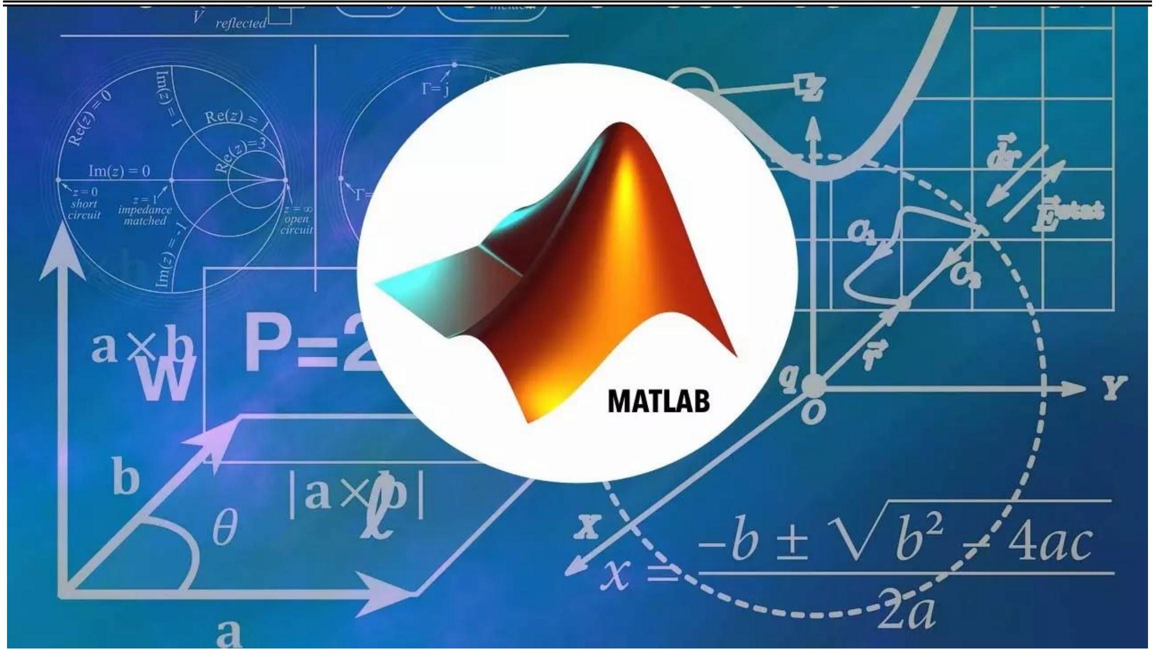
H1038Z Programming for Engineers (II)



>>Semester:2022/23 Autumn Semester

>>Course Name: Programming for Engineers (II)

>>Class: AI Communication Engineering 2101

>>Name: Chen Yiming

>>Sussex ID:254321

>>ZJSU ID:2137020126

Problem 1 [15 Marks]

The natural gas accounts of residents are calculated based on the following rules. First of all, the total charge for each resident consists of a basic service fee and a cost based on consumption. Second, the basic service fee of $10 is charged, no matter how much natural gas is consumed. Third, the consumption-based cost is calculated as follows.

- If 50 units or fewer are consumed, the cost is $0.5 per unit.
- If more than 50 but not more than 100 units are consumed, the cost is $25 for the first 50 units and $1 for every unit in excess of 50.
- If more than 100 units are consumed, the cost is $75 for the first 100 units plus $2 for every unit in excess of 100.

Please write a program (**script M-file**) that

(a) Asks the user to enter into a vector three consumptions that are related to your candidate number: If your **candidate number** is 123456, then the three consumptions are 12, 34, and 56, respectively.

(b) Uses a for loop to calculate and display the total charge for each consumption in two columns: one for the natural gas consumed, and the other for the total charge.

(c) Plots the total charge versus consumption for the consumption amount ranging from 0, 1, 2 to 200 units on a single figure with a correct title and labels.

Answer:

1.1 Matlab script with discussion comments (The script M file also submitted)

Problem_1.m:

