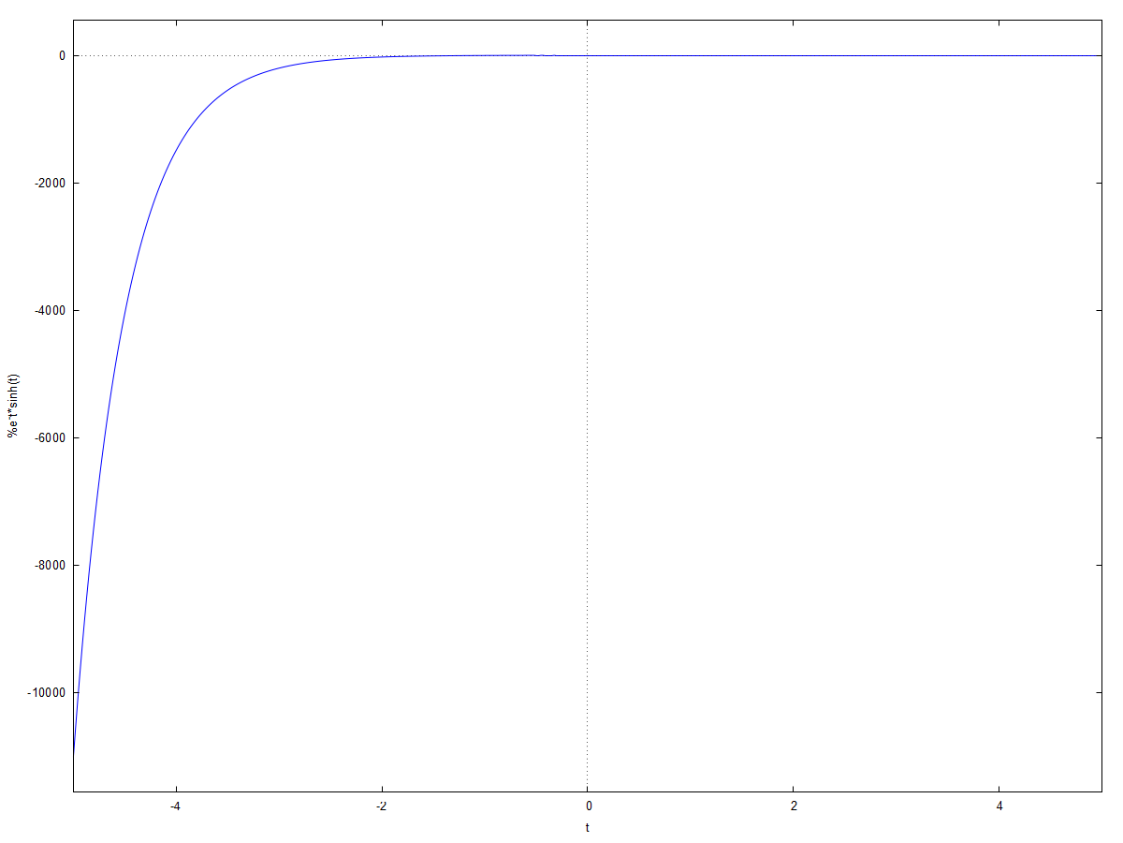
1. e^(2t) cos3t:



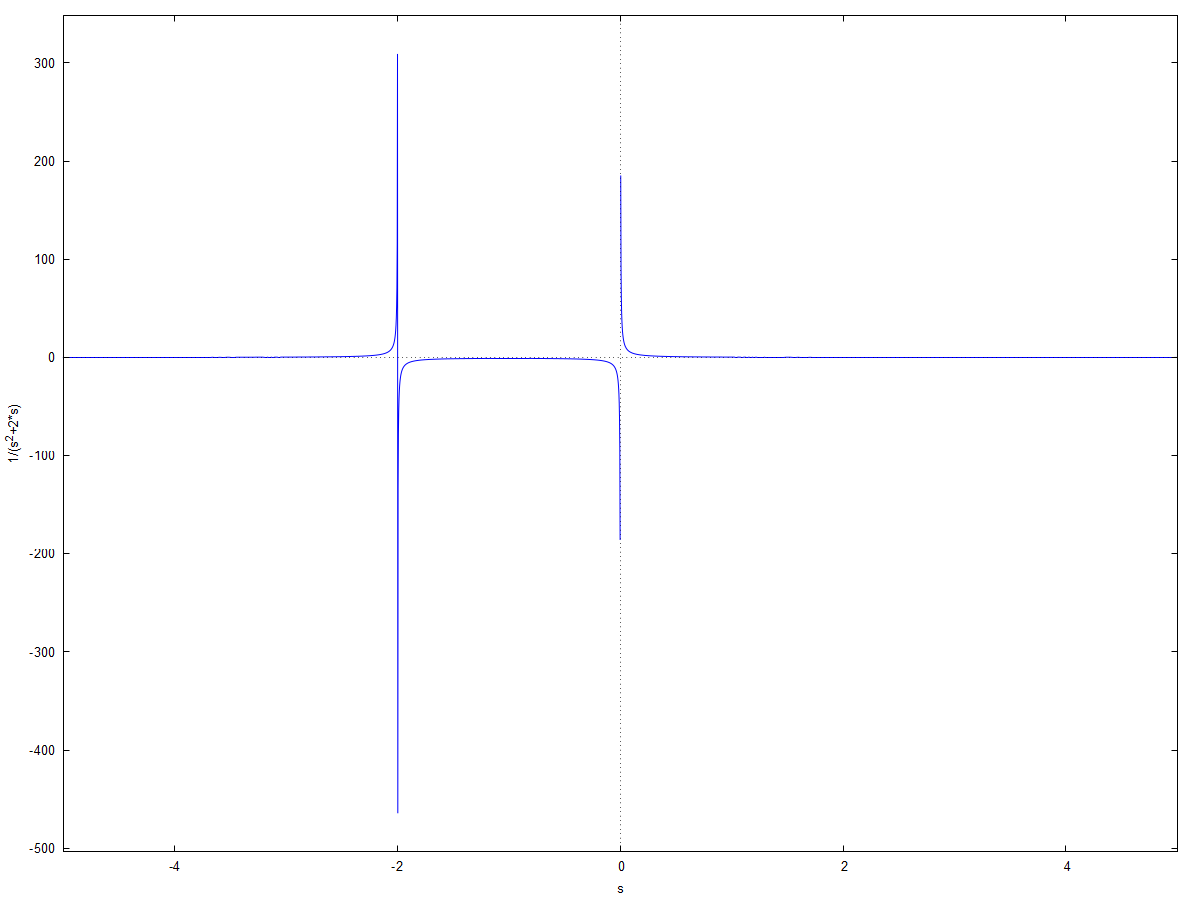
Laplace:



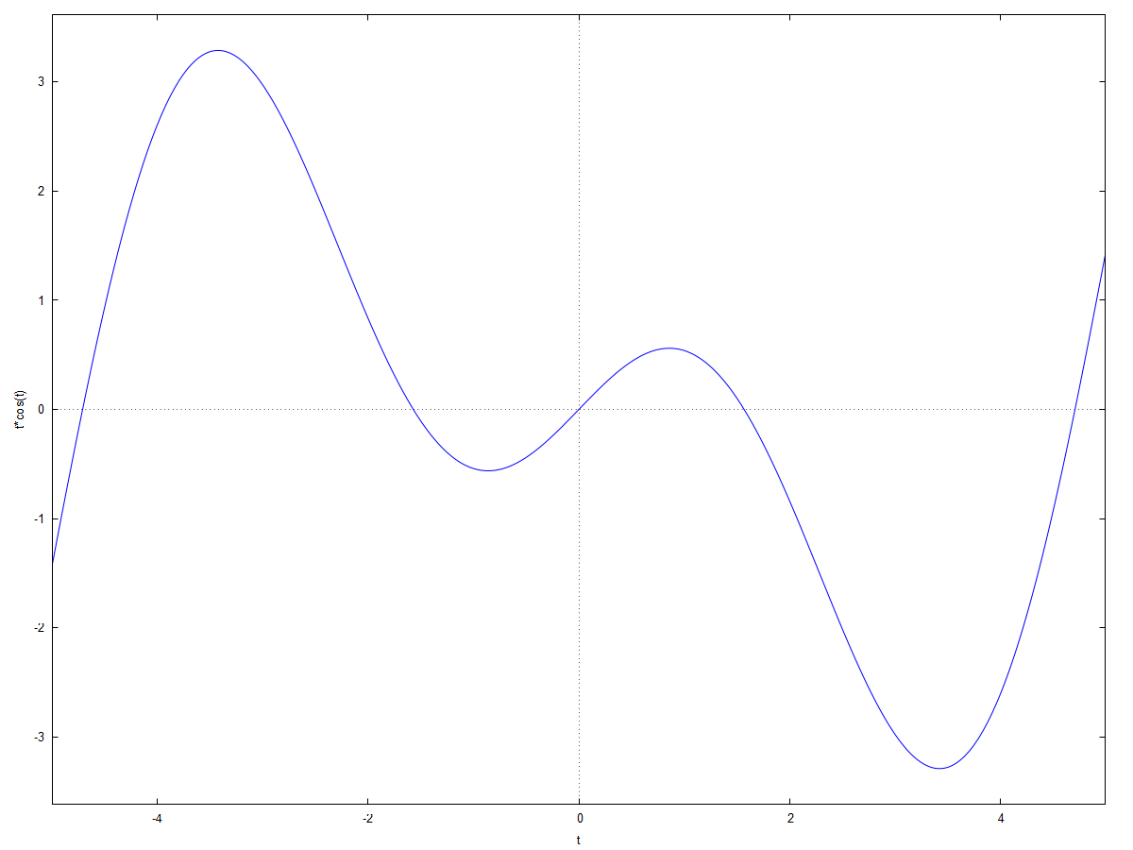
1. e^(-t) sinht:



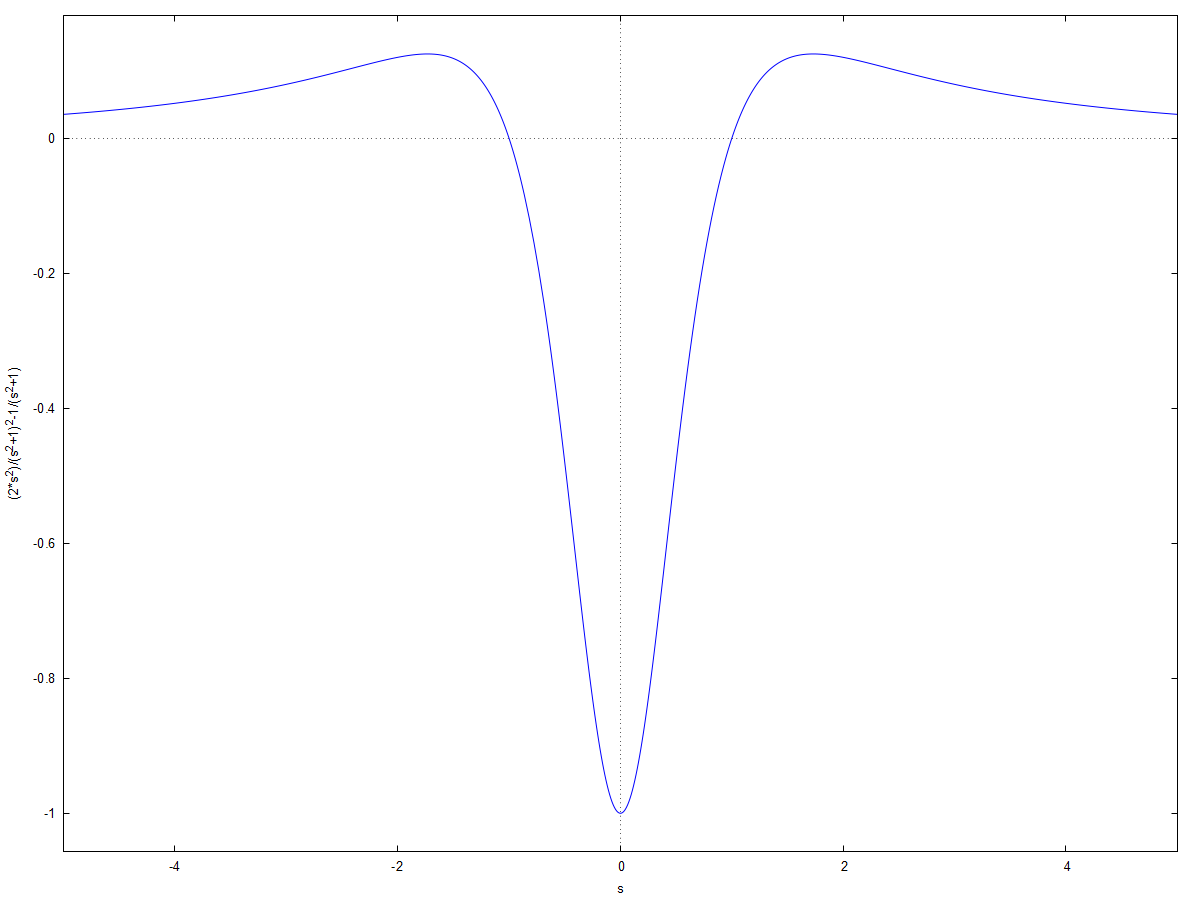
Laplace:

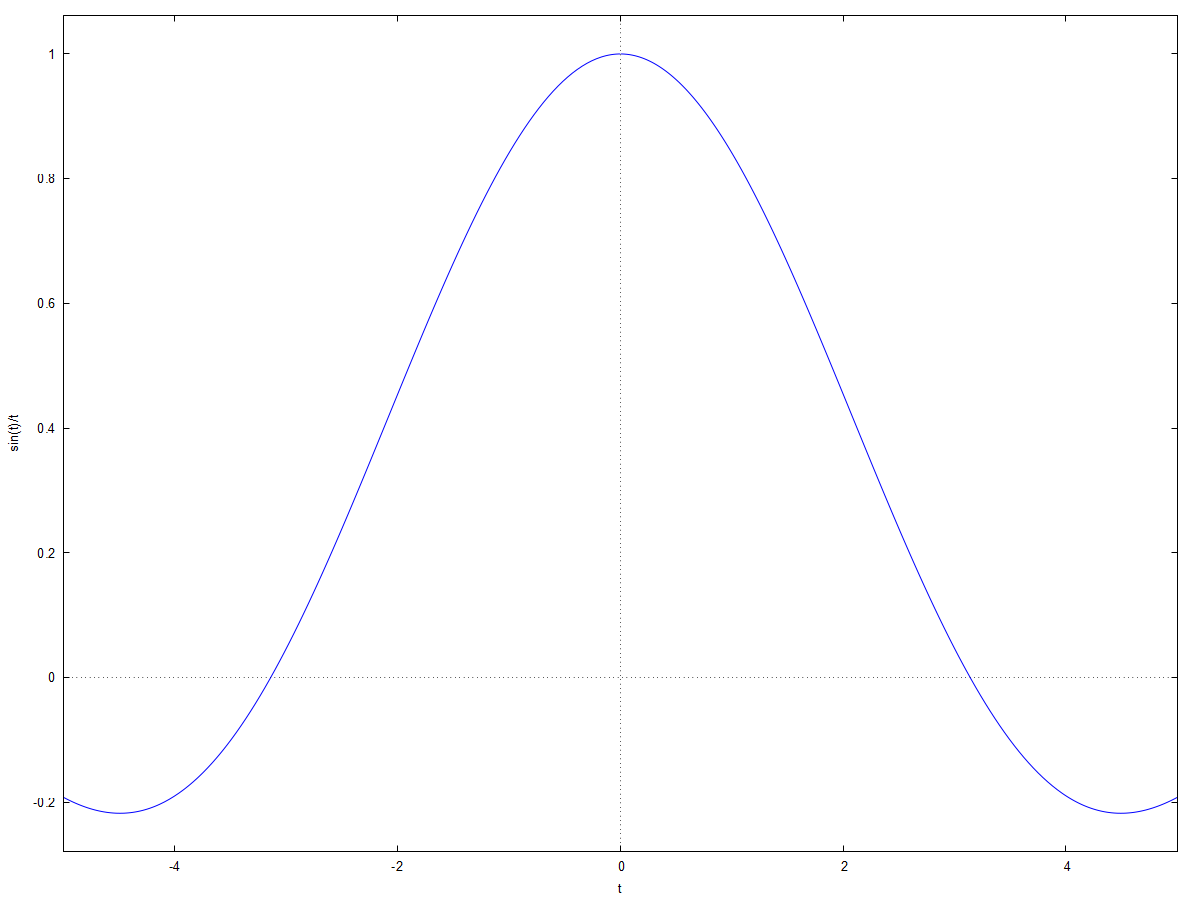


1. t cost:

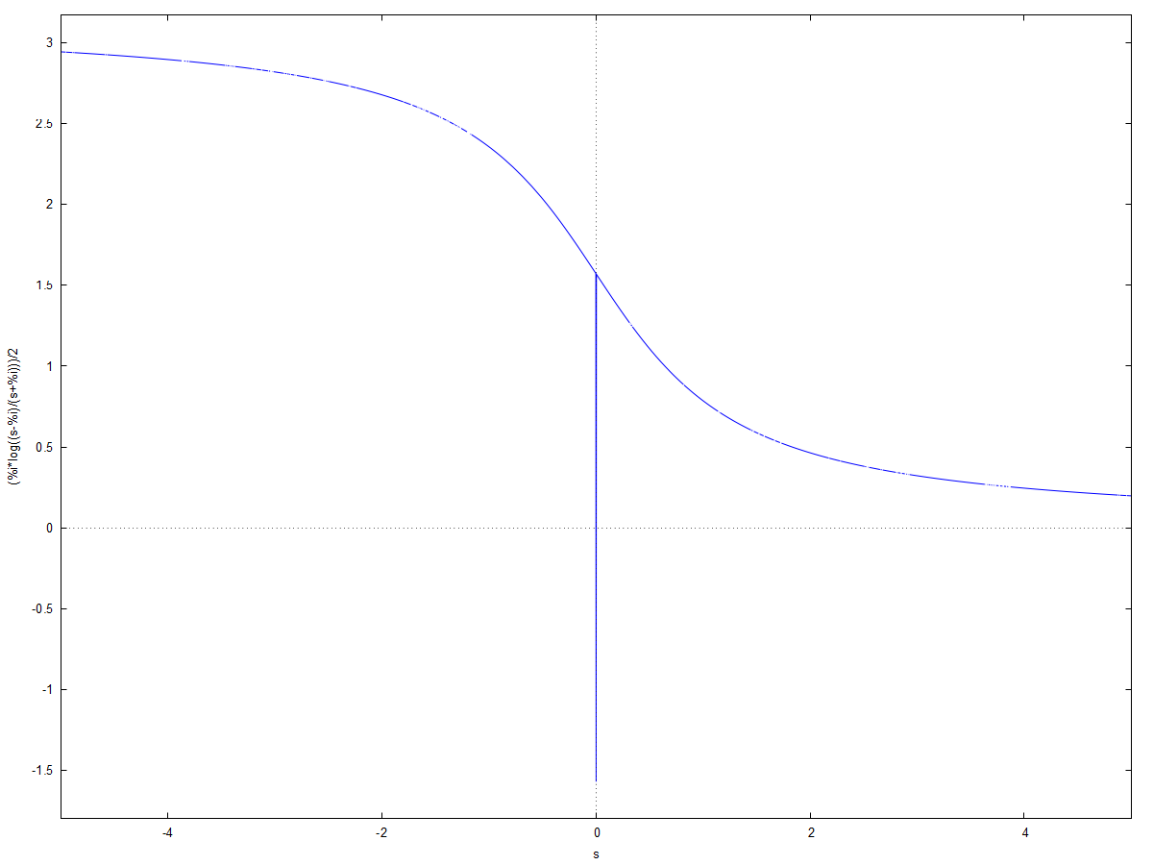


Laplace:

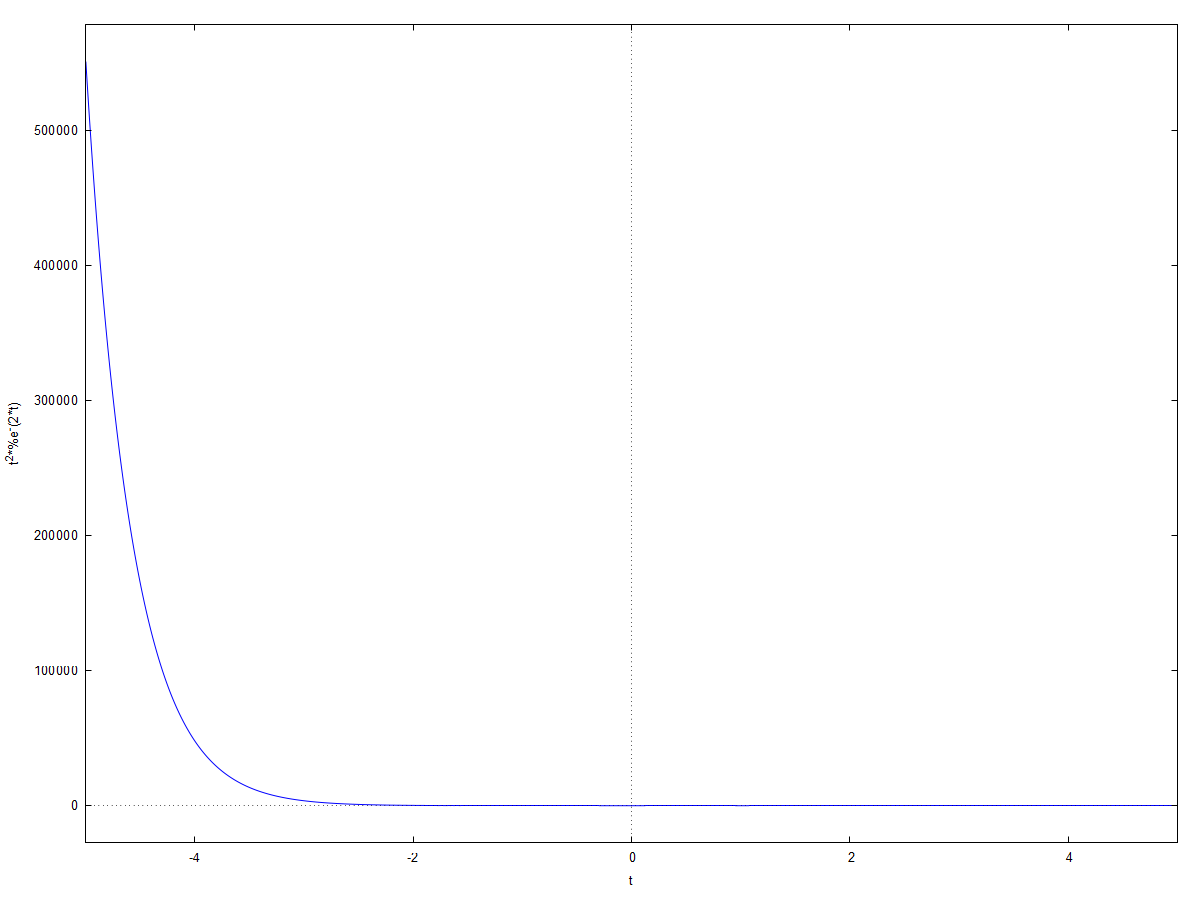


1. sint/t:  
   

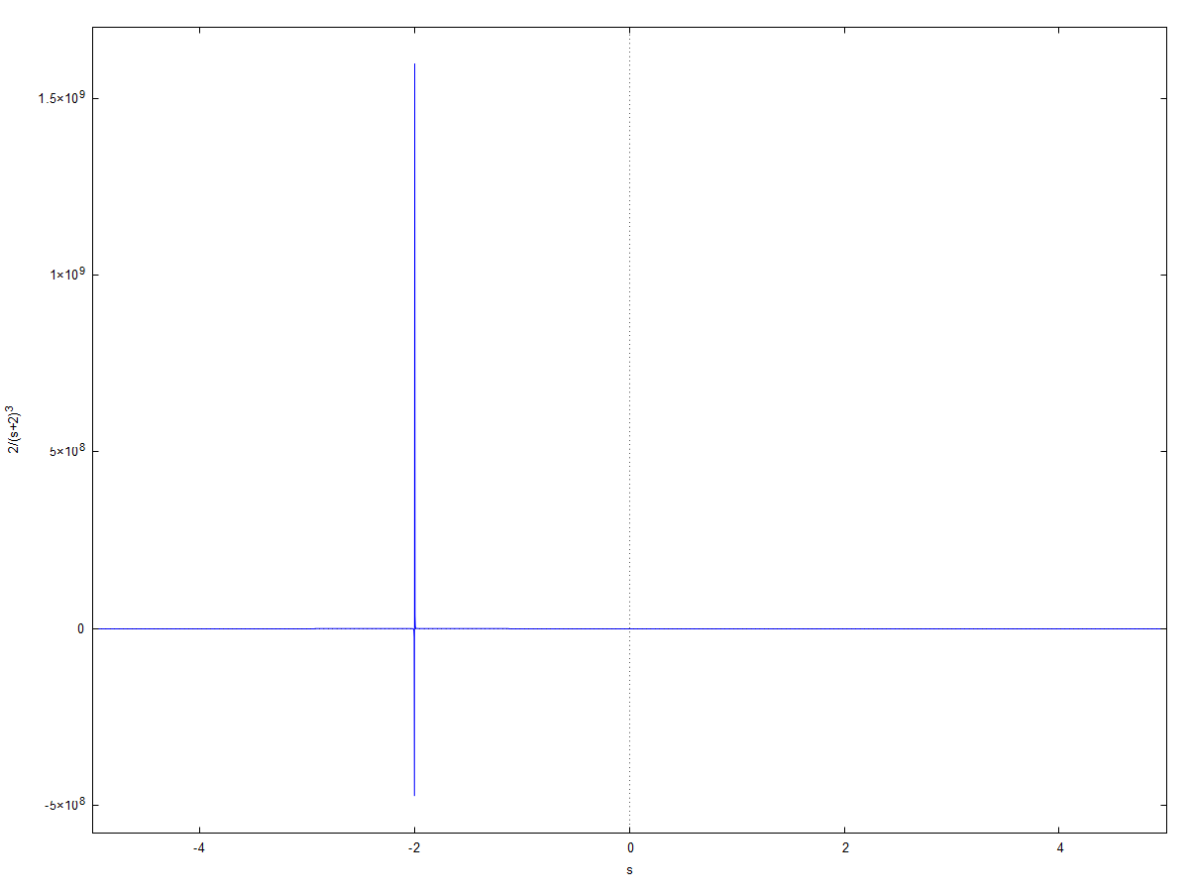
Laplace:



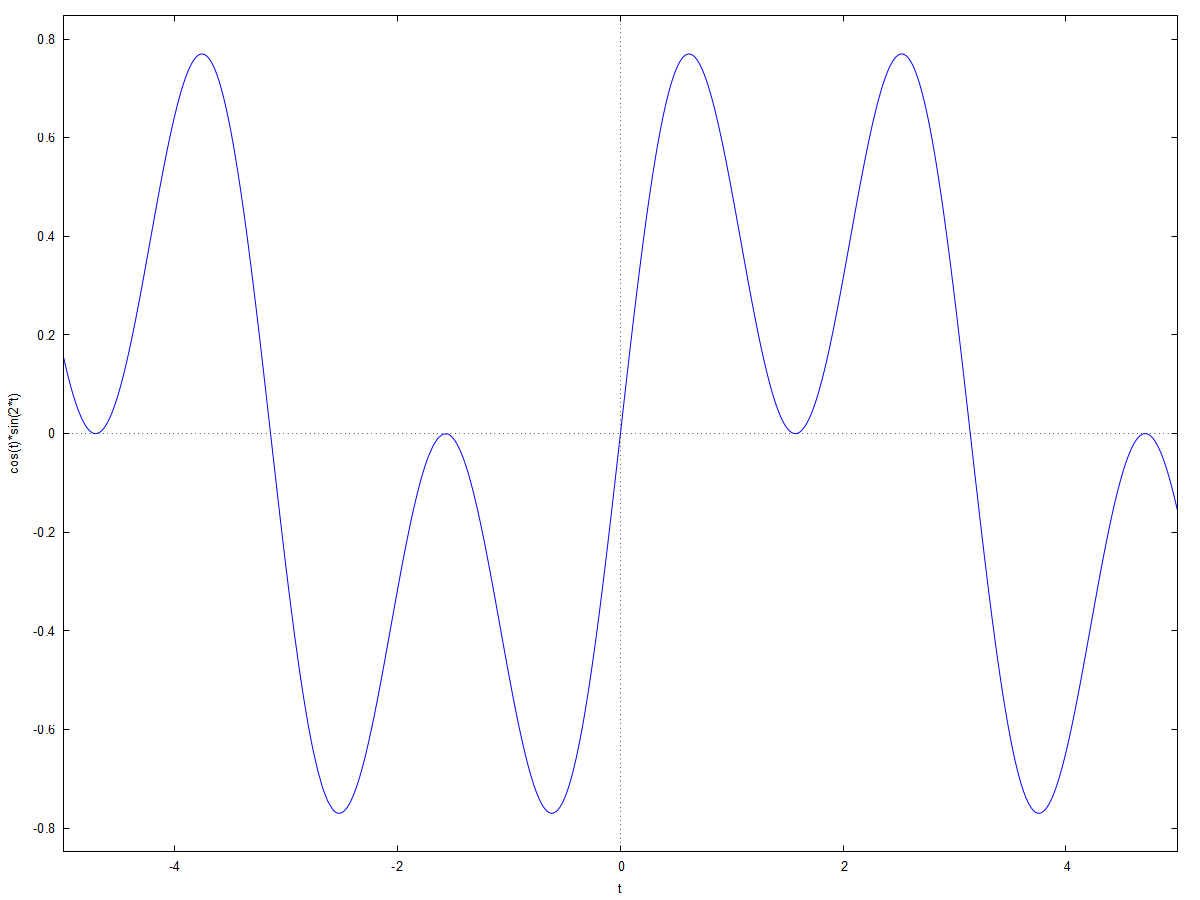
1. t^2 e^(-2)



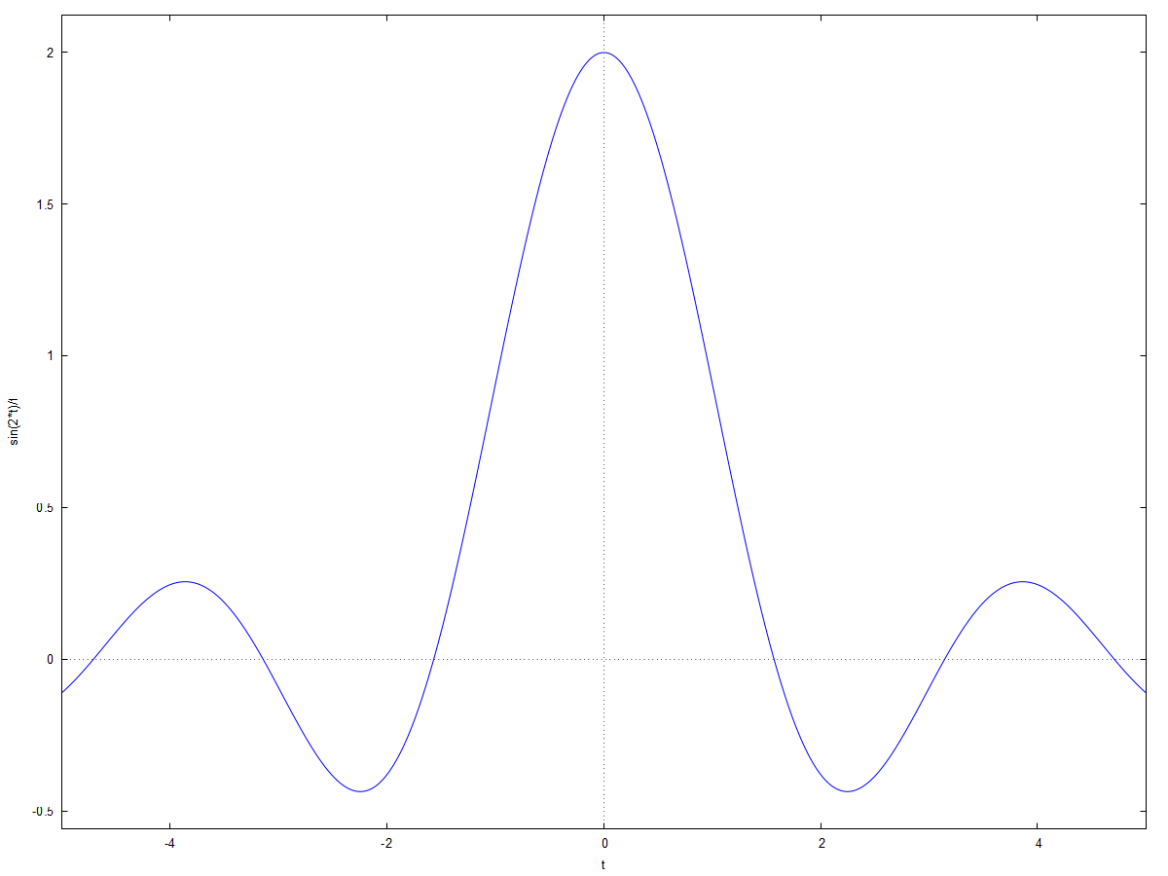
Laplace:



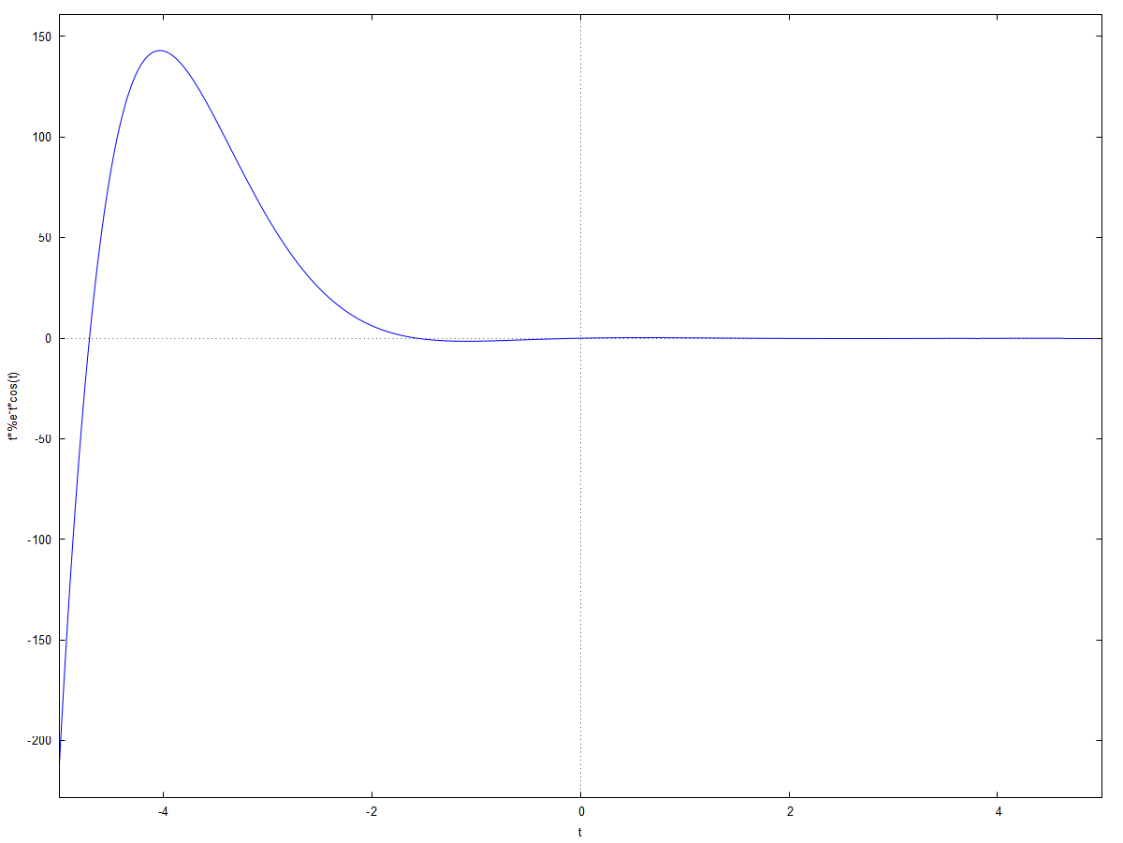
1. Cos t sin2t:



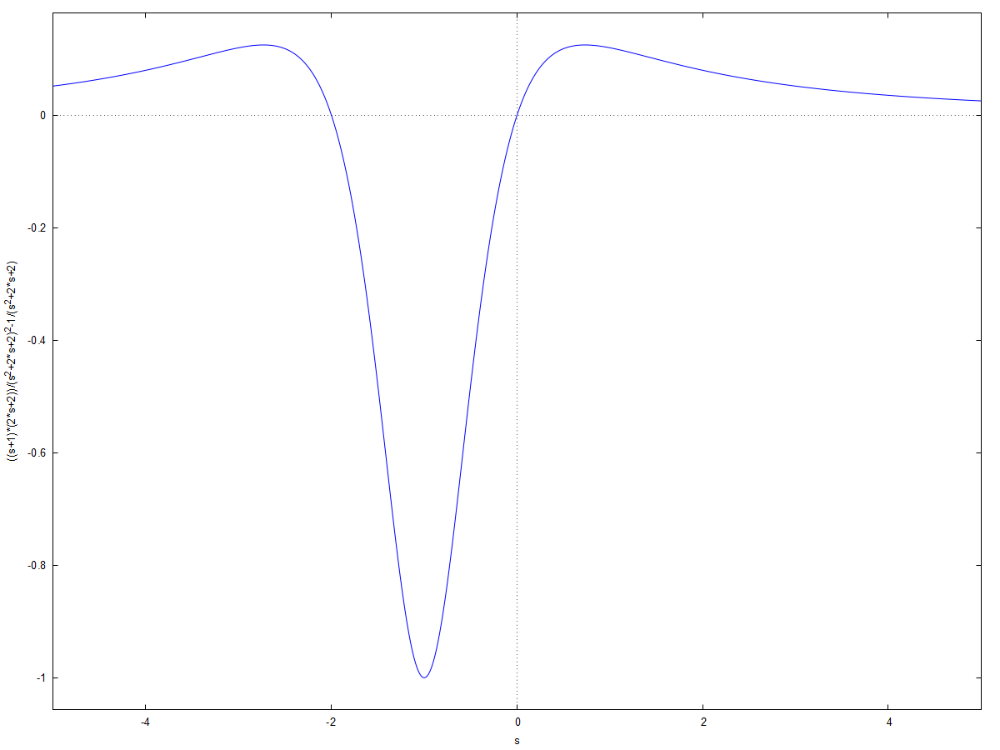
Laplace:



1. te^(-t) cost:



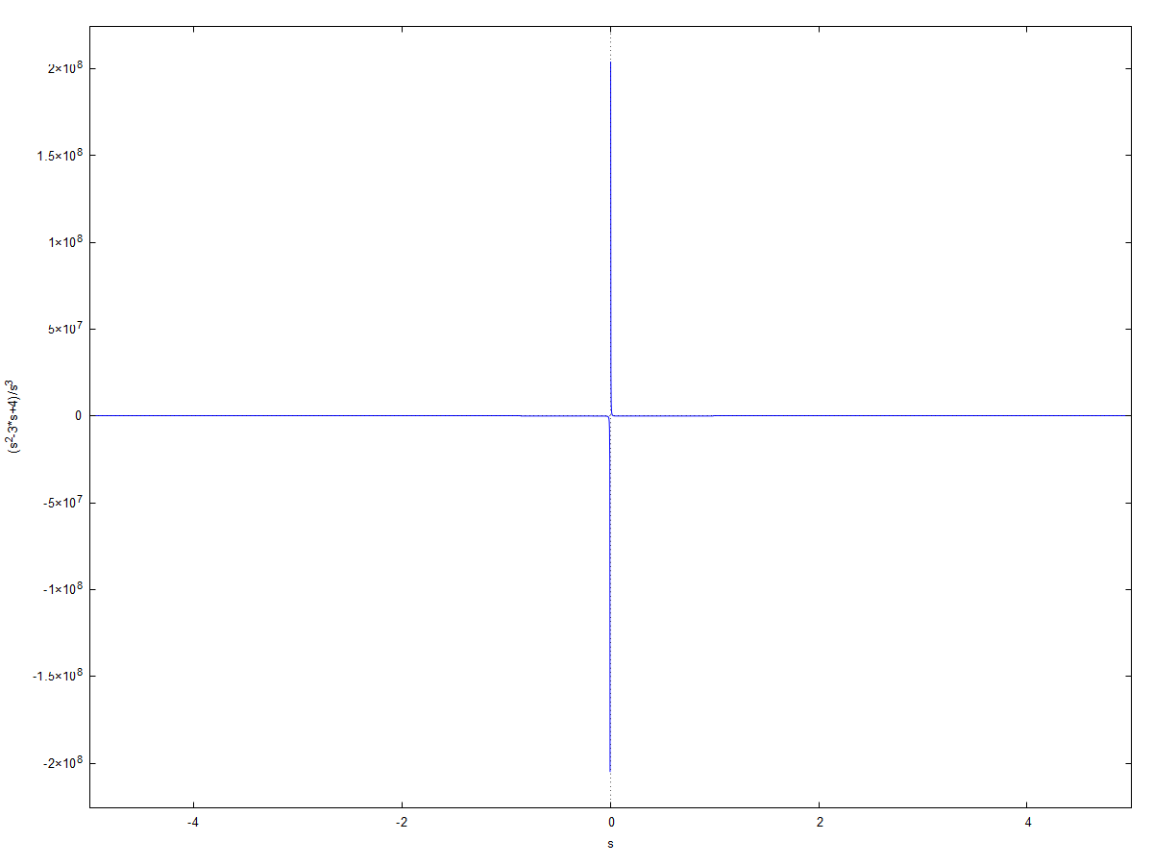
Laplace:



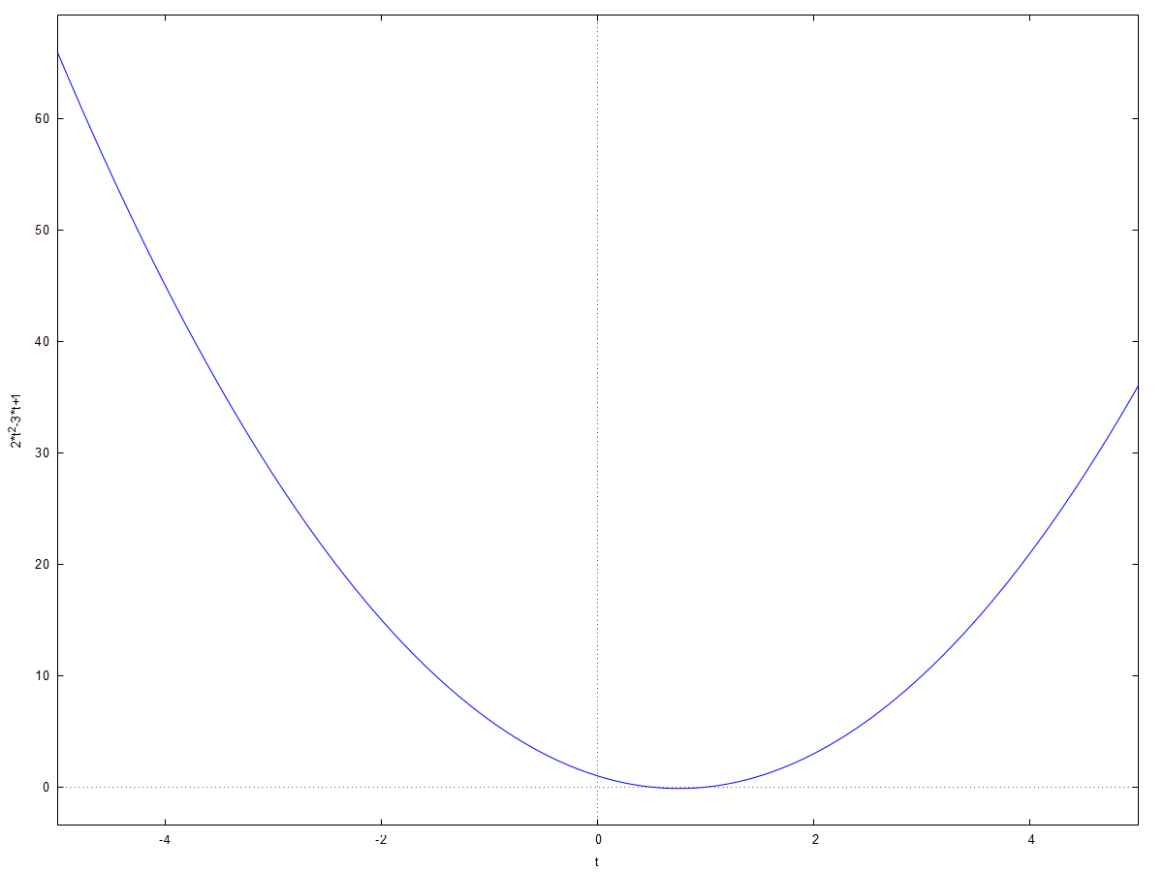
**Practical 2B**

**Inverse Laplace Transform**

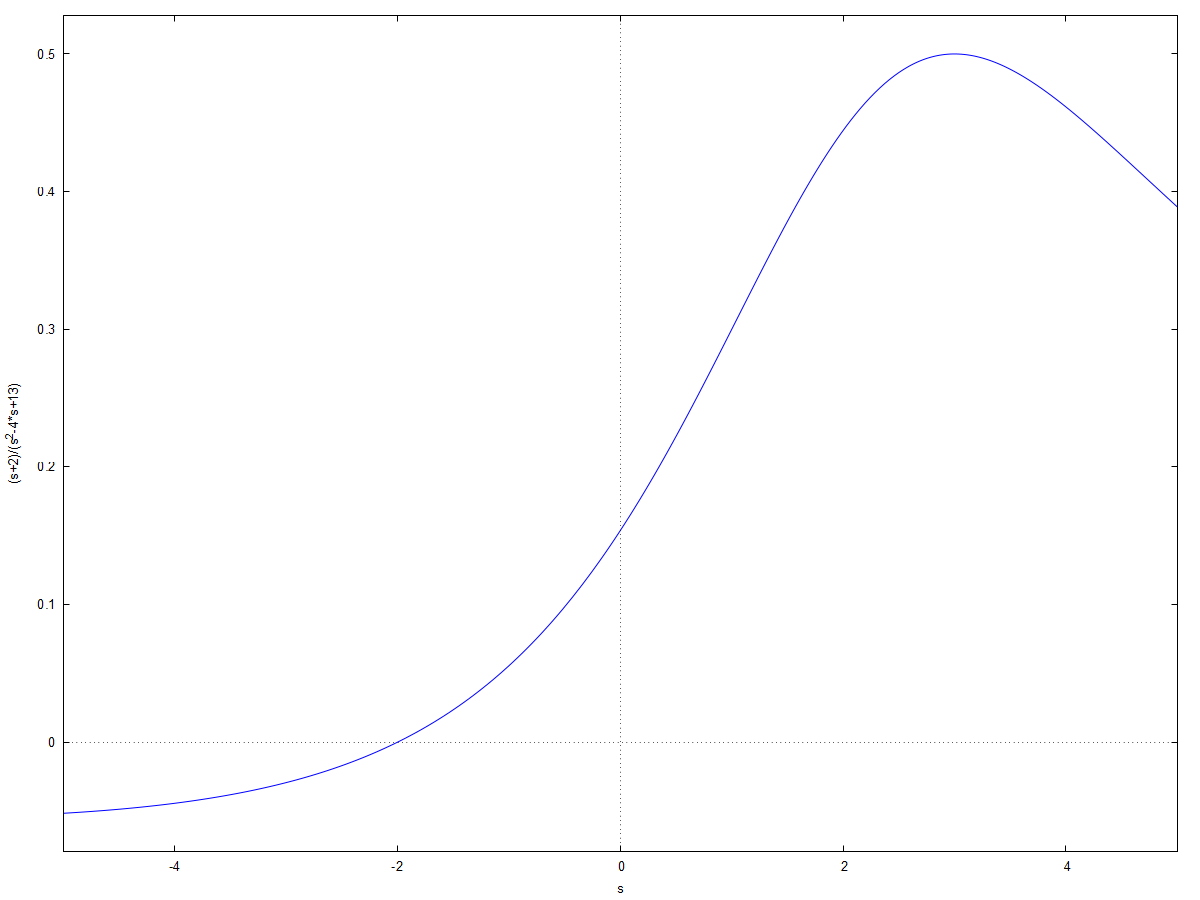
Q. (s^2-3\*s+4)/(s^3)



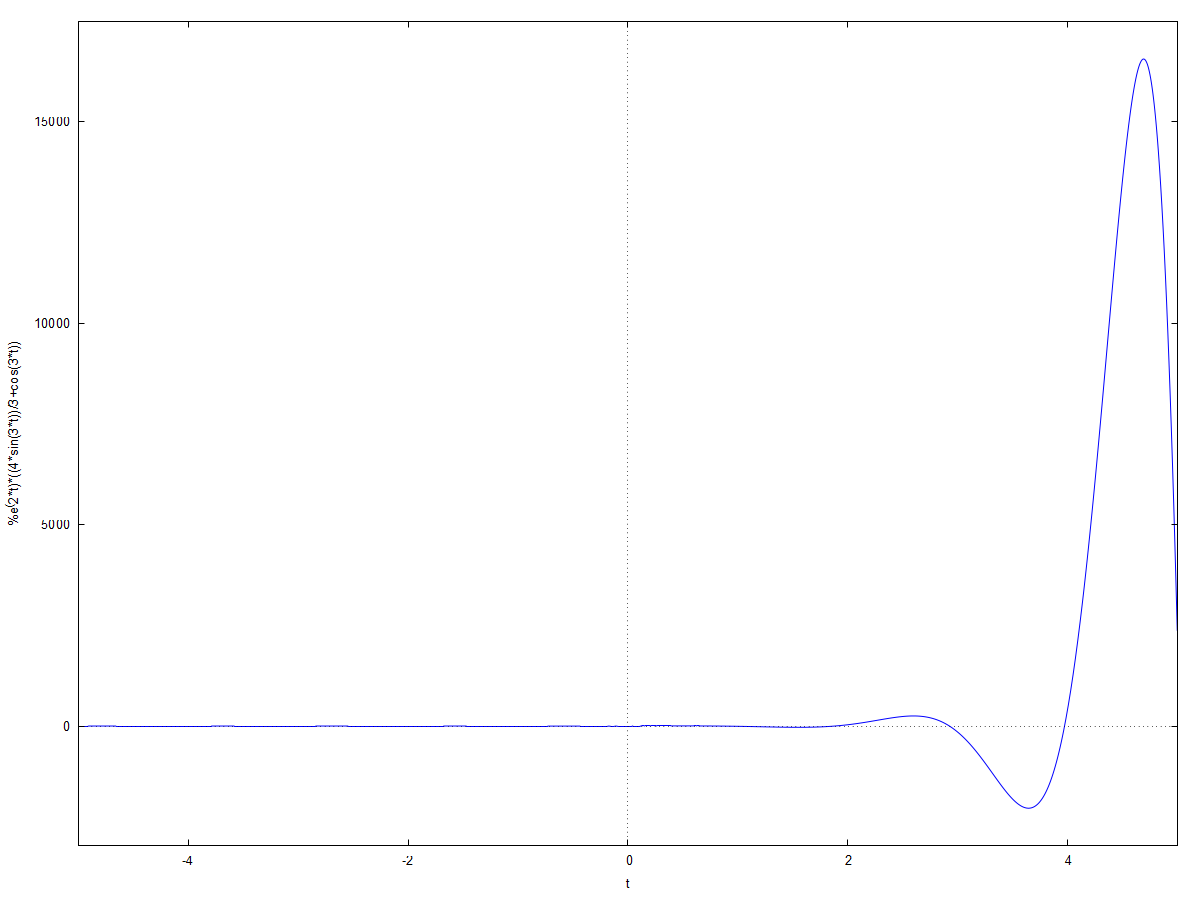
Inverse laplace:



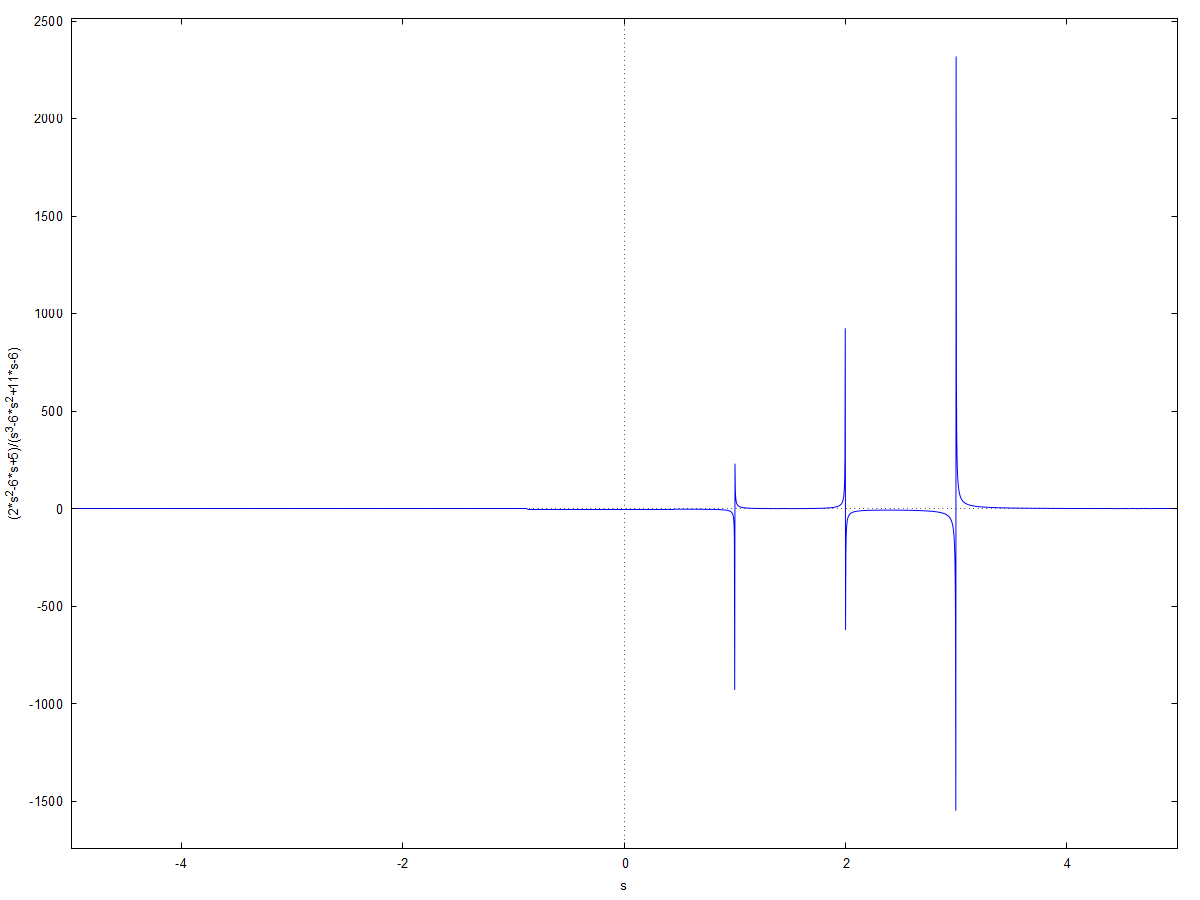
Q. (s+2)/(s^2-4\*s+13)



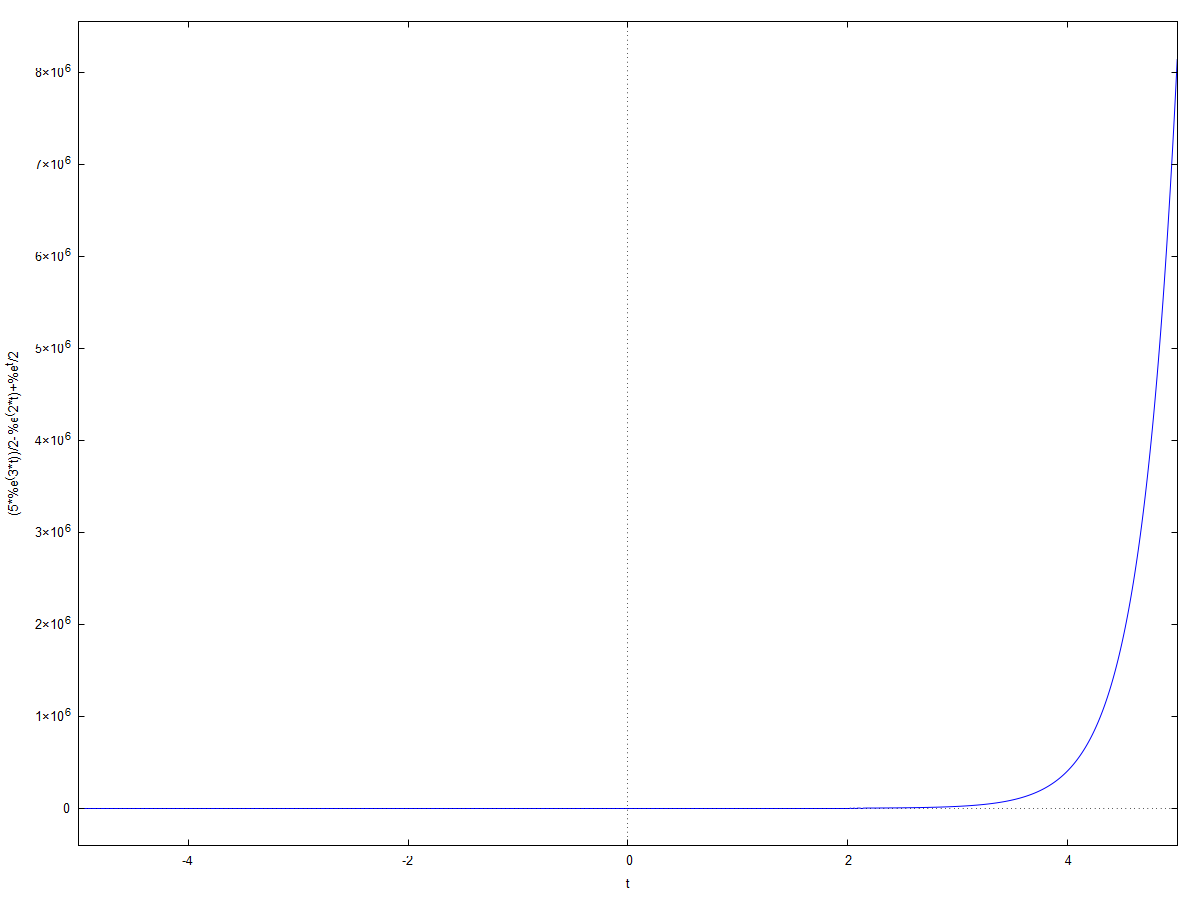
Inverse laplace:



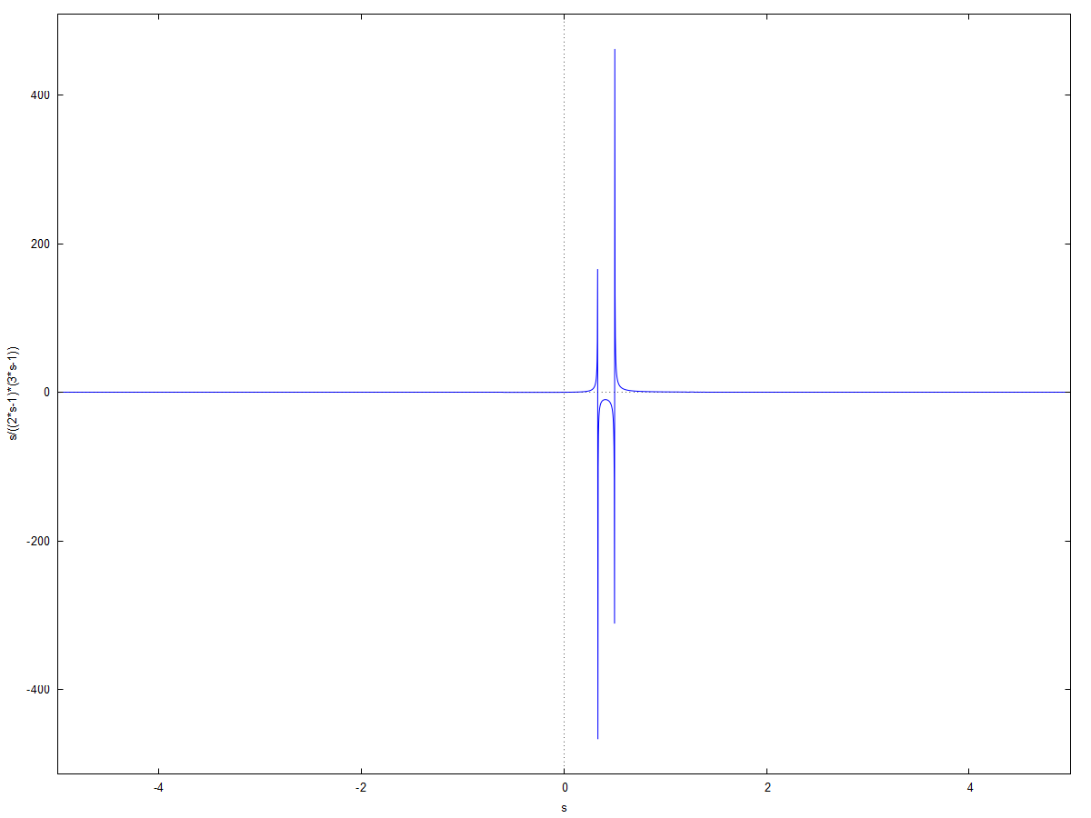
Q. (2\*s^2-6\*s+5)/(s^3-6\*s^2+11\*s-6)



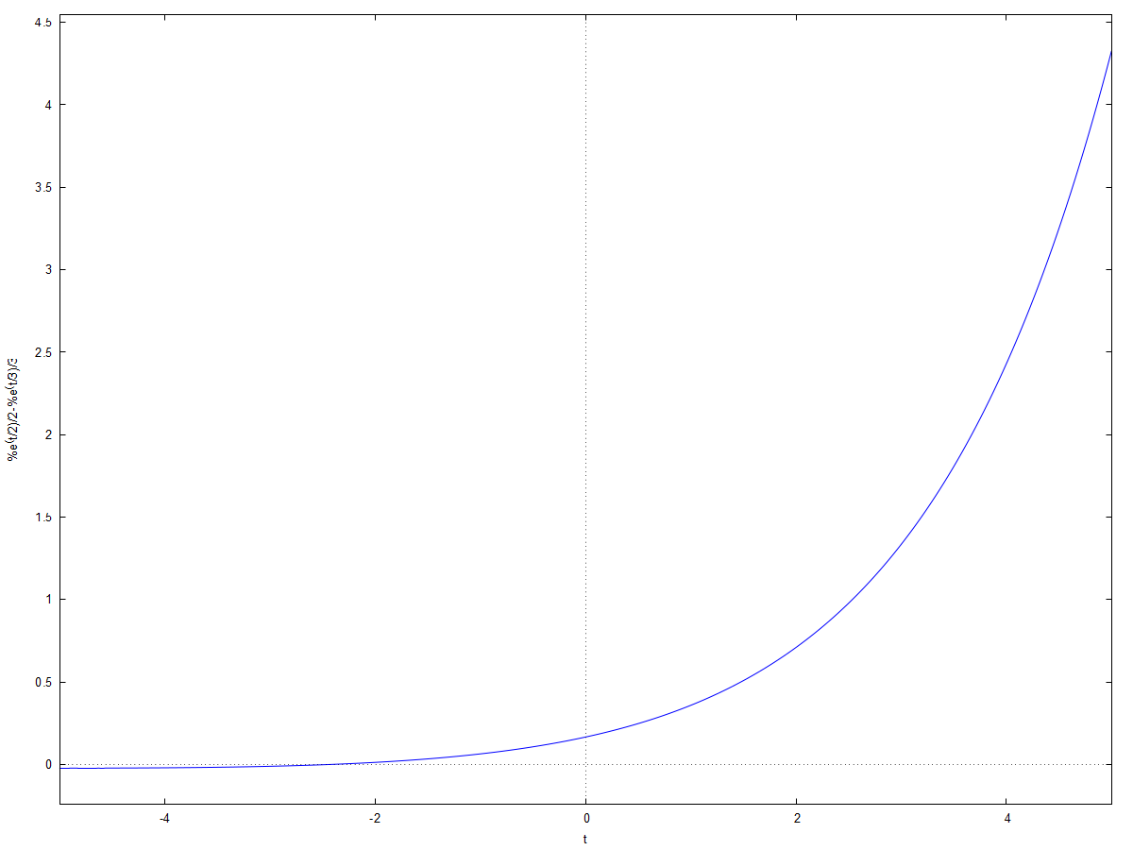
Inverse Laplace:



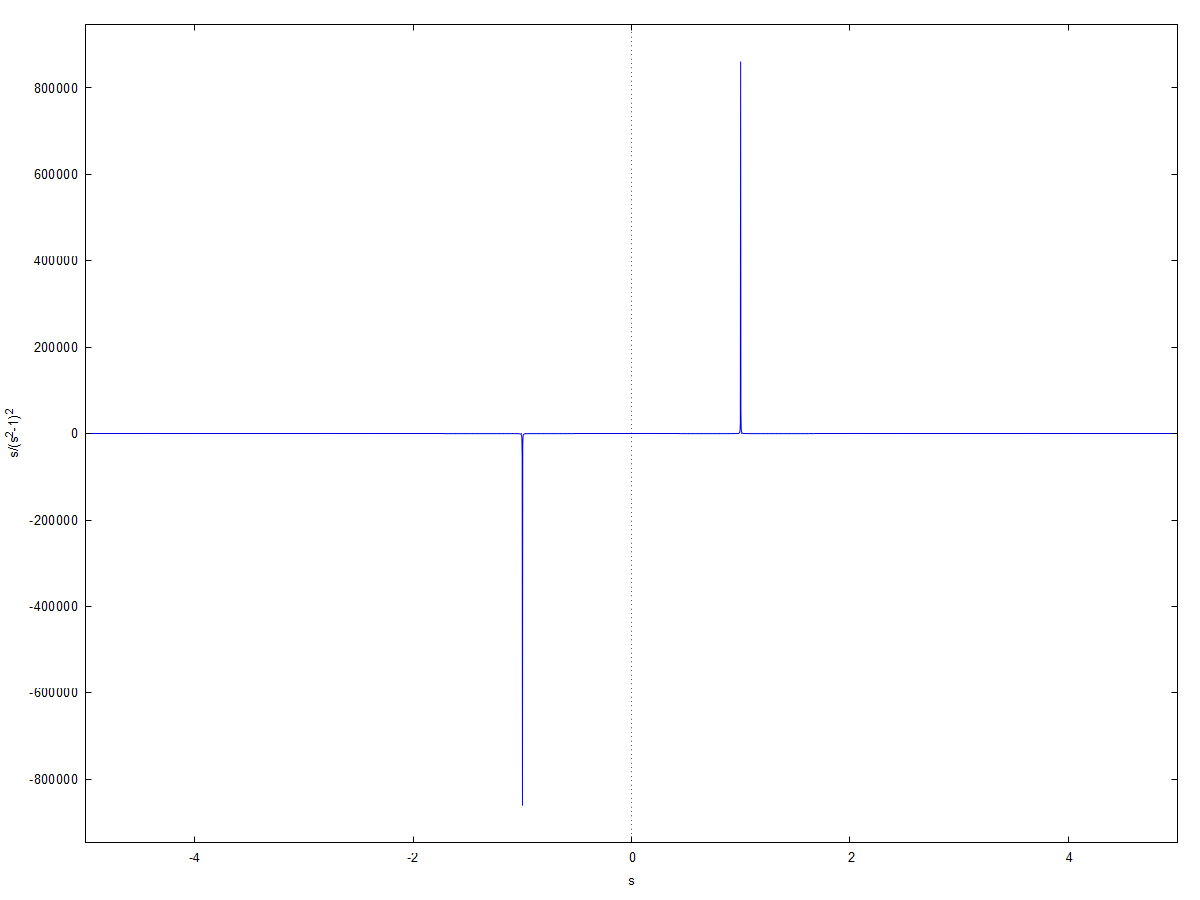
Q. (s)/((2\*s-1)\*(3\*s-1))



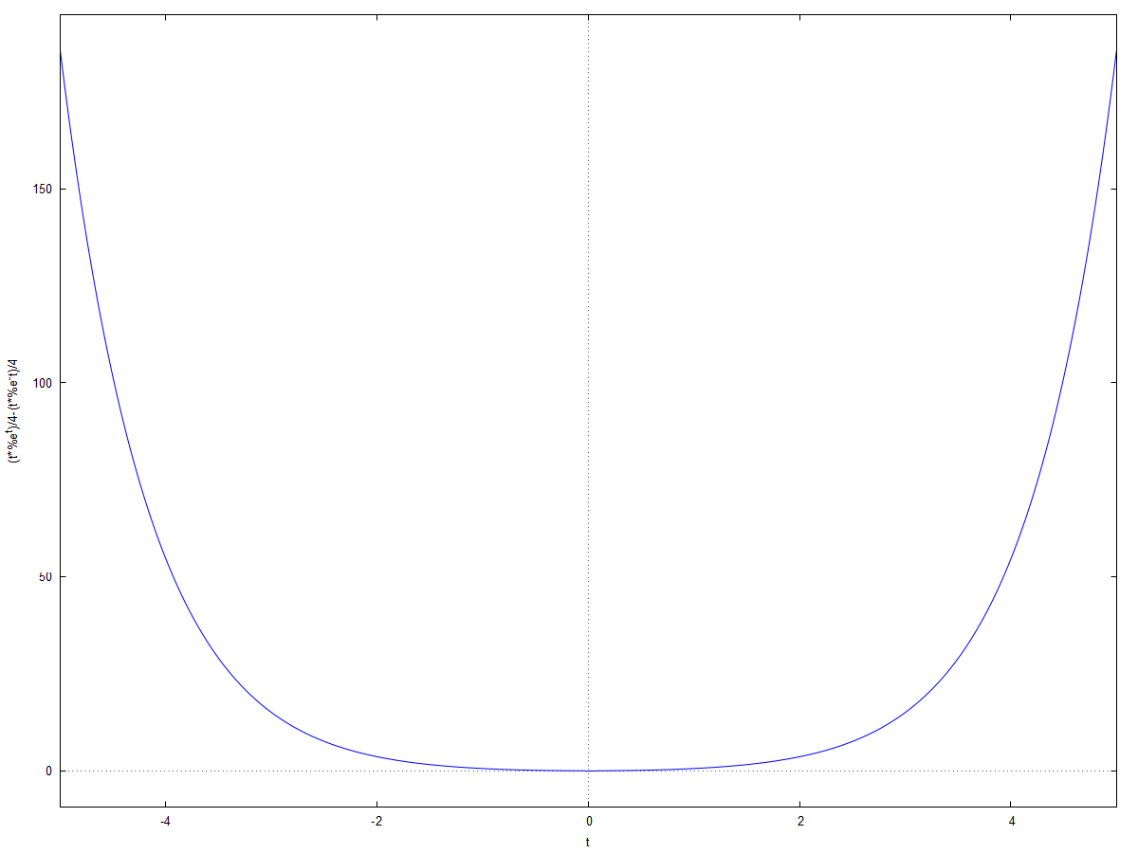
Inverse Laplace:



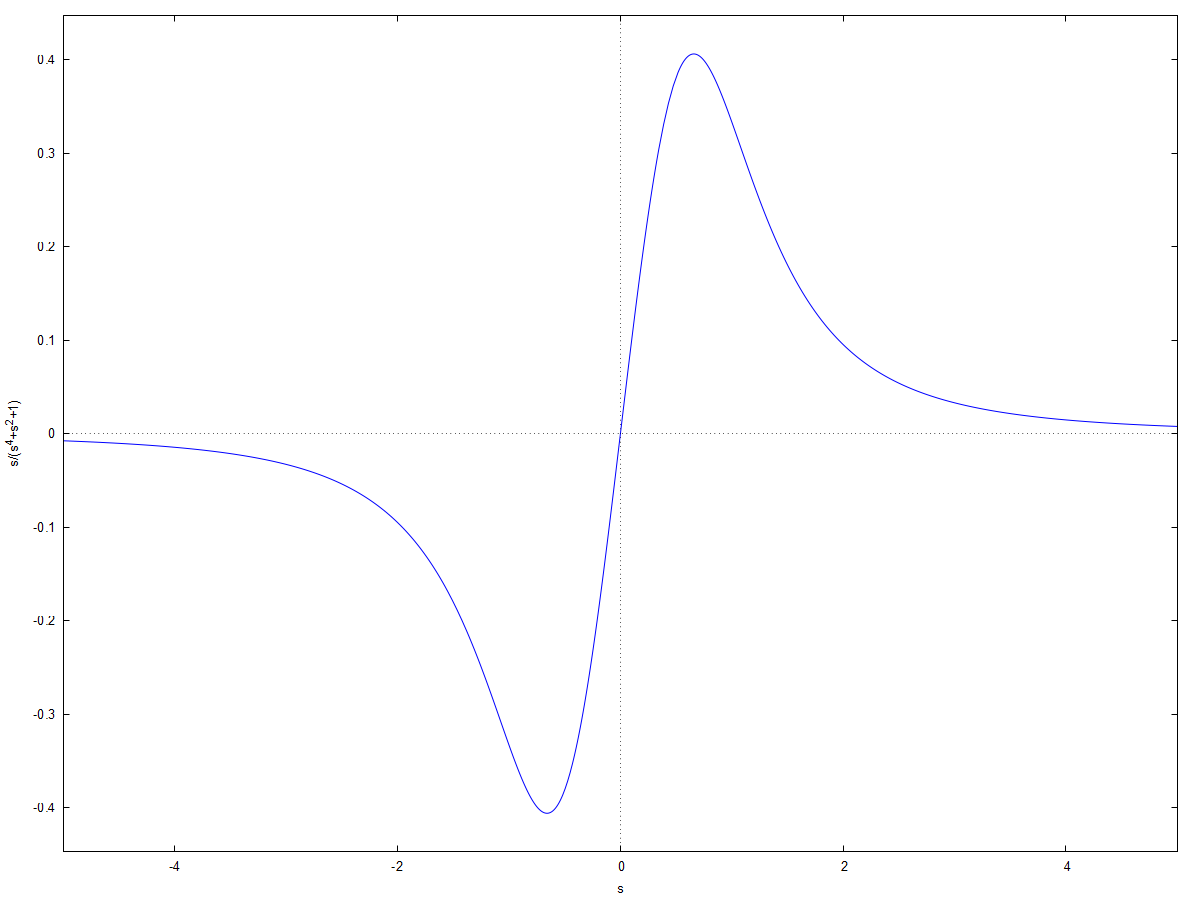
Q. (s)/(s^2-1)^2



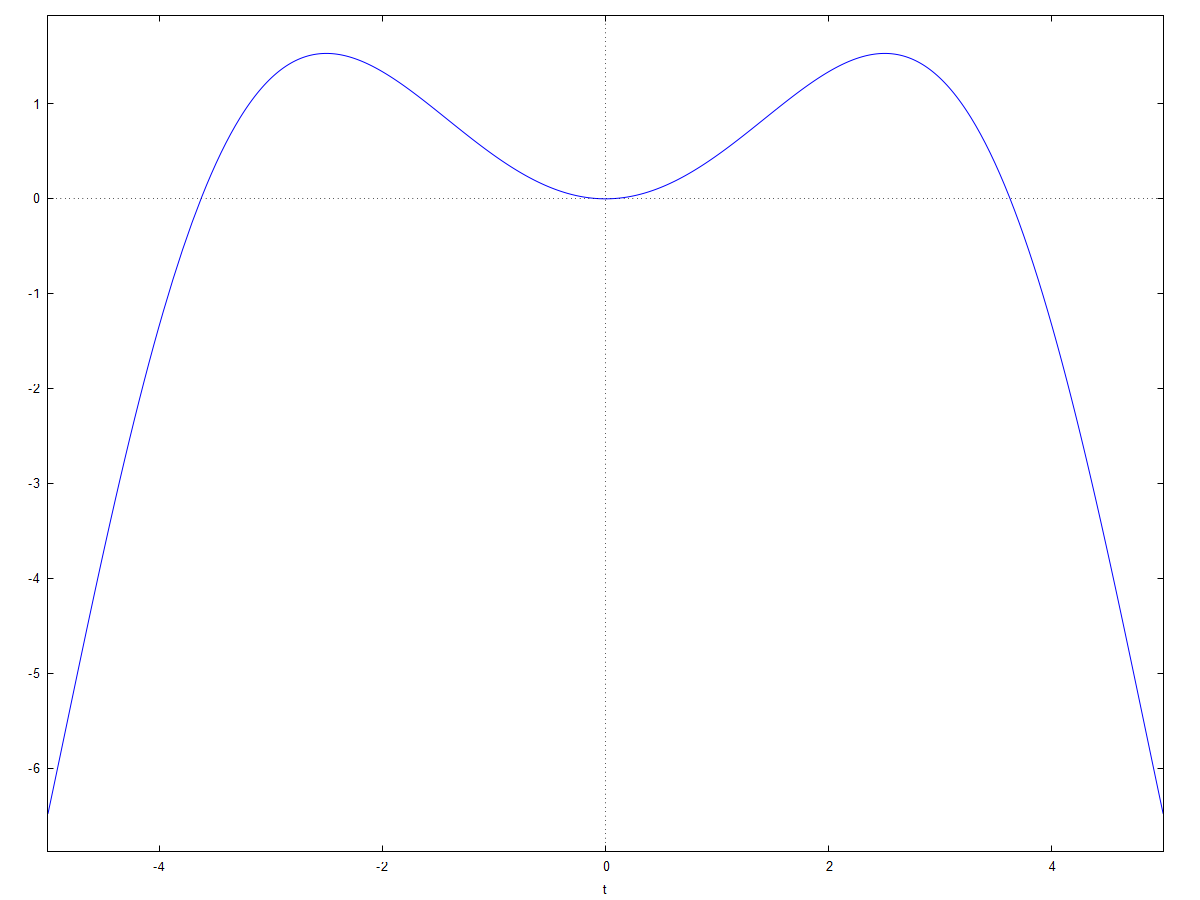
Inverse Laplace:



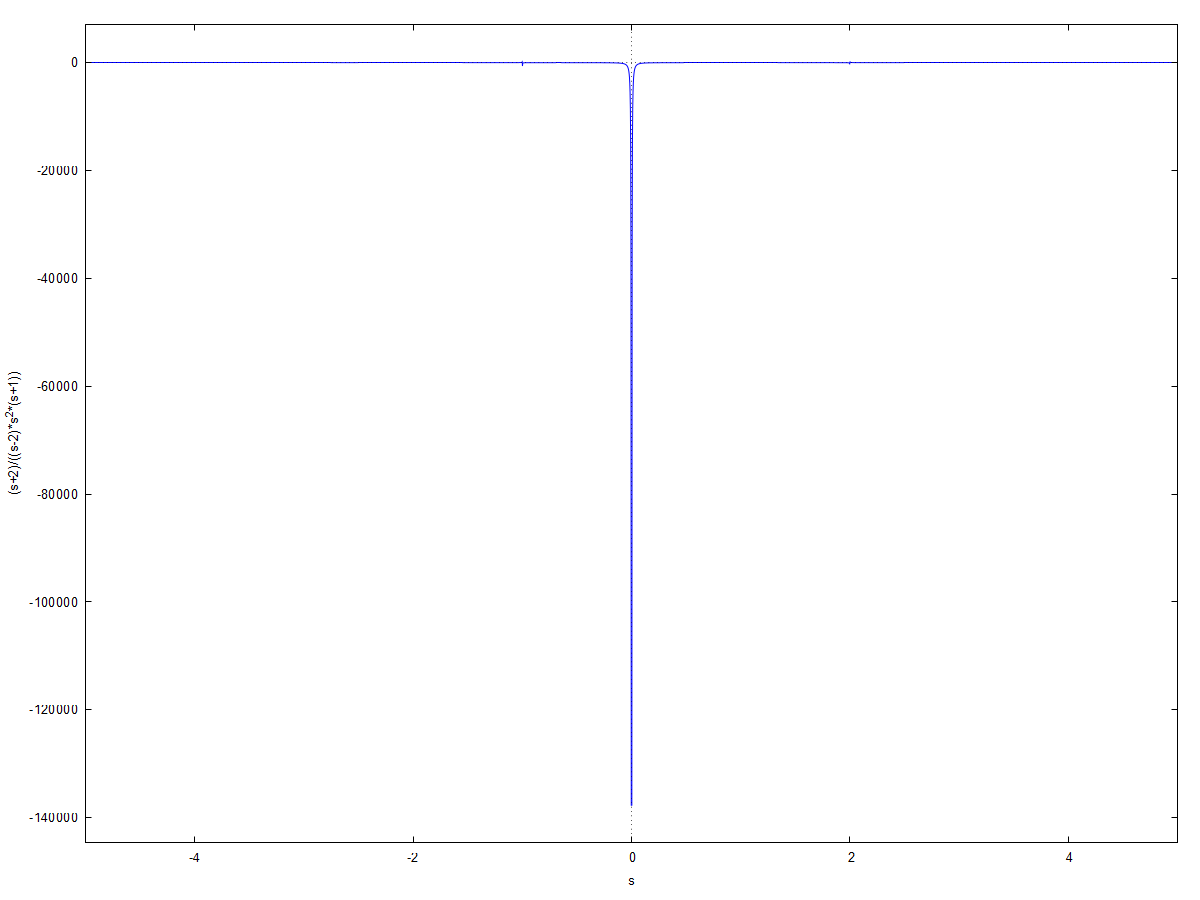
Q. (s)/(s^4+s^2+1)



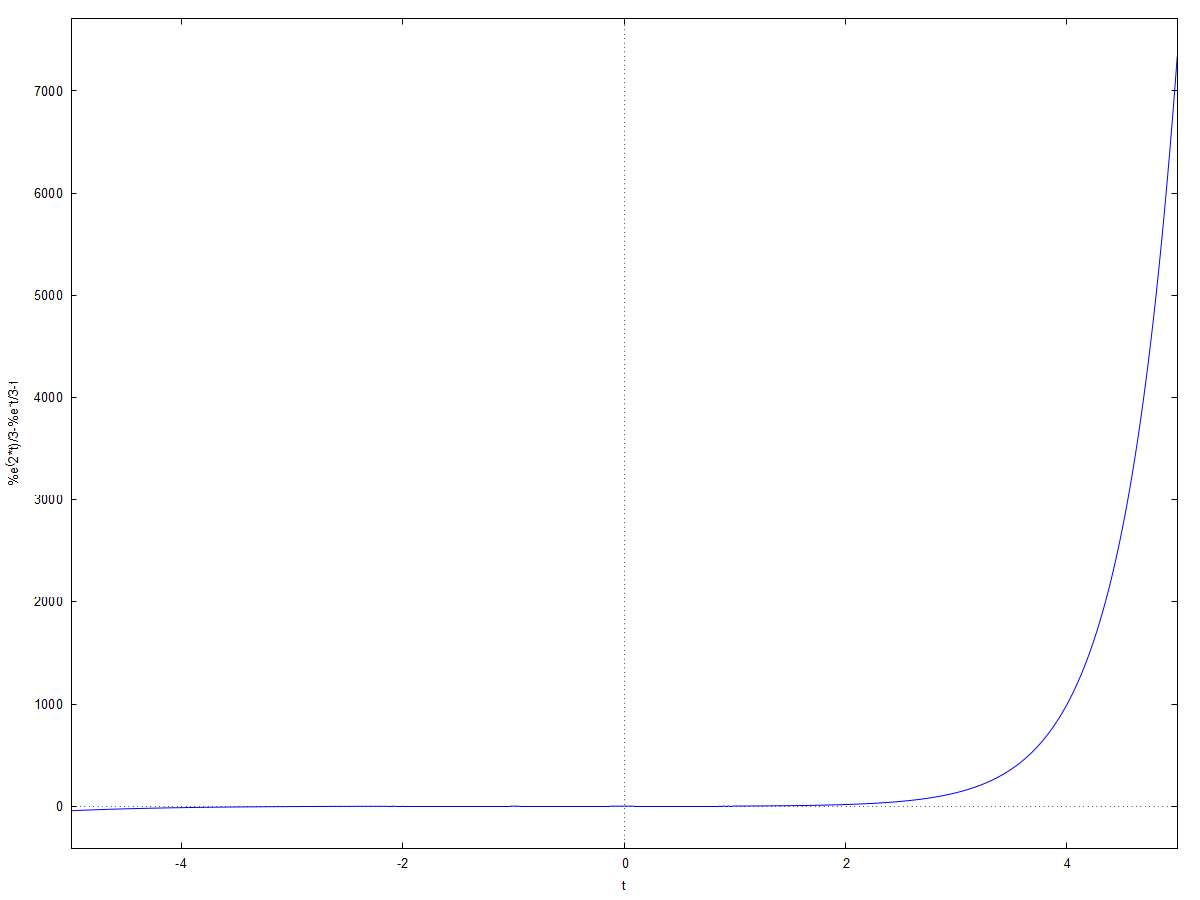
Inverse Laplace:



Q. (s+2)/(s^2\*(s+1)\*(s-2))



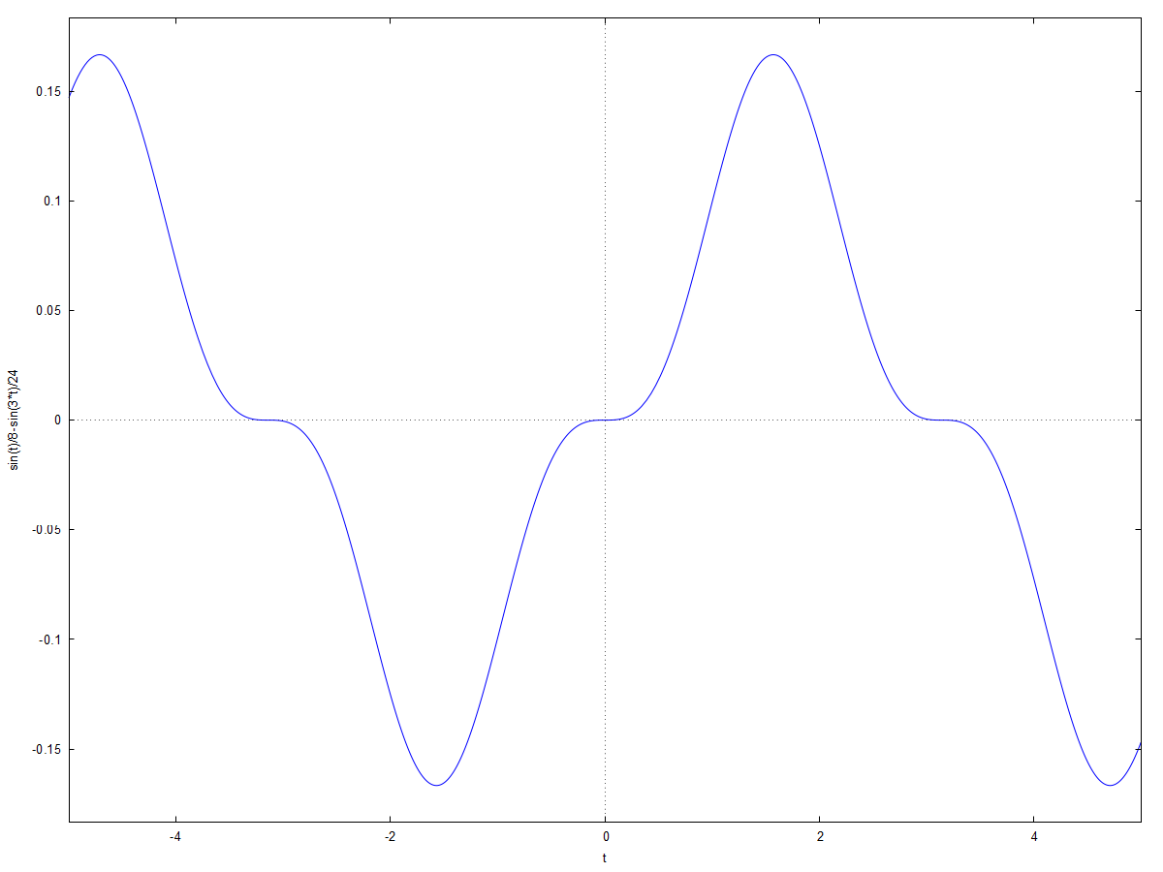
Inverse Laplace:



Q. 1/((s^2+1)\*(s^2+9))



Inverse Laplace:



**Practical 3**

**Differential Equation**

1]

x0=0; xinc=0.001; xf=1; x=x0:xinc:xf;

// Define x

// Calculate analytic solution

y=sqrt(x.^2+2\*x+0.01);

//Plot analytic solution

subplot(2,1,1), plot(x,y), xgrid

ylabel('y(x)', 'fontsize', 2)

title('Analtic solution','fontsize', 2)

// Define differential equation

deff('yprim=f(x,y)','yprim=(x+1)/y');

// Solve differential equation

y0-0.1;

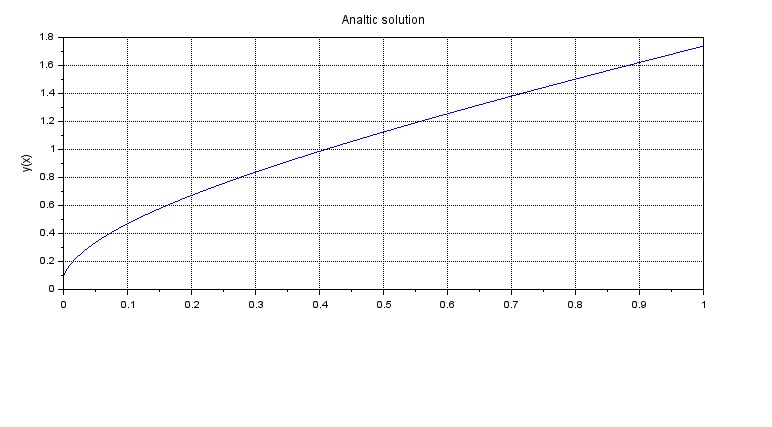
subplot(2,1,2), plot(x,ydiff,'r'), xgrid

title('Numeric solution', 'fontsize',2)

ylabel('y(x)', 'fontsize', 2)

ydiff=ode(y0,x0,x,f); // Plot numeric solution

xlabel('x','fontsize',2)



Ex 2

function ydot=f(t, y)

ydot=y^2-y\*sin(t)+cos(t)

endfunction

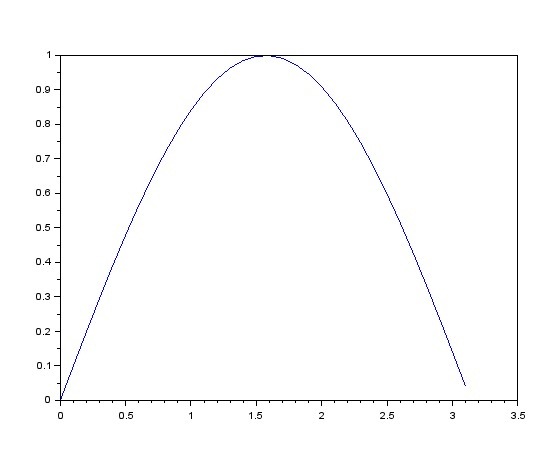
y0=0;

t0=0;

t=0:0.1:%pi;

y=ode(y0,t0,t,f);

plot(t,y)



Ex 3

function xdot=linear(t, x, A, u, B, omega)

xdot=A\*x+B\*u(t,omega)

endfunction

function ut=u(t, omega)

ut=sin(omega\*t)

endfunction

A= [1 1:0 2];

B=[1;1];

omega=5;

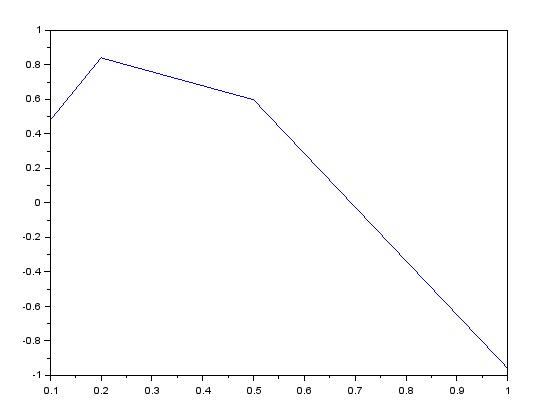
y0=[1;0];

t0=0;

t=[0.1,0.2,0.5,1];

ode(y0, t0,t,list(linear, A,u,B,omega))

plot(t,u);



Ex 4

function y=u(t)

y=(sign(t)+1)/2

endfunction

L=0.001

R=10

C=0.000001

function zdot=f(t,y)

zdot(1)=y(2);

zdot(2)=(u(t)-y(1)-L\*y(2)/R)/(L\*C);

endfunction

y0=[0;0];

t0=0;

t=0:0.00001:0.001;

out=ode(y0,t0,t,f);

clf();

plot(out);

function y=u(t)

y=(sign(t)+1)/2

endfunction

L=0.001

R=10

C=0.000001

function zdot=f(t,y)

zdot=[y(2);(u(t)-y(1)-L\*y(2)/R)/(L\*C)];

endfunction

y0=[0;0];

t0=0;

t=0:0.00001:0.001;

out=ode(y0,t0,t,f);

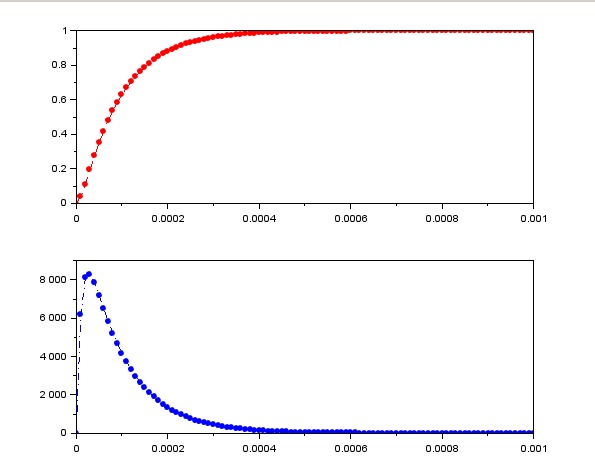
clf();

subplot(211)

plot(t,out(1,:),"r.--");

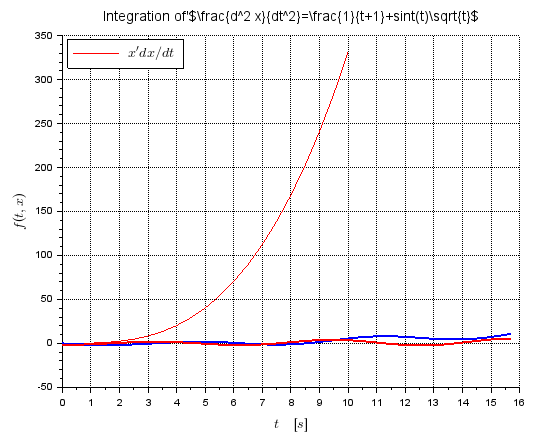
subplot(212)

plot(t,out(2,:),"b-..");



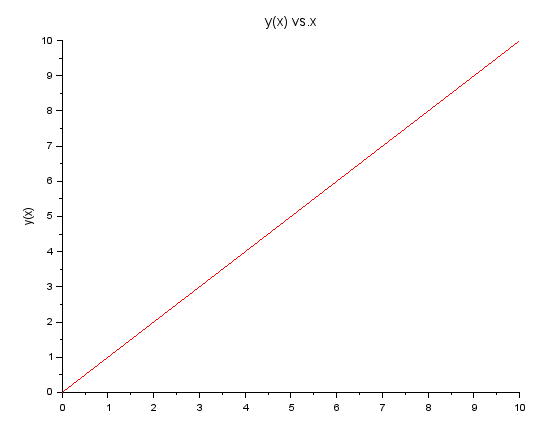
Ex 5

function **dx**=f(**t**, **x**)  
 **dx**(1)=**x**(2);  
 **dx**(2)=1/(**t**+1)+sin(**t**)\*sqrt(**t**);  
endfunction  
t=0:0.01:5\*%pi;  
t0=min(t);  
y0=[0;-2];  
y=ode(y0,t0,t,f);  
plot(t,y(1,:),'LineWidth',2)  
plot(t,y(2,:),'r','LineWidth',2)  
xgrid();  
xlabel('$t\quad[s]$','FontSize',3)  
ylabel('$f(t,x)$','FontSize',3)  
title(['Integration of''$\frac{d^2 x}{dt^2}=\frac{1}{t+1}+sint(t)\sqrt{t}$'],'FontSize',3)  
legend(['$\Large{x}$''$\Large{dx/dt}$'],2)



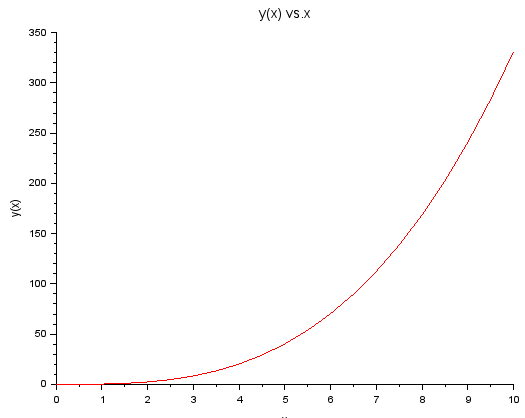
Ex 6

funcprot(0)  
clf;  
function **dx**=f(**x**, **y**)  
 **dx**=exp(-x0);  
endfunction  
y0=0;  
x0=0;  
x=[0:0.5:10];  
sol=ode(y0,x0,x,f);  
plot2d(x,sol,5)  
xlabel('x');  
ylabel('y(x)');  
xtitle('y(x) vs.x')



Ex 7

funcprot(0)  
clf;  
function **dx**=f(**x**, **y**)  
 **dx**=**x**^2-exp(-**x**)\***y**;  
endfunction  
y0=0;  
x0=0;  
x=[0:0.5:10];  
sol=ode(y0,x0,x,f);  
plot2d(x,sol,5)  
xlabel('x');  
ylabel('y(x)');  
xtitle('y(x) vs.x')



**Practical 4A**

A]

clear;

*//Fourier Transform of x(t)=exp(-a\*t)\*cos(wc\*t)\*u(t)*

clear;

clc;

a=1;

wc=1;

Dt=0.005;

t=0:Dt:10;

xt=(exp(t\*(-a+wc))+exp(t\*(-a-wc)))/2;

Wmax=2\*%pi\*1;

K=4;

k=0:(K/1000):K

W=k\*Wmax/K;

XW=xt\*exp(-sqrt(-1)\*t'\*W)\*Dt;

XW\_Mag=abs(XW);

[XW\_Phase,db]=phasemag(XW);

*//Plotting Continuos Time Signal*

figure(1)

plot(t,xt);

xlabel('t in sec');

ylabel('x(t)')

title('Continuous Time Signal')

figure(2)

*//Plotting Magnitude Response of CTS*

subplot(2,1,1)

plot(W,XW\_Mag);

xlabel('Frequency in Radians/Seconds>W');

ylabel('abs(X(jW))')

title('Magnitude Response(CTFT)')

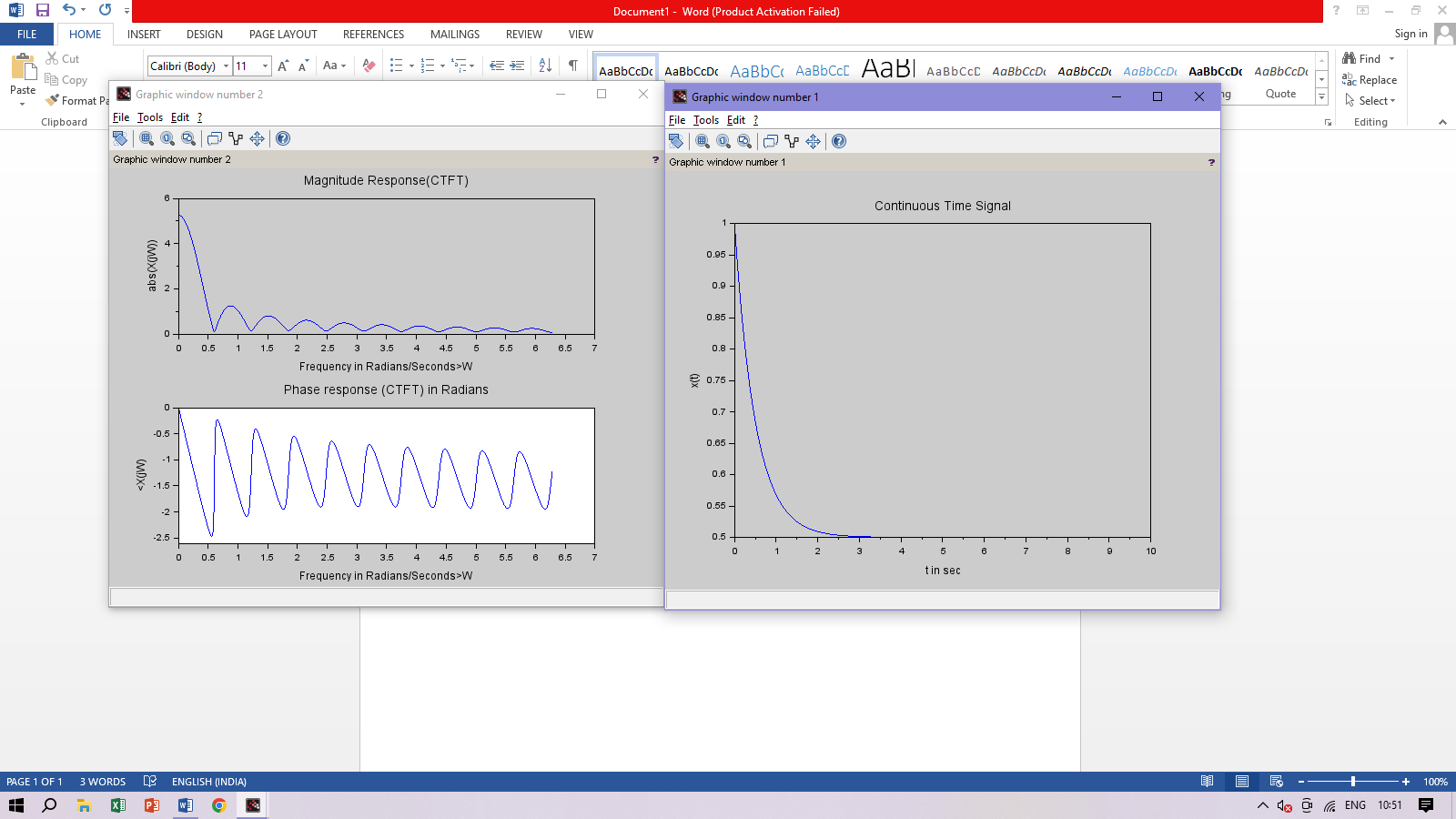
*//Plotting Phase Response of CTS*

subplot(2,1,2);

plot(W,XW\_Phase\*%pi/180);

xlabel('Frequency in Radians/Seconds>W');

ylabel('<X(jW)')

title('Phase response (CTFT) in Radians')  


B]

*//Continuous Time Fourier Transforms of*

*// Sinusoidal waveforms (a)sin(Wot) (b)cos(Wot)*

clear;

clc;

close;

*//CTFT*

T1 = 2;

T=4\*T1;

Wo=2\*%pi/T;

W=[-Wo,0,Wo];

ak =(2\*%pi\*Wo\*T1/%pi)/sqrt(-1)

XW=[-ak,0,ak]

ak1=(2\*%pi\* Wo\*T1/%pi);

XW1=[-ak1,0,ak1]

figure;

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d3('gnn',W,imag(XW),2);

poly1=a.children(1).children(1);

poly1.thickness=3;

xlabel('')

title('CTFT of sin(Wot)')

figure

a=gca();

a.y\_location="origin";

a.x\_location="origin";

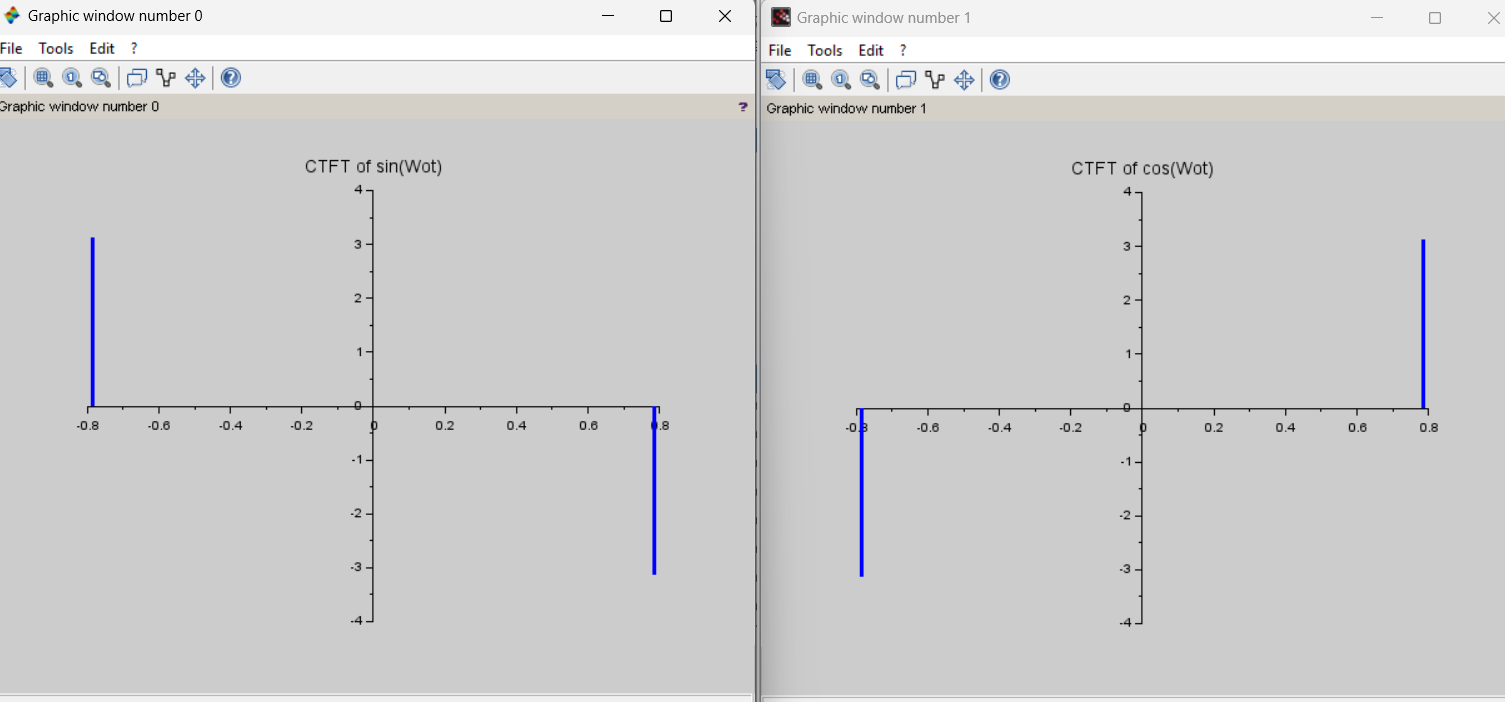
plot2d3('gnn', W,XW1,2);

poly1=a.children(1).children(1);

poly1.thickness=3;

xlabel('')

title('CTFT of cos(Wot)')



C] clear;

*//Fourier Transform of x(t) exp(-t)\*sin(wc\*t)\*u(t)*

clear;

clc;

wc=1;

Dt =0.005;

t=0:Dt:10;

xt=(exp(t\*(-1+ wc))-exp(t\*(-1-wc)))/(2\*%i);

Wmax=2\*%pi\*1;

K =4;

k = 0:(K / 1000):K;

W=k\*Wmax/k

XW=xt\*exp(-sqrt(-1)\*t'\*W)\*Dt;

XW\_Mag=abs(XW);

[XW\_Phase,db]=phasemag(XW);

*//Plotting Continuous Time Signal*

figure(1)

plot(t,xt);

xlabel('t in sec.');

ylabel('x(t)');

title('Continuous Time Signal')

figure(2);

*//Plotting Magnitude Response of CTS*

subplot(2,1,1);

plot(W,XW\_Mag);

xlabel('Frequency in Radians/Seconds>W')

ylabel('abs(X(jW))')

title('Magnitude Response (CTFT)')

*//Plotting Phase Reponse of CTS*

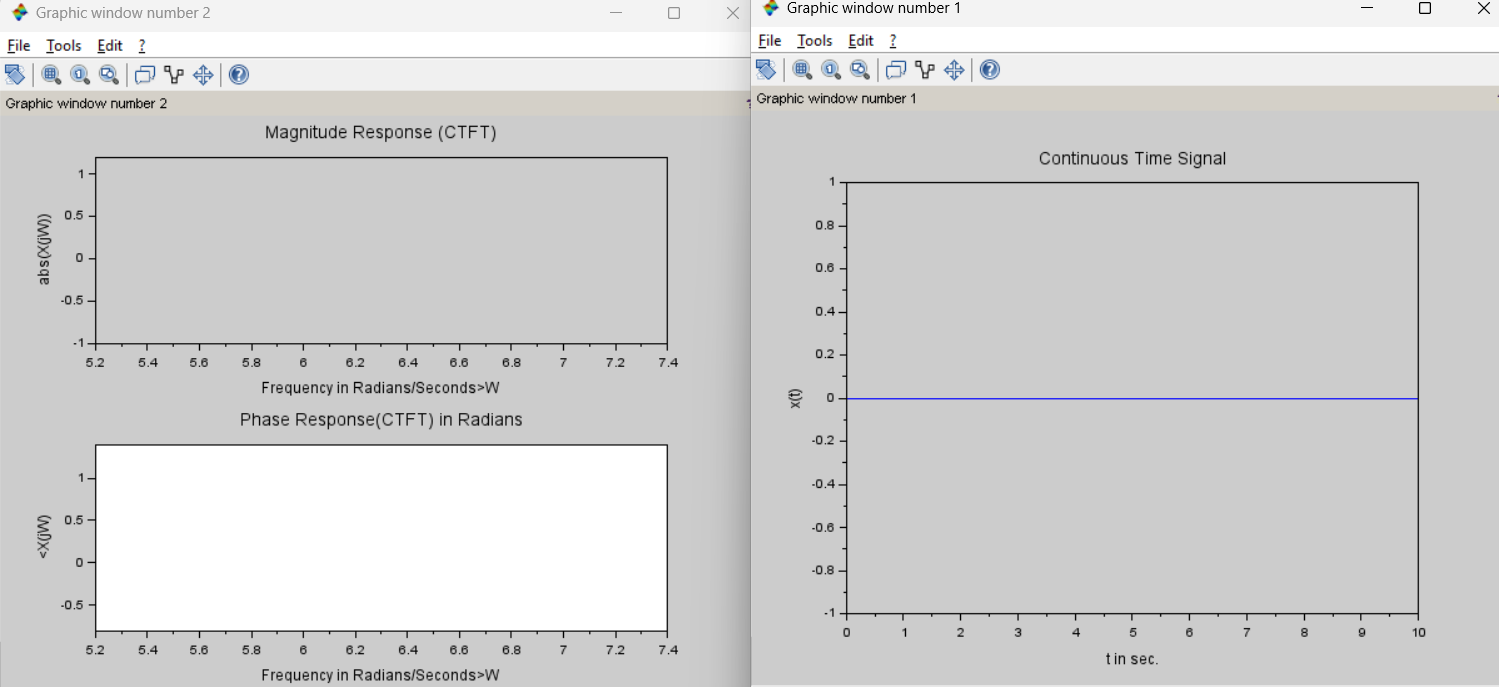
subplot(2,1,2);

plot(W,XW\_Phase\*%pi/180);

xlabel('Frequency in Radians/Seconds>W');

ylabel('<X(jW)')

title('Phase Response(CTFT) in Radians')



**Practical 4B**

***//Inverse Continuous Time Fourier Transform***

*//X(jW)=1, from -T1 to T1*

clear;

clc;

close;

*//CTFT*

A=1; *//Amplitude*

Dw=0.006;

W1=4; *//Time in seconds*

w=-W1/2:Dw:W1/2;

for i=1:length(w)

XW(i)=A;

end

XW=XW';

plot(w,XW);

xlabel('w in radians');

title('Continuous time fourier transform x(t)')

*//*

*//Inverse continuous-time Fourier Transform*

t=-%pi:%pi/length(w):%pi;

xt=(1/(2\*%pi))\*XW\*exp(sqrt(-1)\*w'\*t)\*Dw;

xt=real(xt);

figure

a=gca();

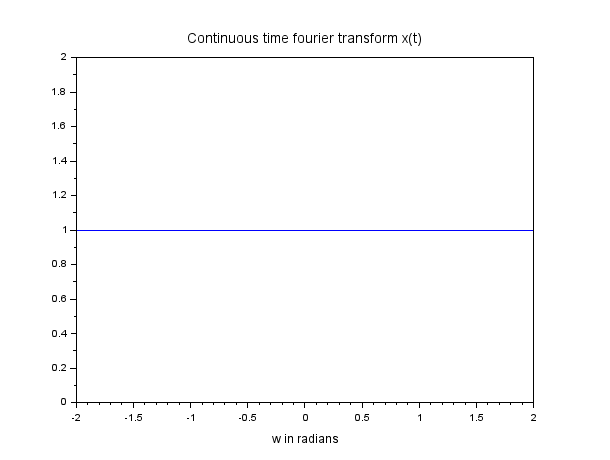
a.y\_location="origin";

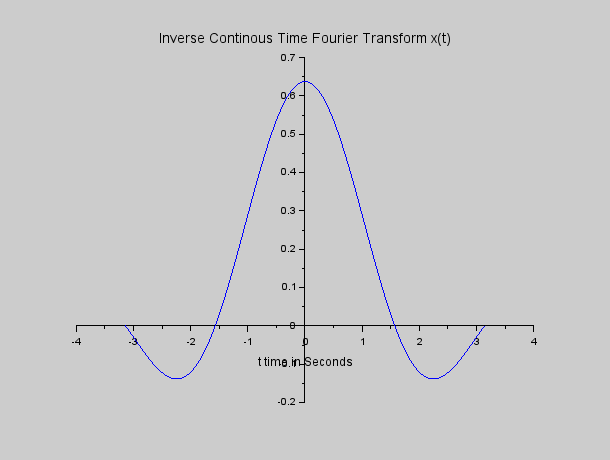
a.x\_location="origin";

plot(t,xt);

xlabel('t time in Seconds');

title('Inverse Continous Time Fourier Transform x(t)')





**Practical 4C**

DISCRETE TIME FOURIER TRANSFORM

1)

//Discreate time fourier tranform of discreate sequence x[n]=(a^n).u[n].a>0 and a<0

clear;

clc;

close;

//DTS signal

a1=0.5;

a2=-0.5;

max\_limit=10;

for n=0:max\_limit-1

x1(n+1)=(a1^n);

x2(n+1)=(a2^2);

end

n=0:max\_limit-1;

//discreate time fourier transform

Wmax=2\*%pi;

K=4;

k=0:(K/1000):K;

W=k\*Wmax/K;

x1=x1';

x2=x2';

XW1=x1\*exp(-sqrt(-1)\*n'\*W);

XW2=x2\*exp(-sqrt(-1)\*n'\*W);

XW1\_Mag=abs(XW1);

XW2\_Mag=abs(XW2);

[XW1\_Phase,db]=phasemag(XW1);

[XW2\_Phase,db]=phasemag(XW2);

//plot for a>0

figure

subplot(3,1,1);

plot2d3('gnn',n,x1);

xtitle('Discrete Time Sequence x[n] for a>0')

subplot(3,1,2);

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d(W,XW1\_Mag);

title('Magnitude Response abs(X(jW))')

subplot(3,1,1);

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d(W,XW1\_Phase);

title('Phase Response<(X(jW))')

//plot for a<0

figure

subplot(3,1,1);

plot2d3('gnn',n,x2);

xtitle('Discreate Time sequence x[n] for a>0')

subplot(3,1,2);

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d(W,XW2\_Phase);

title('Magnitude Response abs(X(jW))')

subplot(3,1,3);

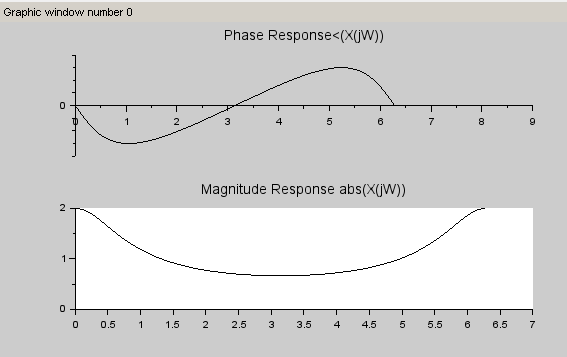
a=gca();

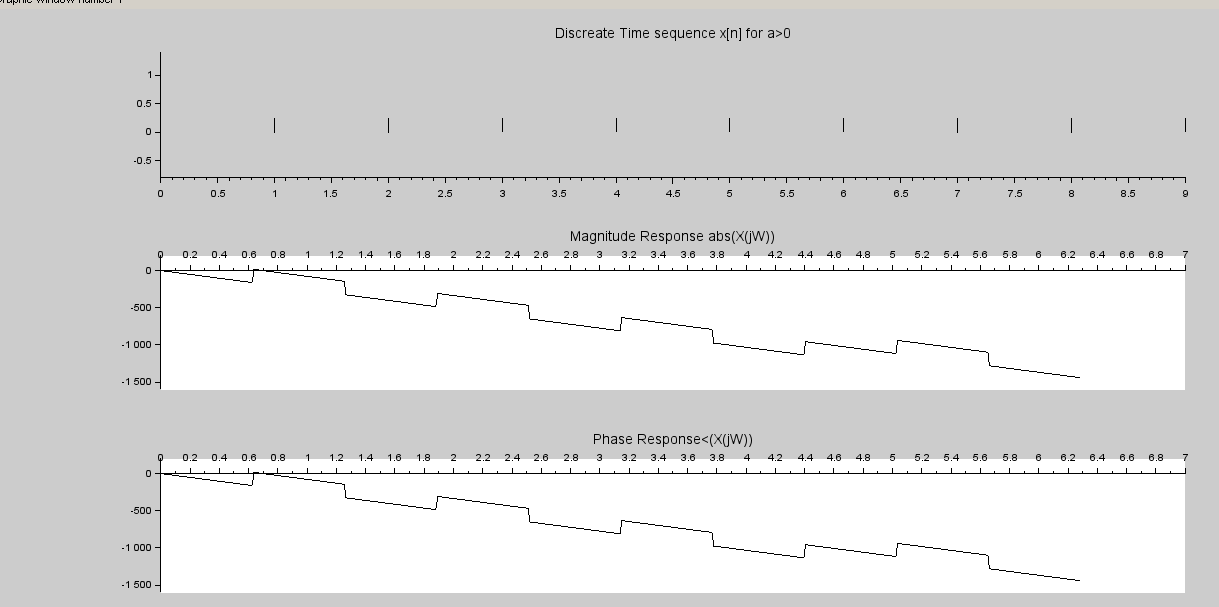
a.y\_location="origin";

a.x\_location="origin";

plot2d(W,XW2\_Phase);

title('Phase Response<(X(jW))')





3)

//discrete Time Fourier Transform of x[n]=1, abs(n)<=N1

clear;

clc;

close;

//DTS Signal

N1=2;

n=-N1:N1;

x=ones(1,length(n));

//Discrete-time Fourier Transform

Wmax=2\*%pi;

K=4;

k=0:(K/1000):K;

W=k\*Wmax/K

XW=x\*exp(-sqrt(-1)\*n'\*W);

XW\_Mag=real(XW);

//plot for abs(a)<1

figure

subplot(2,1,1);

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d3('gnn',n,x);

xtitle('Discrete Time Sequence x[n]')

subplot(2,1,2);

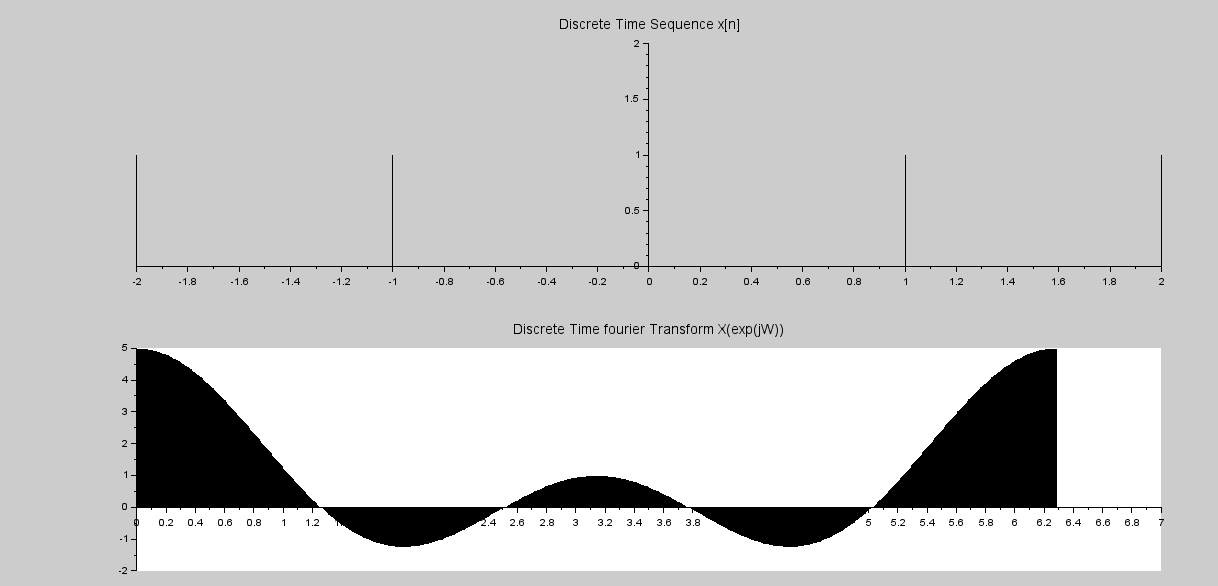
a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d3(W,XW\_Mag);

title('Discrete Time fourier Transform X(exp(jW))')



4)

//Discrete Time Fourier Transform :x[n]=cos(nWo)

clear;

clc;

close;

N=5;

Wo=2\*%pi/N;

W=[-Wo,0,Wo];

XW=[%pi,0,%pi];

//

figure

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d3('gnn',W,XW,2);

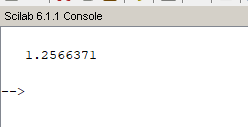
poly1=a.children(1).children(1);

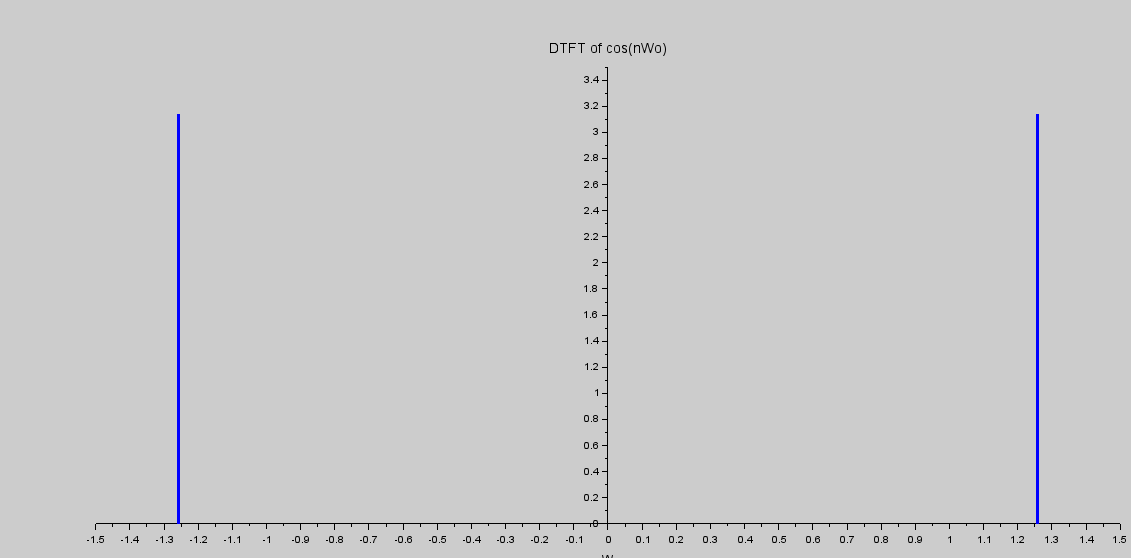
poly1.thickness=3;

xlabel(' W');

title('DTFT of cos(nWo)')

disp(Wo)





5)

Discrete Time Fourier Transform of discrete sequence X[n]=(n)\*(a^n).u[n]

a>0 and a<0

clc;

close;

a1=0.5;

a2=0.5;

max\_limit=10;

for n=0:max\_limit-1

x1(n+1)=(n)\*(a1^n);

x2(n+1)=(n)\*(a2^n);

end

n=0:max\_limit-1;

Wmax=2\*%pi;

K=4;

k=0:(K/1000):K;

W=k\*Wmax/K;

x1=x1';

x2=x2';

XW1=x1\*exp(-sqrt(-1)\*n'\*W);

XW2=x2\*exp(-sqrt(-1)\*n'\*W);

XW1\_Mag=abs(XW1);

XW2\_Mag=abs(XW2);

[XW1\_Phase,db]=phasemag(XW1);

[XW2\_Phase,db]=phasemag(XW2);

figure

subplot(3,1,1);

plot2d3('gnn',n,x1);

xtitle('Discrete Time Sequence x[n] for a>0')

subplot(3,1,2);

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d(W,XW1\_Mag);

title('Magnitude Response abs(X(jW))')

subplot(3,1,3);

a=gca();

a.y\_location="origin";

a.x\_location="origin";

plot2d(W,XW1\_Phase);

title('Phase Response<(X(jW))')

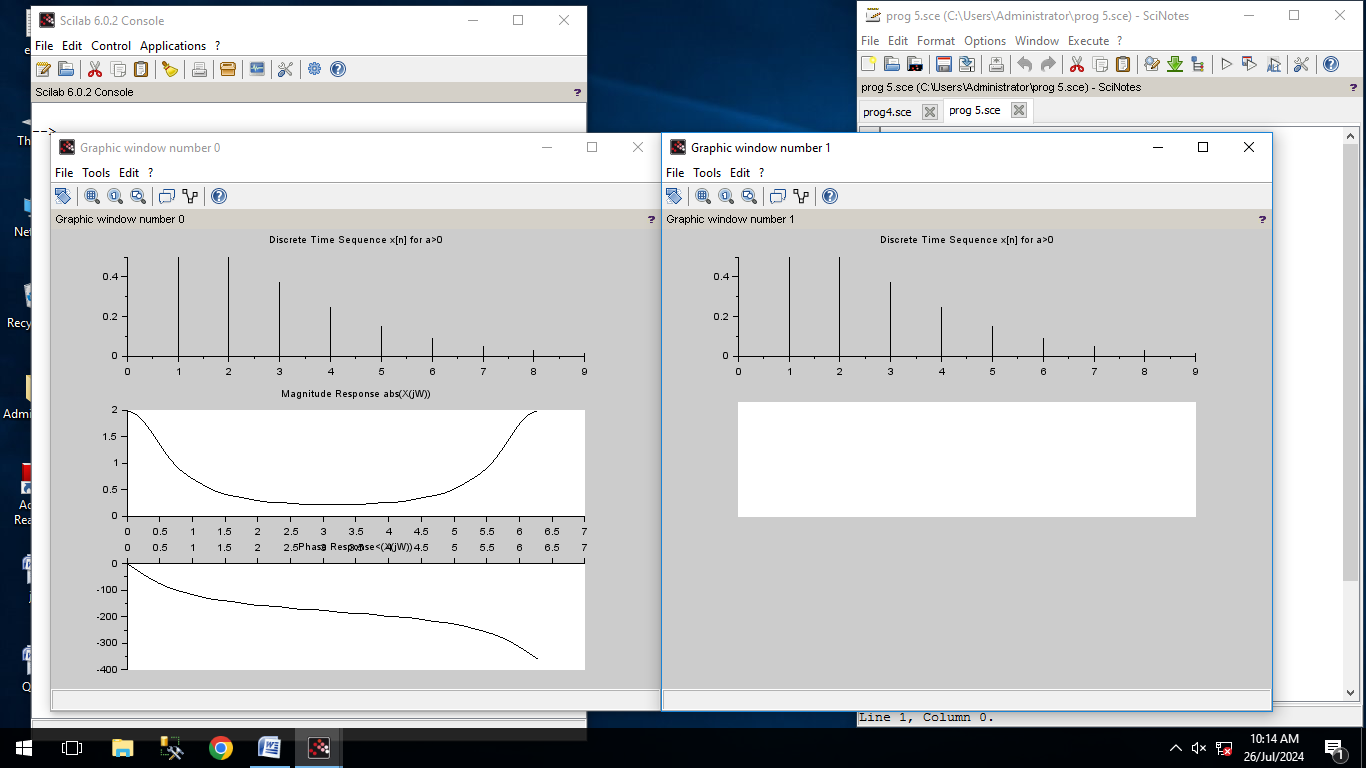
figure

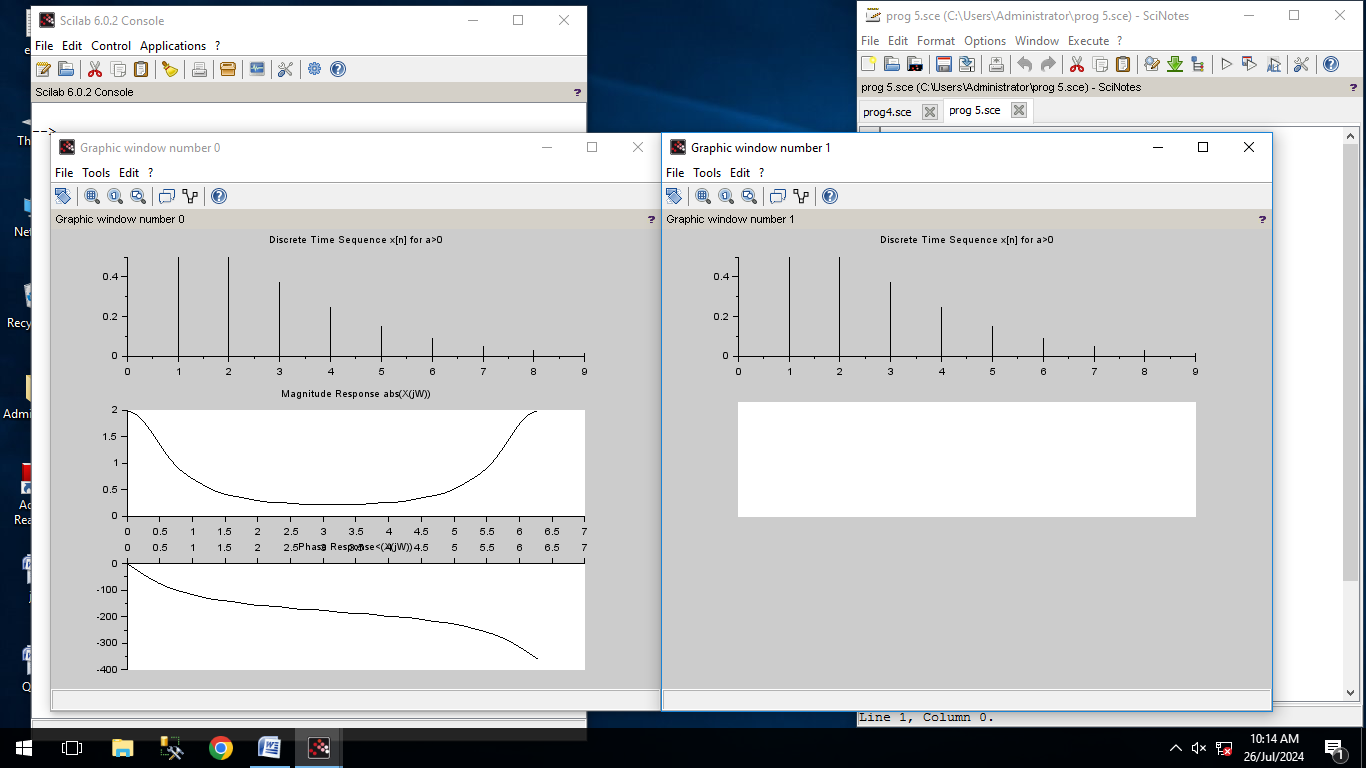
subplot(3,1,1);

plot2d3('gnn',n,x2);

xtitle('Discrete Time Sequence x[n] for a>0')

subplot(3,1,2)





**Practical 4D**

**Practical 5**

**Z transform**

A)

clear;

*//z transform of[2 -1 3 2 1 0 2 3 -1]*

clear;

clc;

close;

function [**za**]=ztransfer(**sequence**, **n**)

z=poly(0,'z','r')

**za**=**sequence**\*(1/z)^**n**'

endfunction

x1=[2 -1 3 2 1 0 2 3 -1]

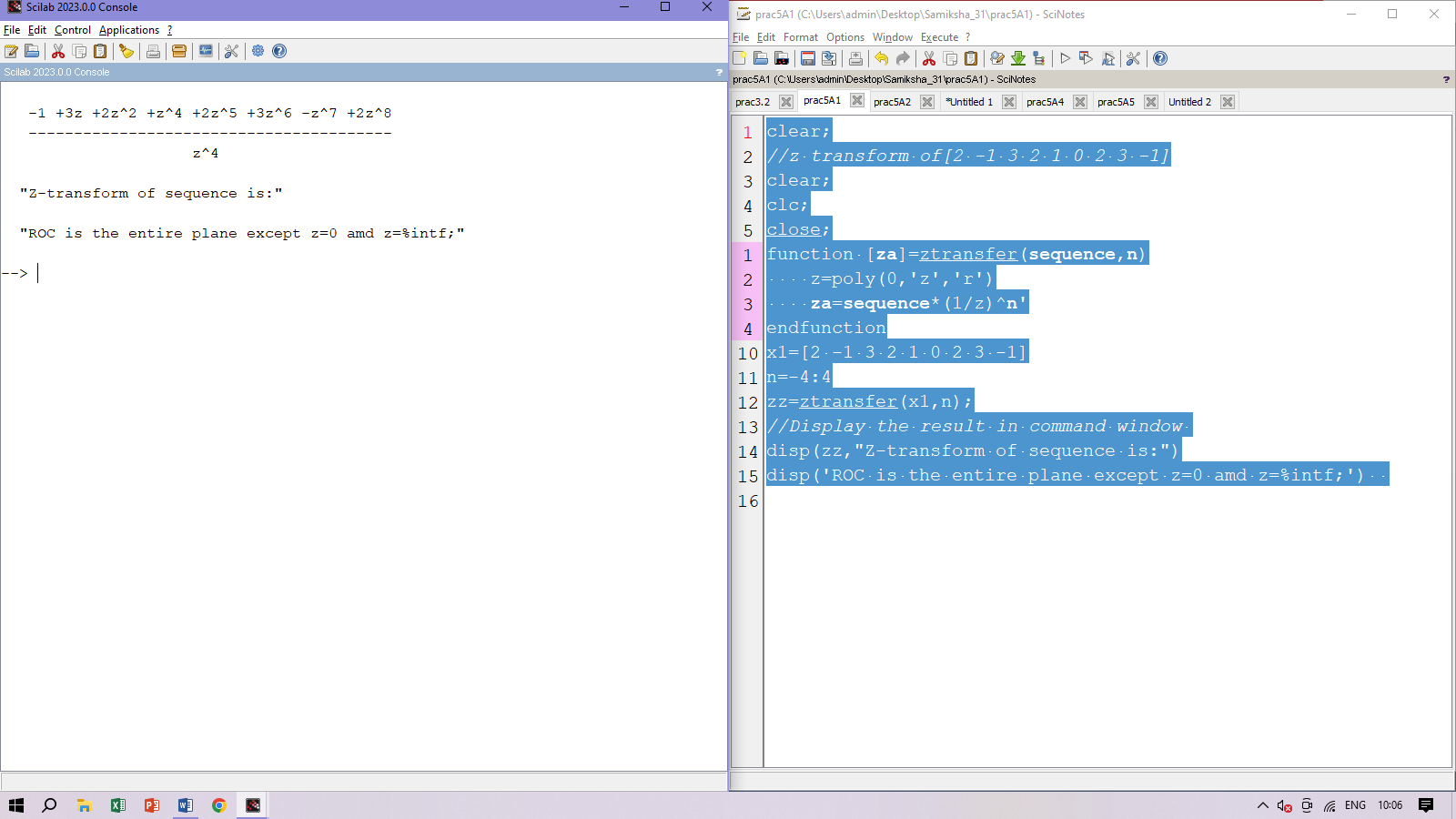
n=-4:4

zz=ztransfer(x1,n);

*//Display the result in command window*

disp(zz,"Z-transform of sequence is:")

disp('ROC is the entire plane except z=0 amd z=%intf;')



B)

clear;

*//Z transform of [1,2,3,4,5,6,7]*

clc;

function [**za**]=ztransfer(**sequence**, **n**)

z=poly(0,'z','r')

**za**=**sequence**\*(1/z)^**n**'

endfunction

x=[1,2,3,4,5,6,7];

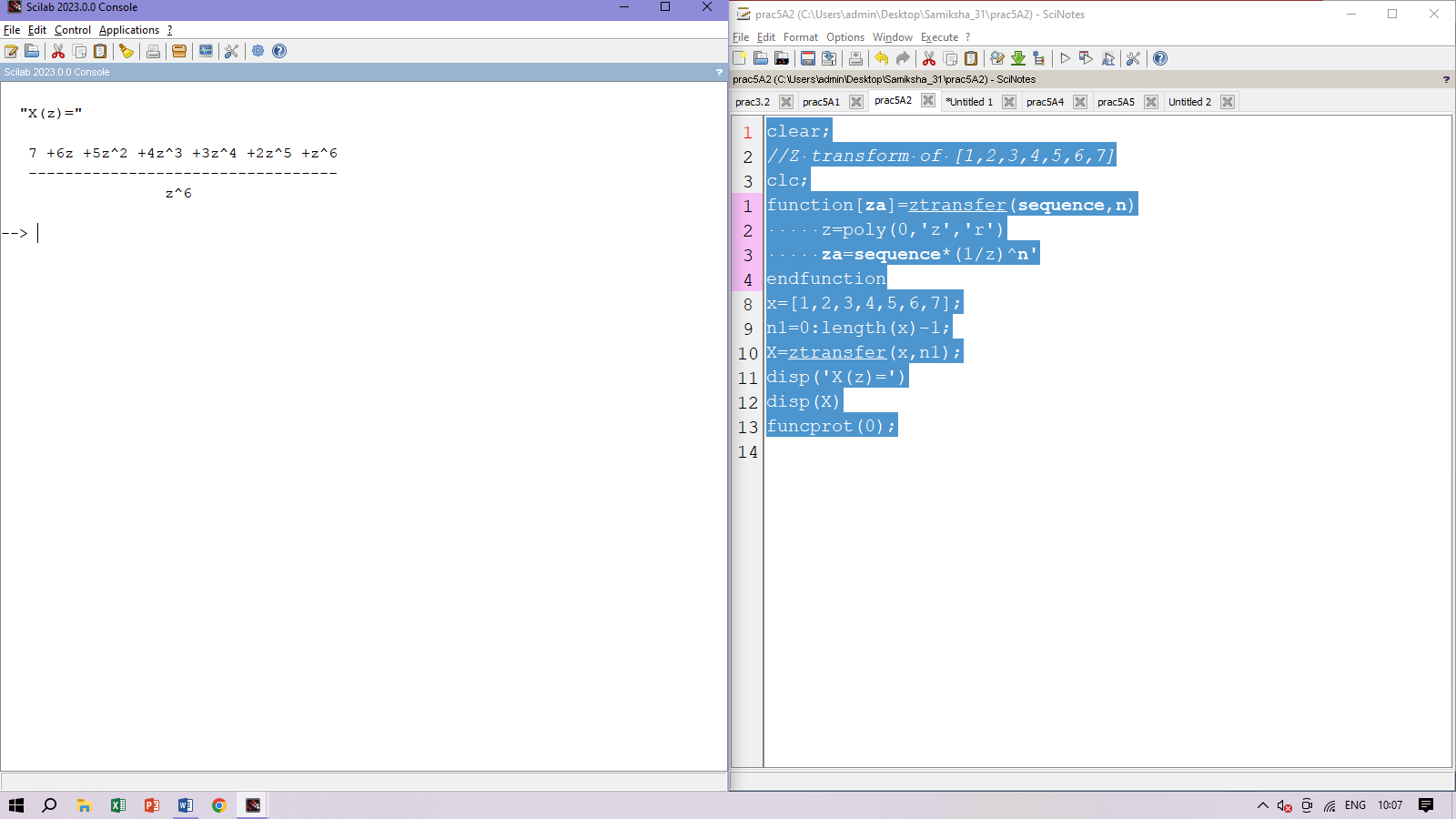
n1=0:length(x)-1;

X=ztransfer(x,n1);

disp('X(z)=')

disp(X)

funcprot(0);



C)

clear

*//Z transform of [1,2,3,4,5,6,7] with different range*

clc;

function [**za**]=ztransfer(**sequence**, **n**)

z=poly(0,'z','r')

**za**=**sequence**\*(1/z)^**n**'

endfunction

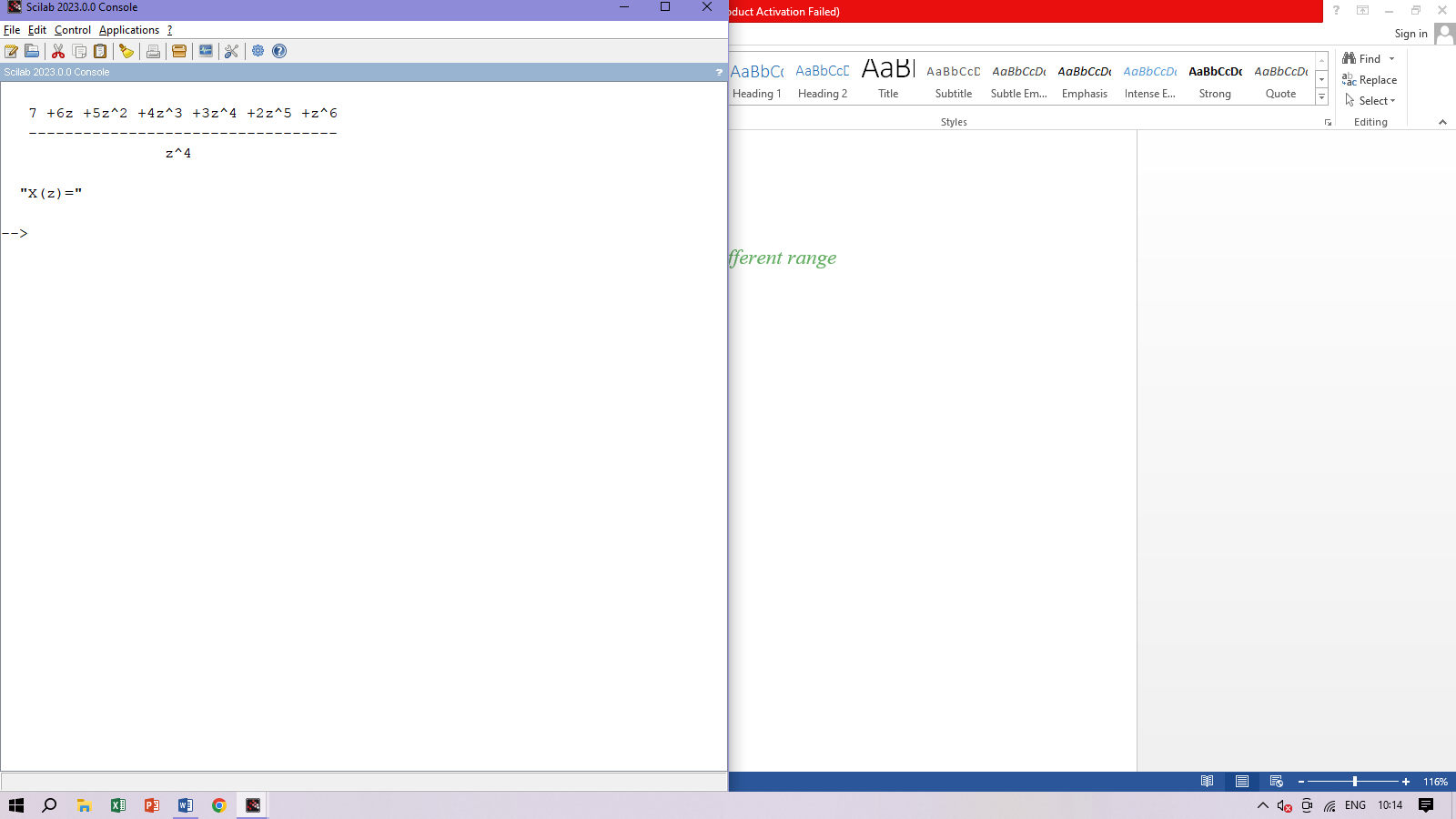
x=[1,2,3,4,5,6,7]

n1=-2:length(x)-3

X=ztransfer(x,n1)

disp(X,'X(z)=')

funcprot(0);



D)

clear;

*//Z-transform of[1,2,3,4,5,0,7]*

clc;

function [**za**]=ztransfer(**sequence**, **n**)

z=poly(0,'z','r')

**za**=**sequence**\*(1/z)^**n**'

endfunction

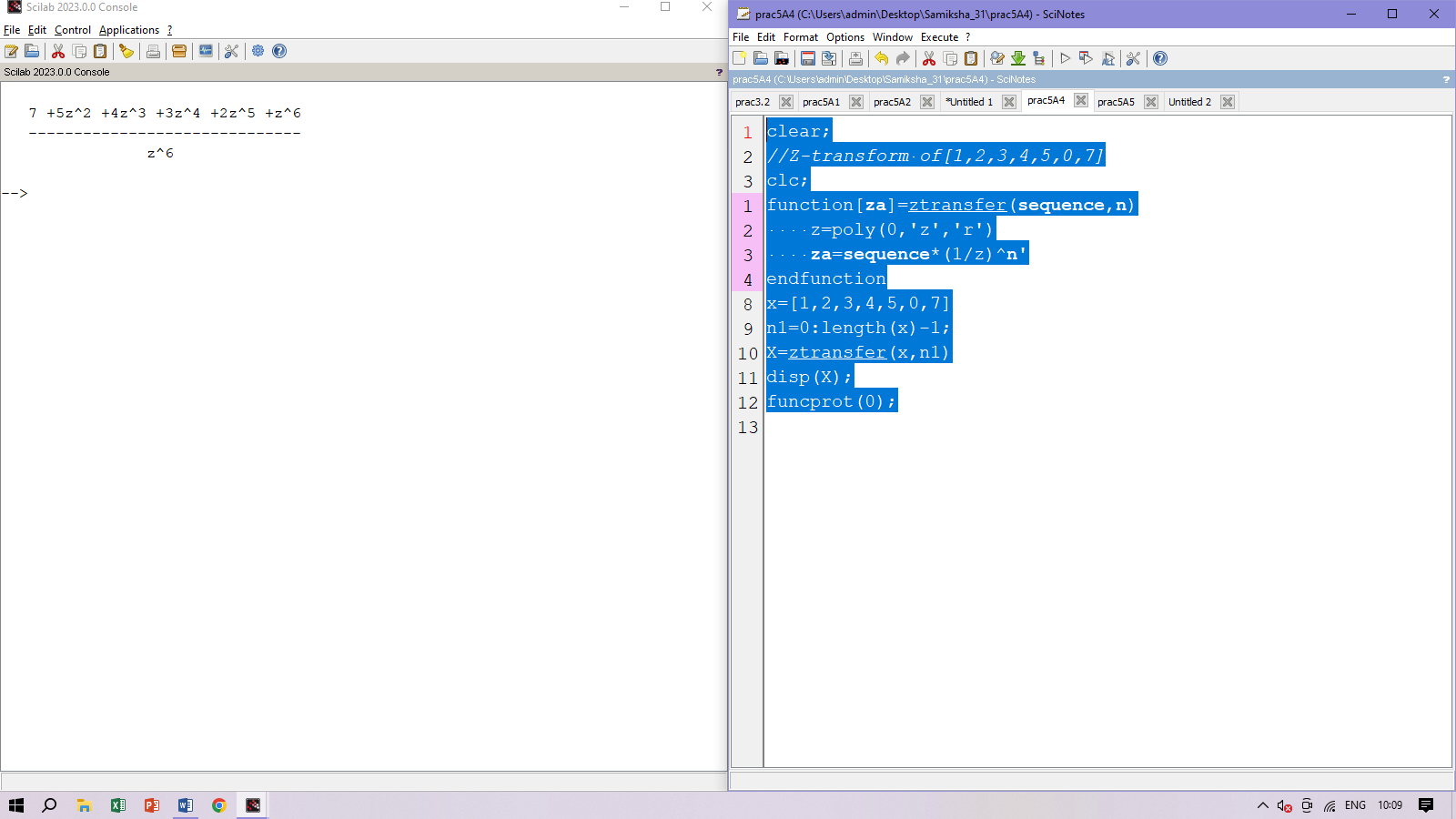
x=[1,2,3,4,5,0,7]

n1=0:length(x)-1;

X=ztransfer(x,n1)

disp(X);

funcprot(0);



E)

*//Z transform of [4,2,-1,0,3,-4]*

clc;

function [**za**]=ztransfer(**sequence**, **n**)

z=poly(0,'z','r')

**za**=**sequence**\*(1/z)^**n**'

endfunction

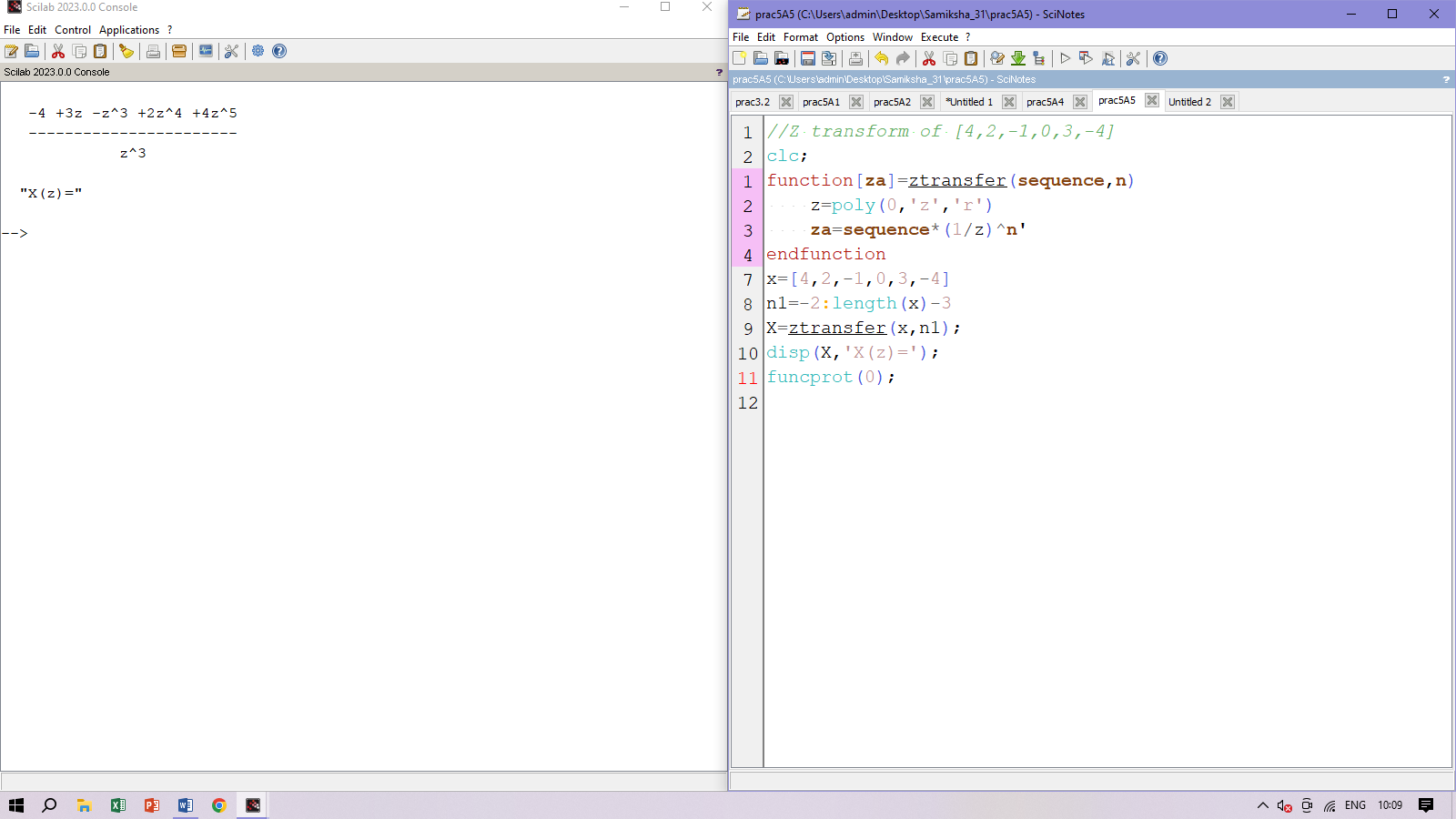
x=[4,2,-1,0,3,-4]

n1=-2:length(x)-3

X=ztransfer(x,n1);

disp(X,'X(z)=');

funcprot(0);



F)

*//Convolution of two signals x1 and x2*

clc;

function [**za**]=ztransfer(**sequence**, **n**)

z=poly(0,'z','r')

**za**=**sequence**\*(1/z)^**n**'

endfunction;

x1=[1,-3,2];

n1=0:length(x1)-1;

X1=ztransfer(x1,n1);

x2=[1,2,1];

n2=0:length(x2)-1;

X2=ztransfer(x2,n2);

X=X1\*X2;

disp(X,'X(z)=');

z=poly(0,'z');

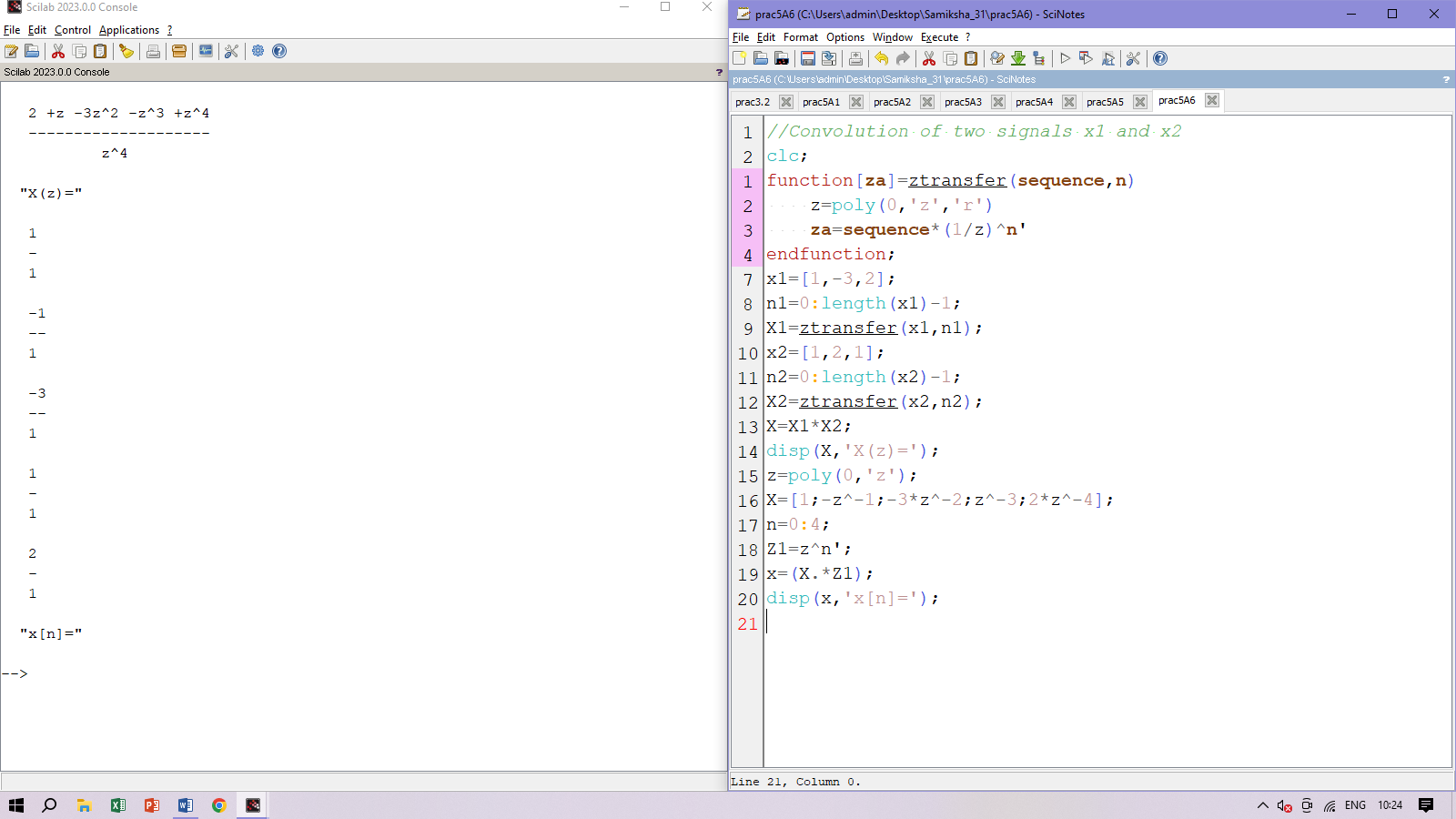
X=[1;-z^-1;-3\*z^-2;z^-3;2\*z^-4];

n=0:4;

Z1=z^n';

x=(X.\*Z1);

disp(x,'x[n]=');



**Practical 5 B**

**Inverse Z-Transform**

A]

clear;

*//Find the inverse Z-transform using long divison method*

clc;

clear;

z=poly(0,'z')

x=ldiv(z^3-10\*z^2-4\*z+4,2\*z^2-2\*z-4,4);

disp(x,'x[n]=');



B]

clear

*//Find the inverse Z-transform*

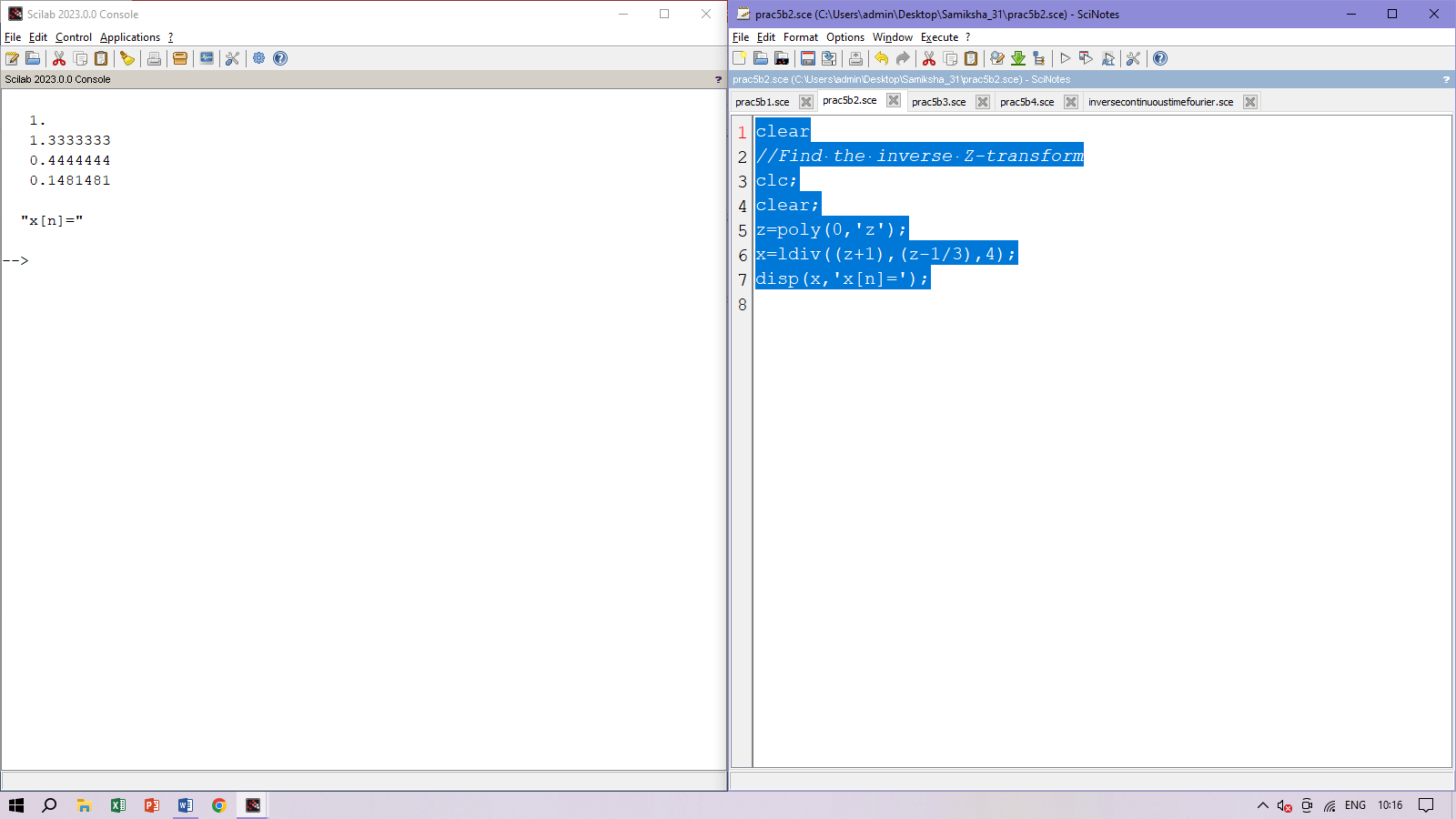
clc;

clear;

z=poly(0,'z');

x=ldiv((z+1),(z-1/3),4);

disp(x,'x[n]=');



C]

clear;

*//Inverse Z-transform using long division method*

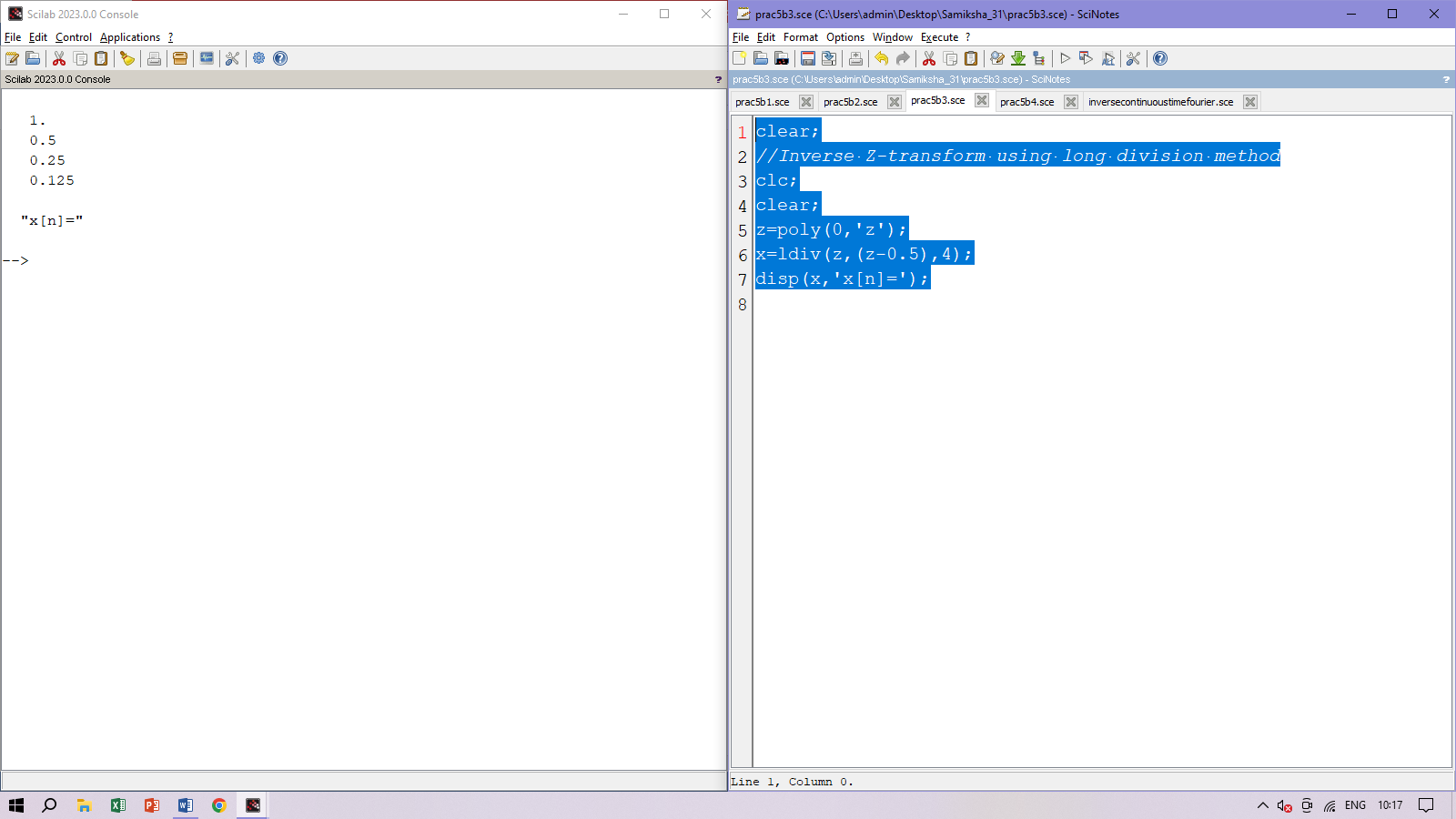
clc;

clear;

z=poly(0,'z');

x=ldiv(z,(z-0.5),4);

disp(x,'x[n]=');



D]clear

*//To find input x(n)*

*//X(z)=1/(2\*z^(-2)+2\*z^(-1)+1);*

clear;

clc;

close;

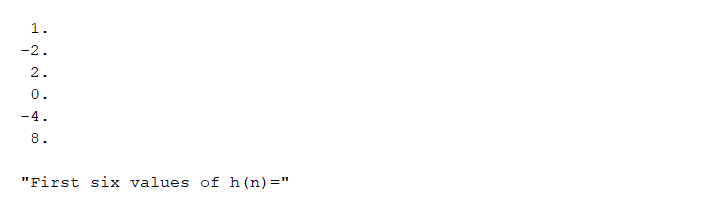
z=%z;

a=(2+2\*z+z^2);

b=z^2;

h=ldiv(b,a,6);

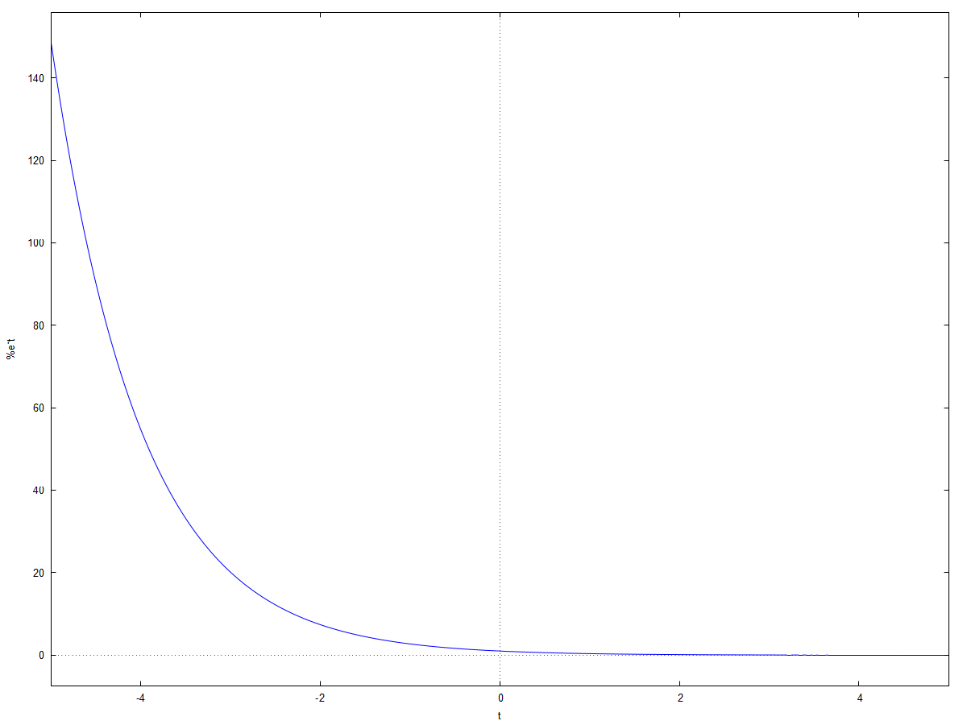
disp(h,"First six values of h(n)=")



**Practical 6**

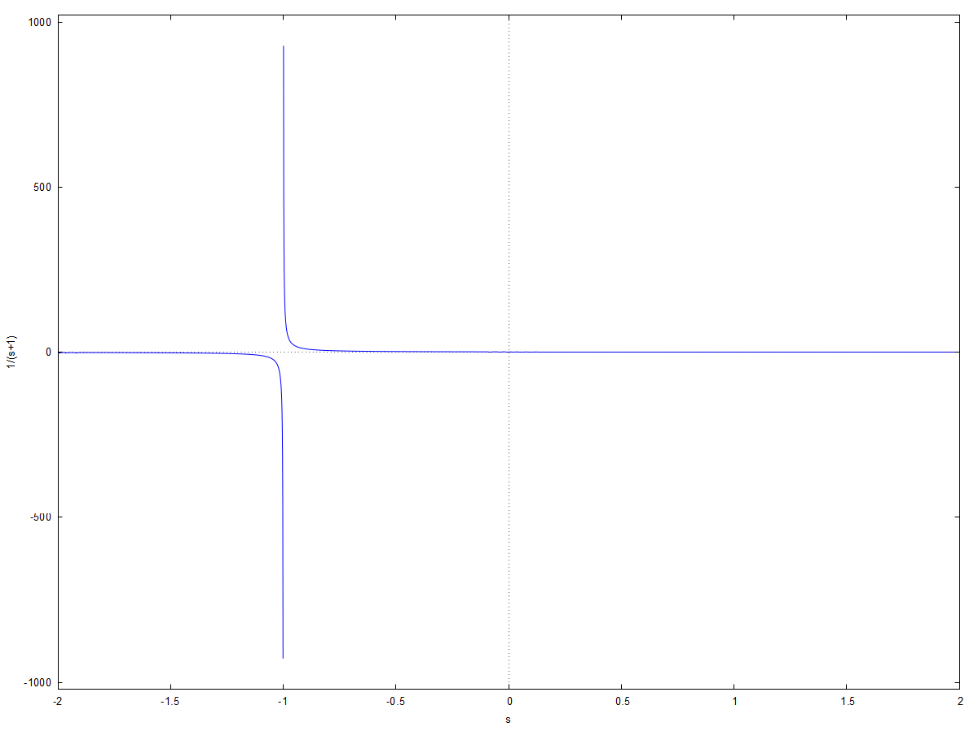
A] Y=e^(-t)

wxplot2d([%e^(-t)], [t,-5,5])$

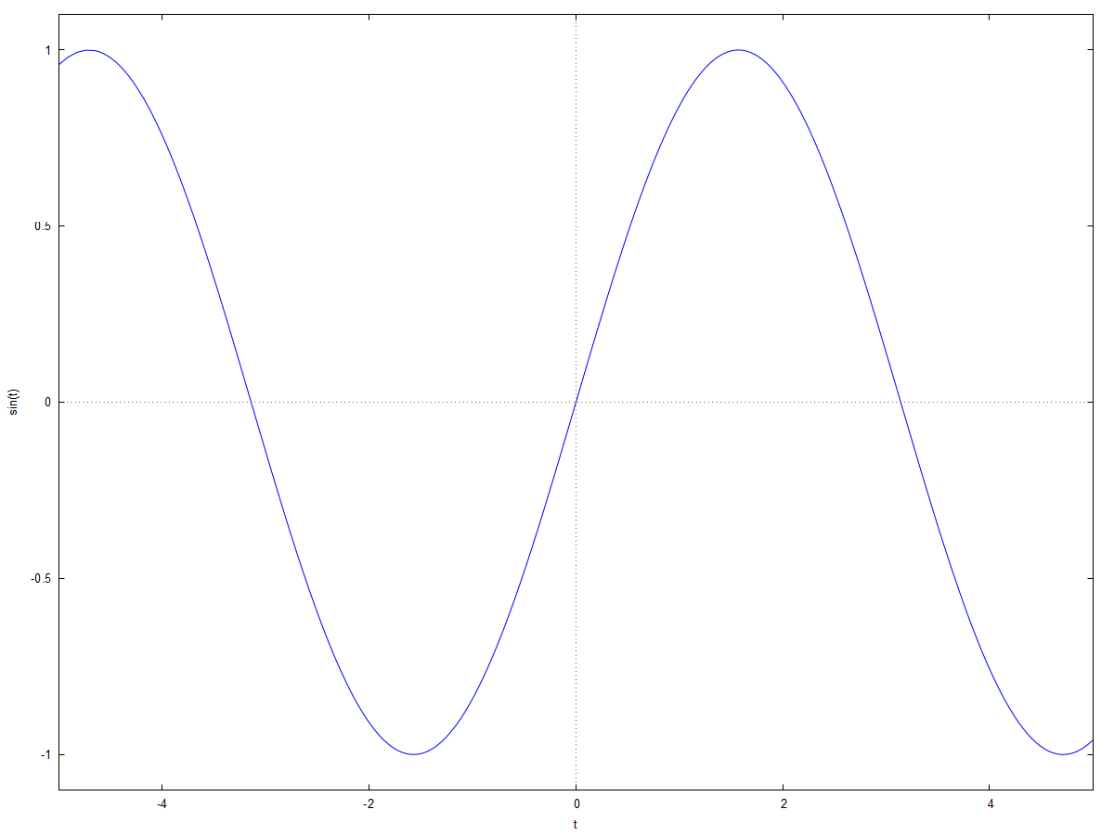


Laplace:

laplace(%e^(-t),t,s);



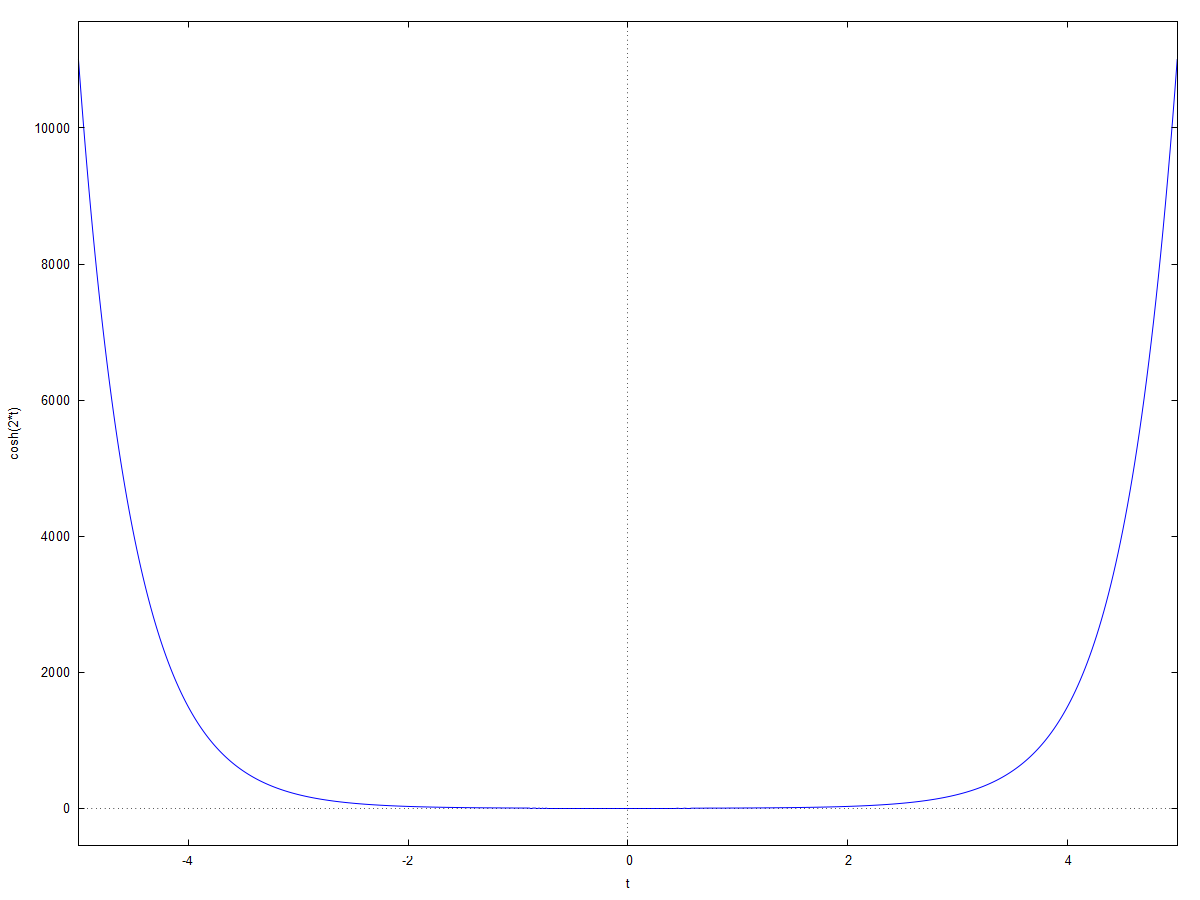
B) sint



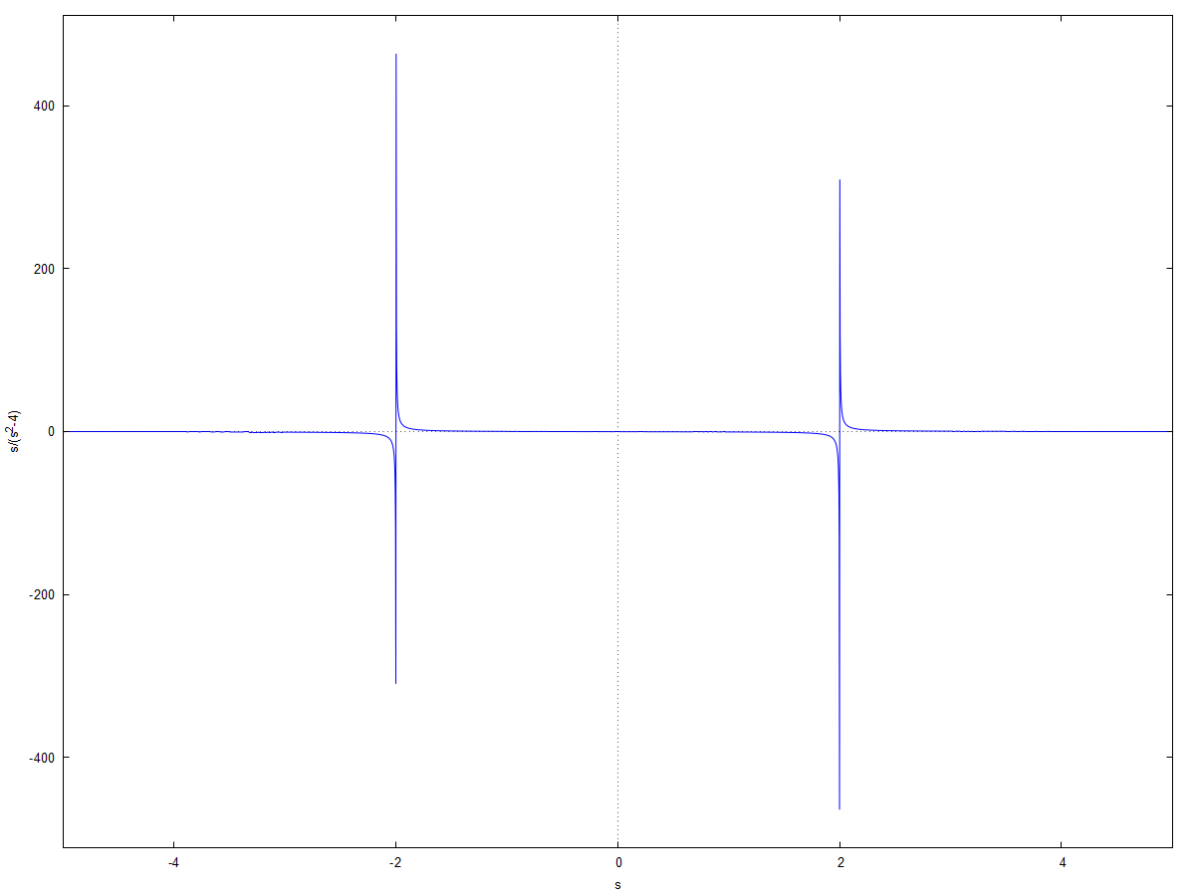
Laplace:



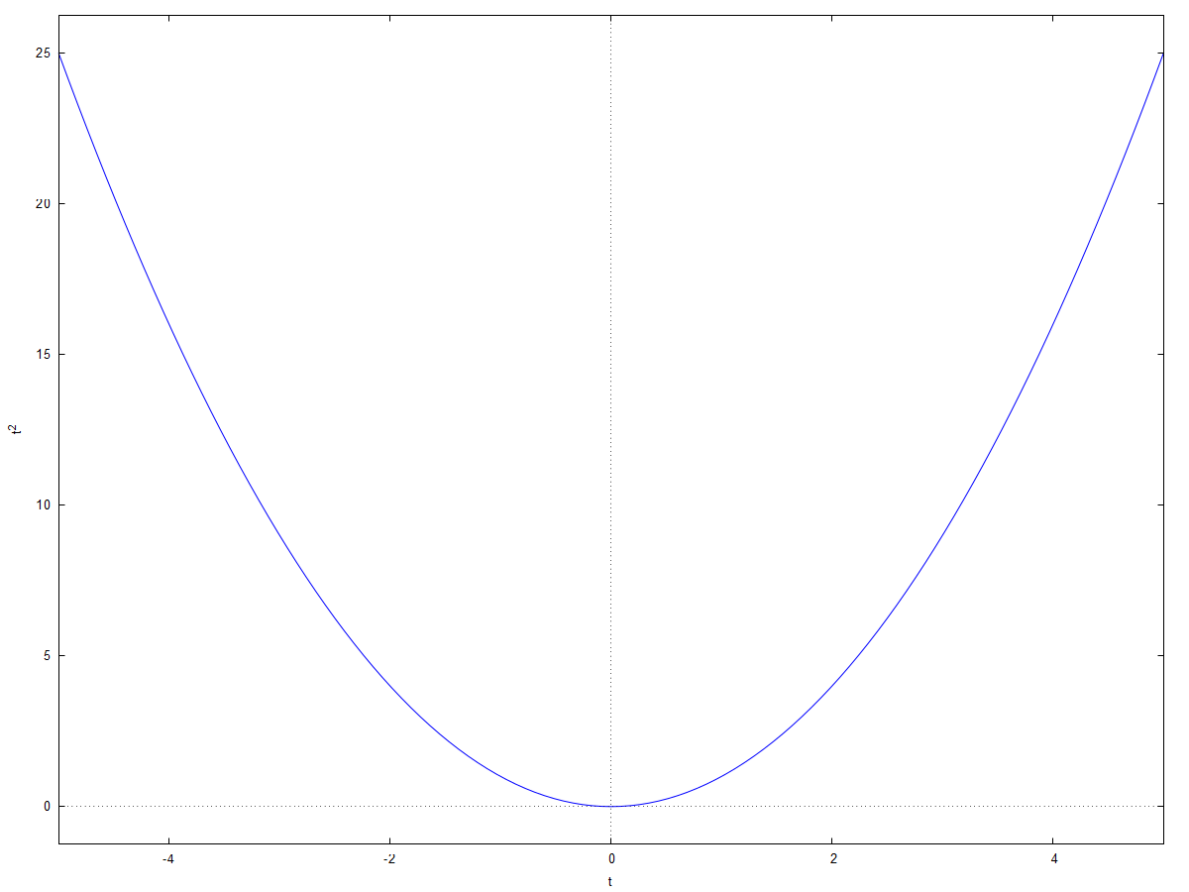
C) cosh2t:



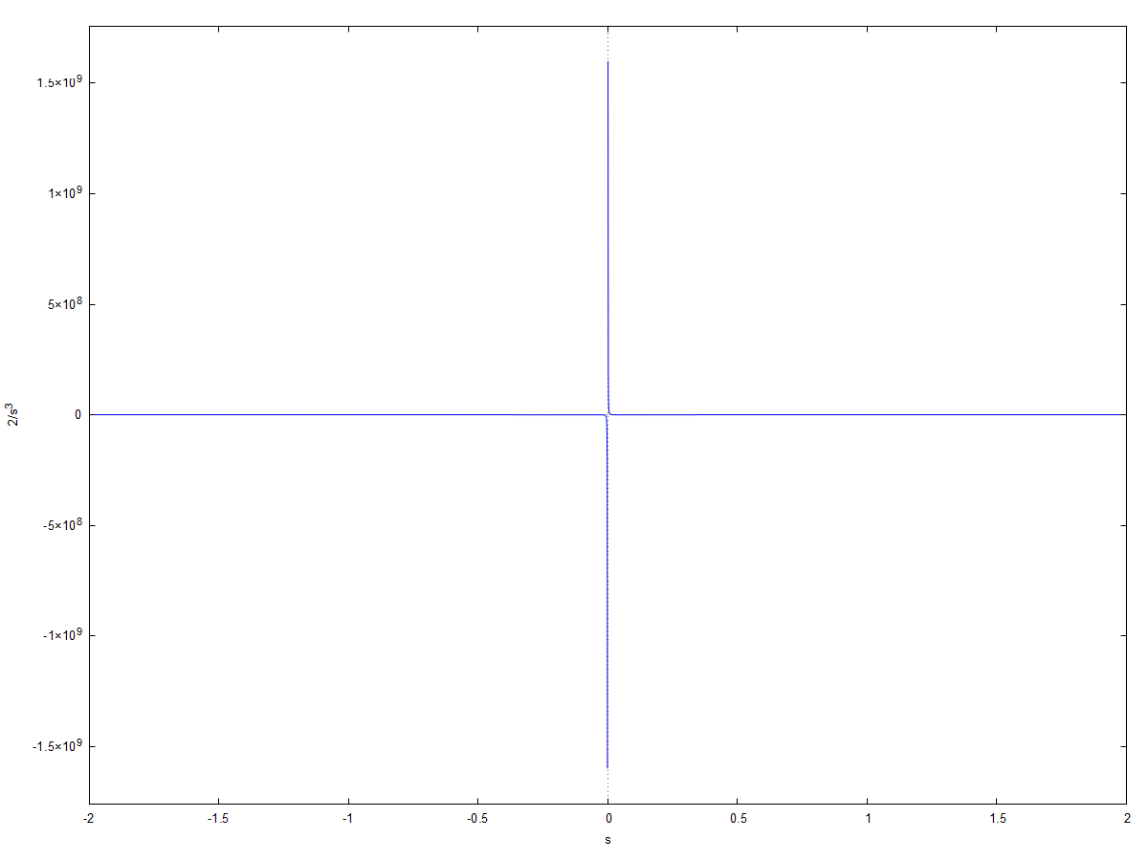
Laplace:



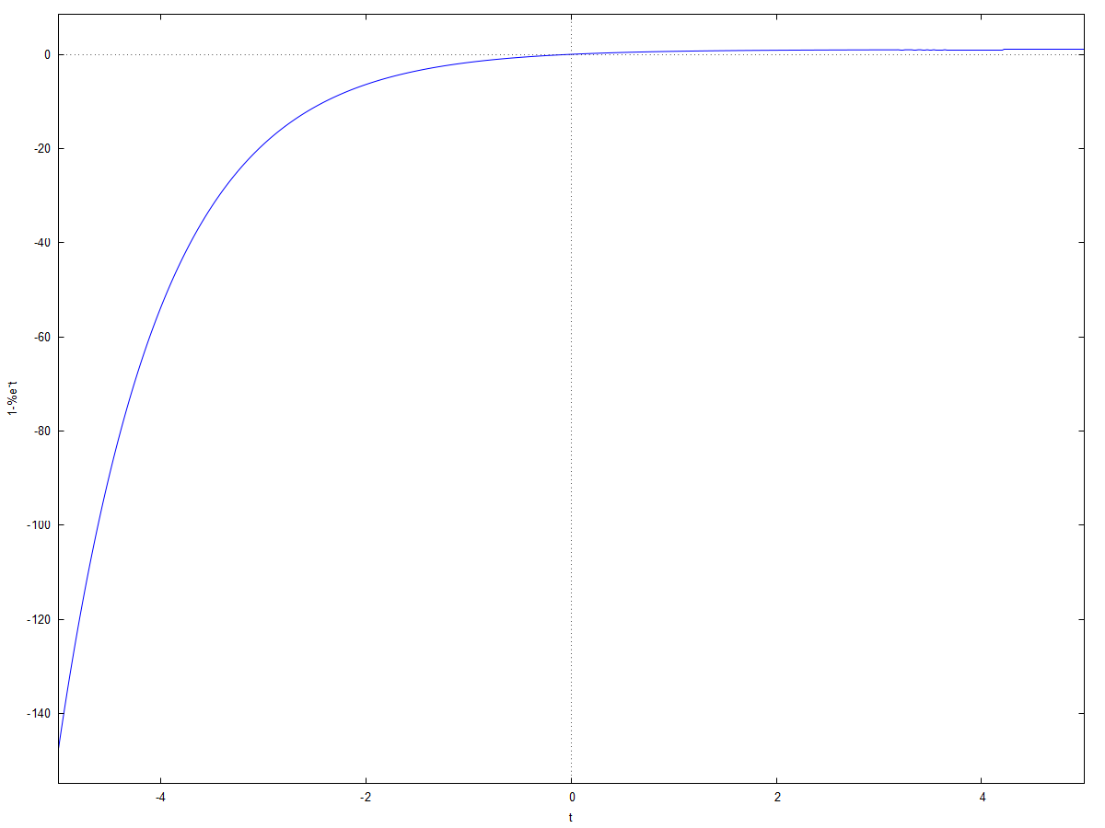
D) t^2:



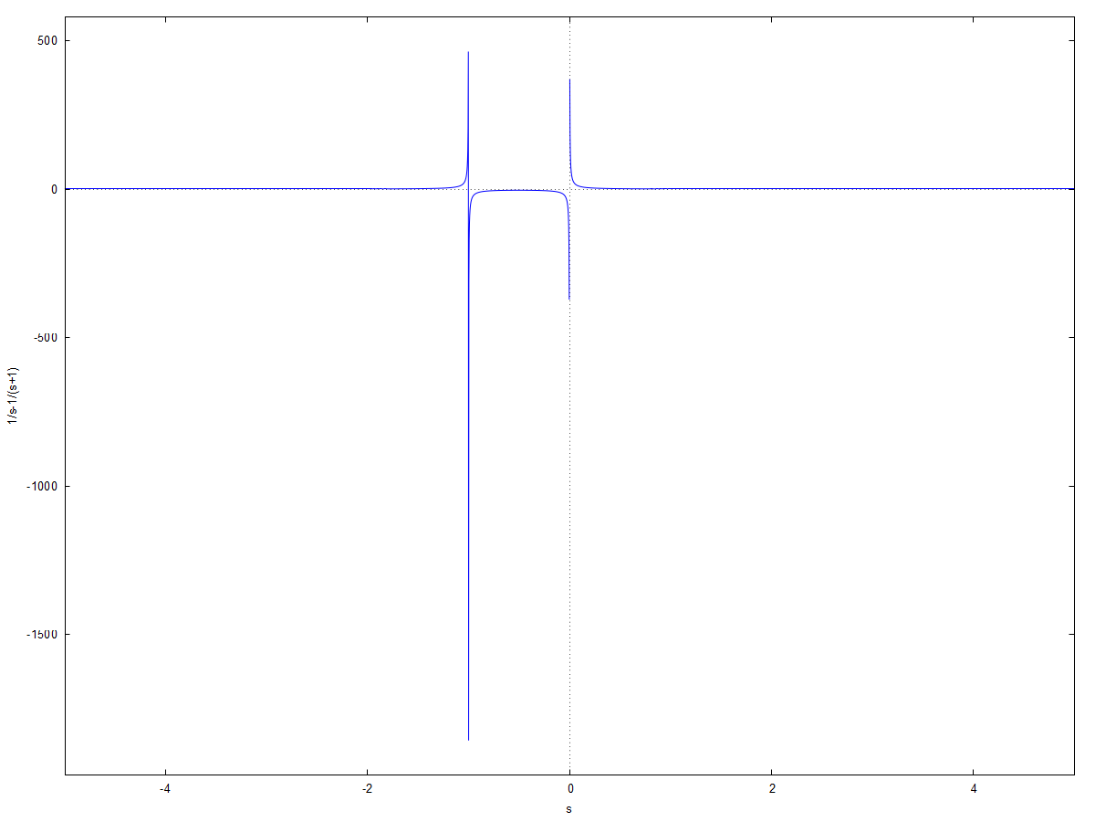
Laplace:



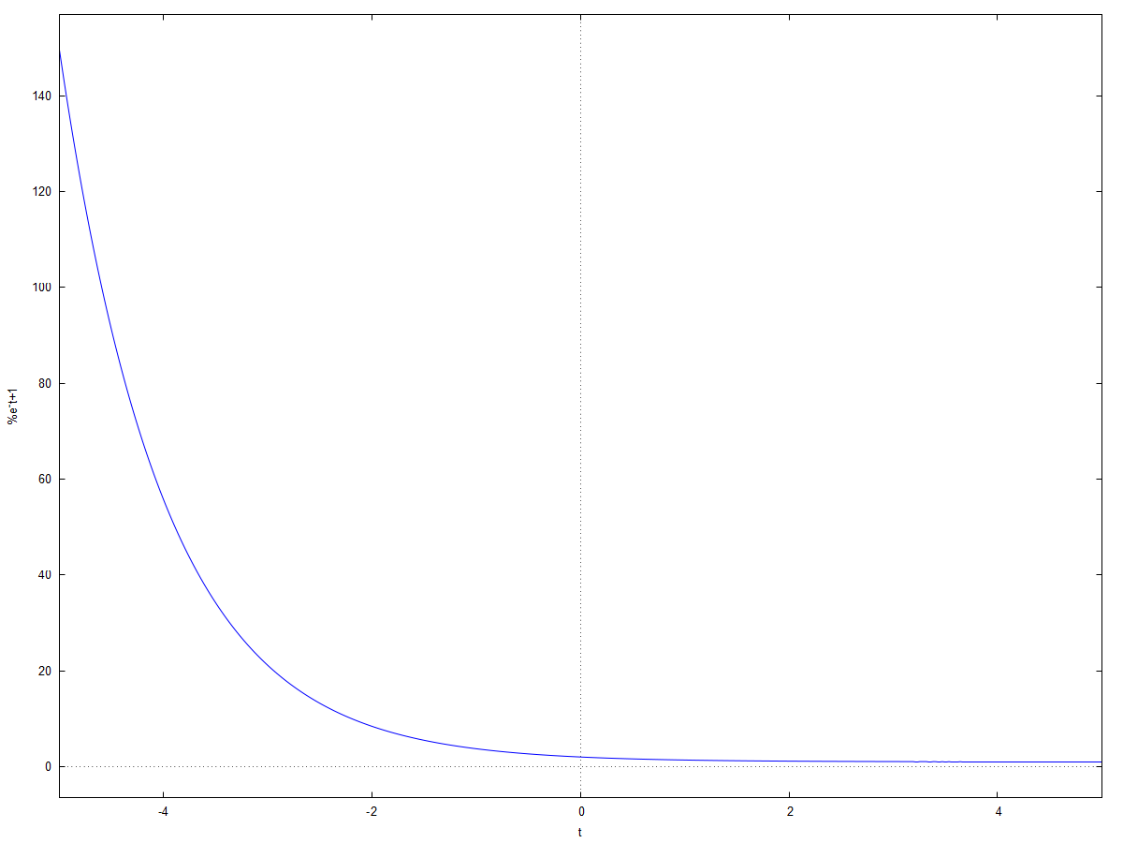
E) 1- e^(-t)



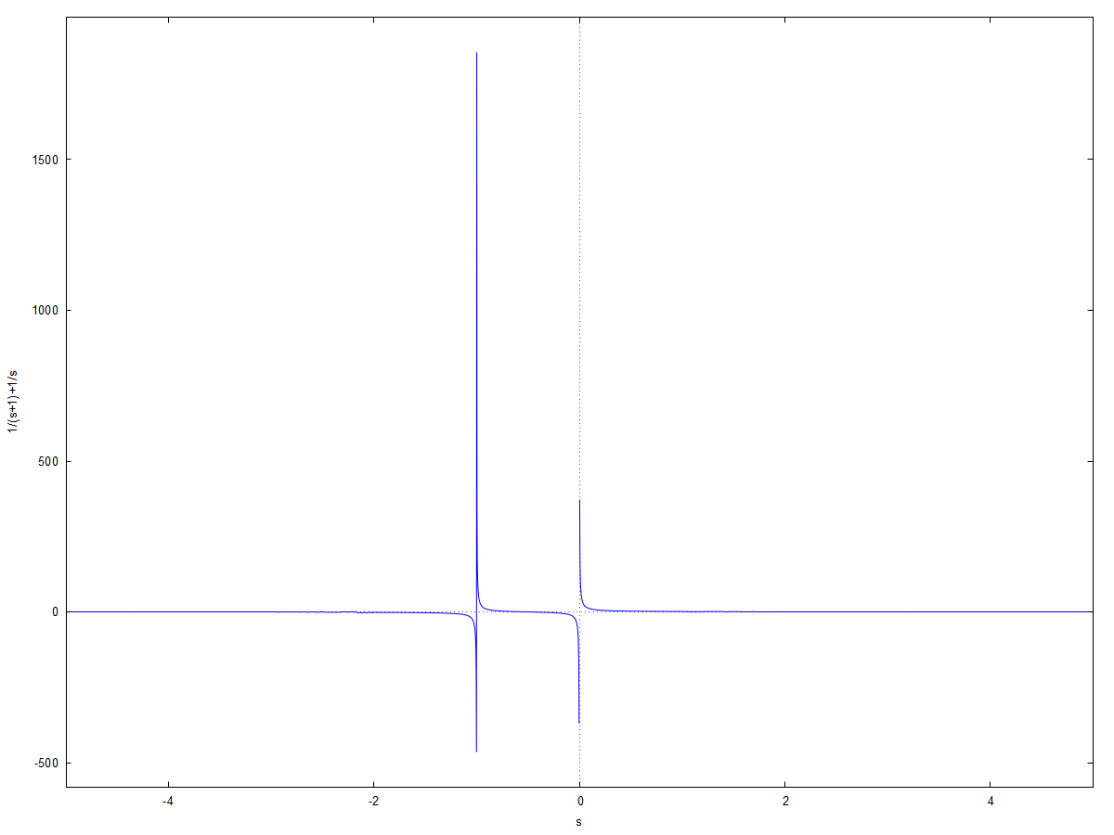
Laplace:



F)1+e^(-t):



Laplace:



**Practical 7**

**System response using Laplace Transform**

A] clear;

clc;

clear;

close;

s=poly(0,'s')

N=(s+1)\*(s+3);

D=(s+2)\*(s+4);

F=N/D;

disp(F,'Given Transfer Function:');

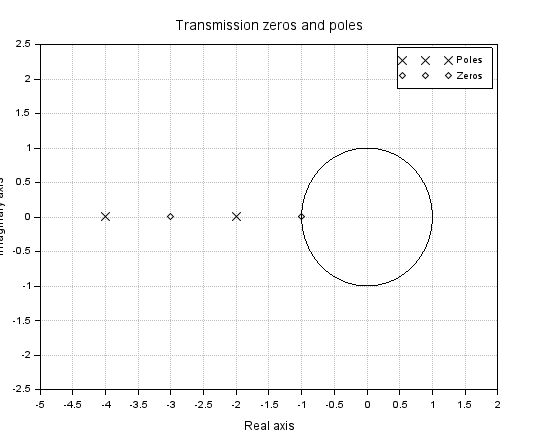
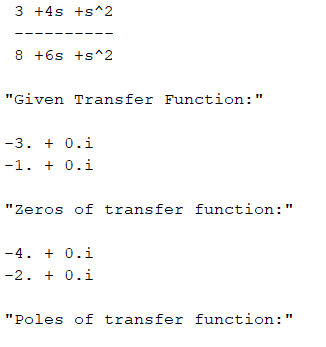
zero=roots(N);

pole=roots(D);

disp(zero,'Zeros of transfer function:');

disp(pole,'Poles of transfer function:');

plzr(F)



B] clear;

clc;

clear;

close;

s=poly(0,'s');

l=3\*s/(s+2)/(s+4)

disp(l,'Given Transfer Function:');

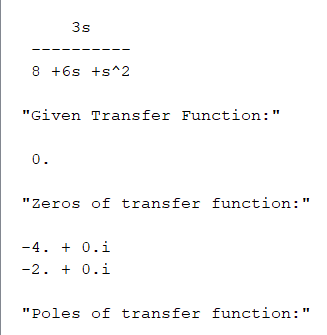
zero=roots(3\*s);

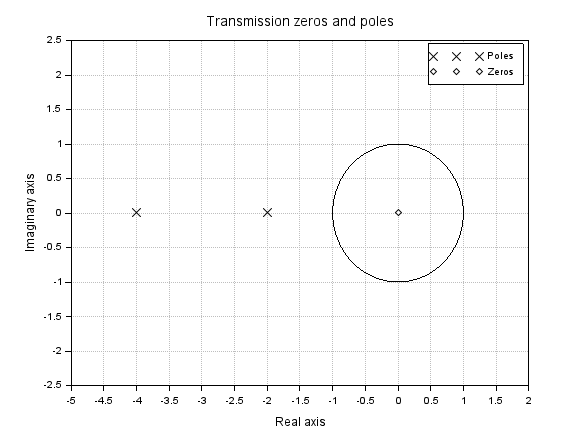
pole=roots((s+2)\*(s+4));

disp(zero,'Zeros of transfer function:');

disp(pole,'Poles of transfer function:');

plzr(l)





C] clear;

clc;

clear;

close;

s=poly(0,'s')

F=10\*s/(s^2+2\*s+2);

disp(F,'Given Transfer Function:');

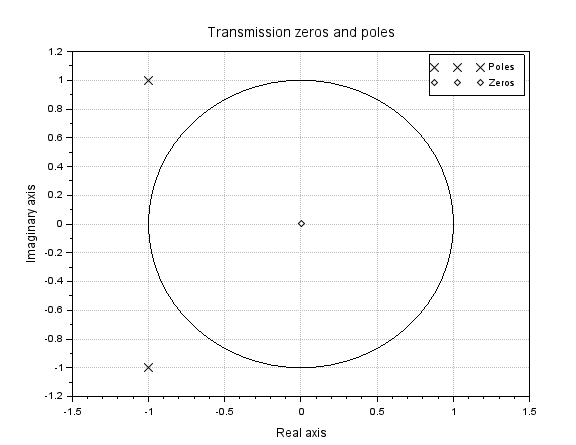
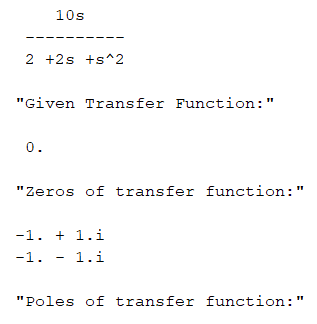
zero=roots(10\*s);

pole=roots(s^2+2\*s+2);

disp(zero,'Zeros of transfer function:');

disp(pole,'Poles of transfer function:');

plzr(F)



**Practical 8**

A]*//To plot the response of the system analytically and using scilab*

clear;

clc;

close;

n=0:1:20;

x=[1 zeros(1,20)];

b=[1 -0.5];

a=[1 -1 3/16];

yanaly=0.5\*(0.75).^n+0.5\*(0.25).^n; *//Analytical Solution*

ymat=filter(b,a,x);

subplot(3,1,1);

plot2d3(n,x);

xlabel('n');

ylabel('x(n)');

title('INPUT SEQUENCE (IMPULSE FUNCTION)');

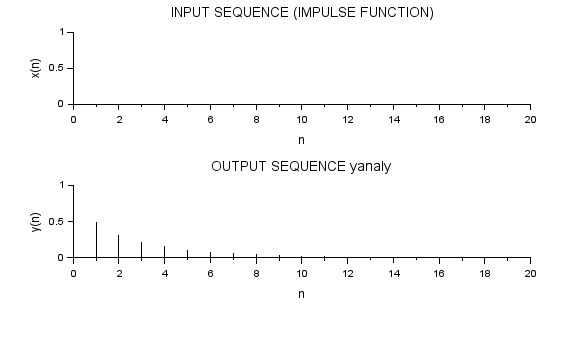
subplot(3,1,2);

plot2d3(n,yanaly);

xlabel('n');

ylabel('y(n)');

title('OUTPUT SEQUENCE yanaly');



B]*//To plot the response of system analytically and using scilab*

clear;

clc;

close;

n=0:1:20;

x=n;

b=[0 1 1];

a=[1 -0.7 0.12];

yanaly=38.89\*(0.4).^n-26.53\*(0.3).^n-12.36+4.76\*n; *//Analytical Solution*

ymat=filter(b,a,x);

subplot(3,1,1);

plot2d3(n,x);

xlabel('n');

ylabel('x(n)');

title('INPUT SEQUENCE(RAMP FUNCTION)');

subplot(3,1,2);

plot2d3(n,yanaly);

xlabel('n');

ylabel('y(n)');

title('OUTPUT SEQUENCE yanaly');

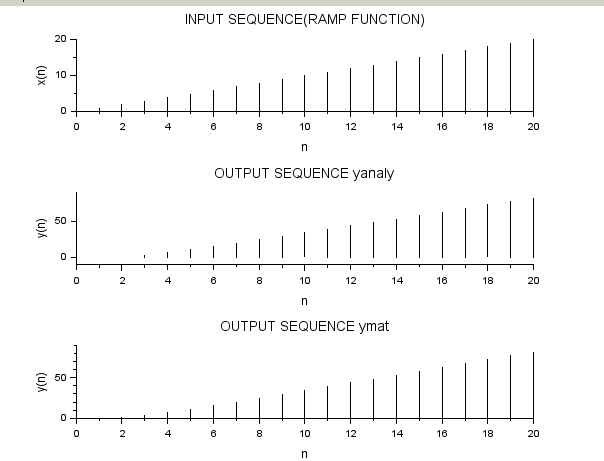
subplot(3,1,3);

plot2d3(n,ymat);

xlabel('n');

ylabel('y(n)');

title('OUTPUT SEQUENCE ymat');



C]

*//To plot the response of system analytically and using scilab*

clear;

clc;

close;

n=0:1:20;

x=ones(1,length(n));

b=[0 1];

a=[1 -1 -1];

yanaly=0.447\*(1.618).^n-0.447\*(-0.618).^n;*//Analytical Solution*

[ymat,zf]=filter(b,a,x);

subplot(3,1,1);

plot2d3(n,x);

xlabel('n');

ylabel('x(n)');

title('Input Sequence(Step function)');

subplot(3,1,2);

plot2d3(n,yanaly);

xlabel('n');

ylabel('y(n)');

title('Output Sequence yanaly');

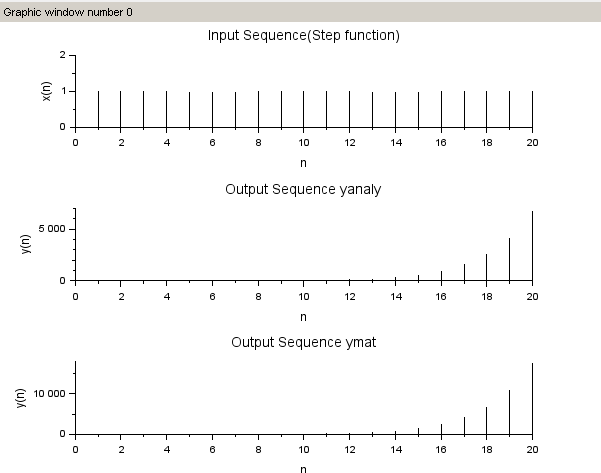
subplot(3,1,3);

plot2d3(n,ymat,zf);

xlabel('n');

ylabel('y(n)');

title('Output Sequence ymat')



D]

clear;

*//To find input h(n)*

*//X(z)=(z+0.2)/((z+0.5)(z-1))*

clear;

clc;

close;

z=%z;

a=(z+0.5)\*(z-1);

b=z+0.2;

h=ldiv(b,a,4);

disp(h,"h(n)=")