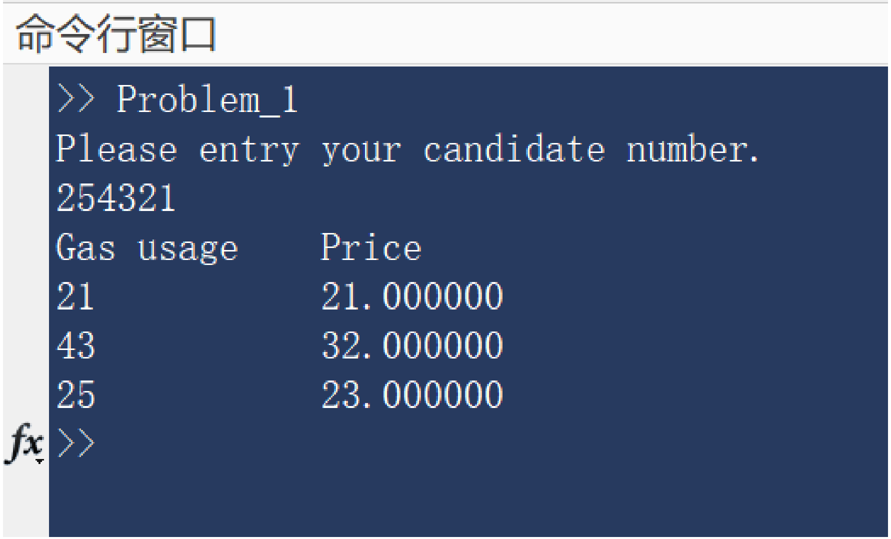
```
n=input('Please entry your candidate number.\n');
%Enter the candidate number and stored in the variable n
%Since we need to extract the data from n and use the mod and divisor
functions,
%we need to convert the input of double n to int N.
N=int32(n);
fprintf("Gas usage \t Price\n" );
for i=1:3
    A=mod(N,100);
    N=idivide(N,100);
    a=Calculating_gas(A);
    fprintf("%d \t\t\t %f\n",A,a);
end

consumption=0:200;
price=0:201:0;
for i=0:200
    price(i+1)=Calculating_gas(consumption(i+1));
end

%The following code is used to make the image
plot(consumption,price,'r');
title("Comparison table of gas consumption and price");
xlabel("Gas consumption");
ylabel("Price");
legend("Price");
grid on;
```

Calculating_gas.m:

```
function x=Calculating_gas(X)
if X<50
    x=10+0.5.*X;
elseif X<100
    x=10+25+(X-50).*1;
else
```

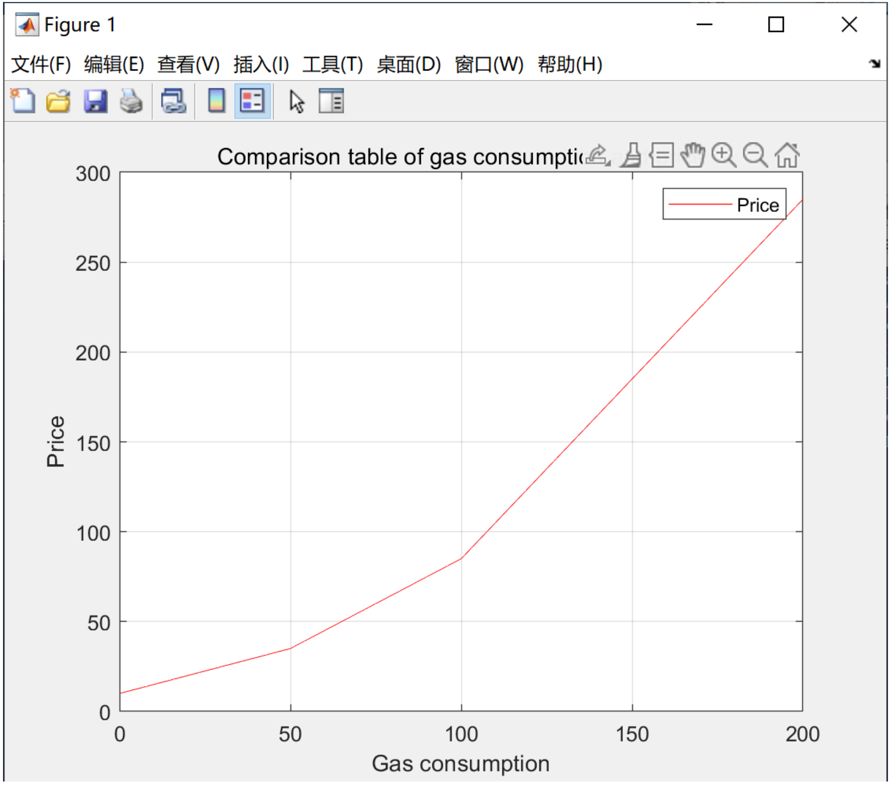
```
    x=10+75+(X-100).*2;
end
end
```

1.2 Screenshots of the evaluation process and results (the displays and figure)



命令行窗口

```
>> Problem_1
Please entry your candidate number.
254321
Gas usage    Price
21           21.000000
43           32.000000
25           23.000000
>>
```



Problem 2 [15 Marks]

A small company offers six annual salary levels (dollars): 20000, 25000, 30000, 35000, 40000, and 50000. The number of employees paid at each level is, respectively: 8, 6, 6, 4, 3, and 3.

Please write a program (**script M-file**) that first uses loops and then uses vectorization without loops to find the following:

(a) The number of employees whose salaries are above and below 32500 dollars, respectively. [Hint: Use logical vectors for vectorization]

(b) The average salary earned by an individual in the company (i.e., the total annual salary divided by the total number of employees).

(c) and (d) Repeat (a) and (b), respectively, by only using vectorization, if the number of employees paid at each level is the corresponding digit of your **candidate number**. For example, the number of employees is, respectively: 1, 2, 3, 4, 5, and 6 if your **candidate number** is 123456.

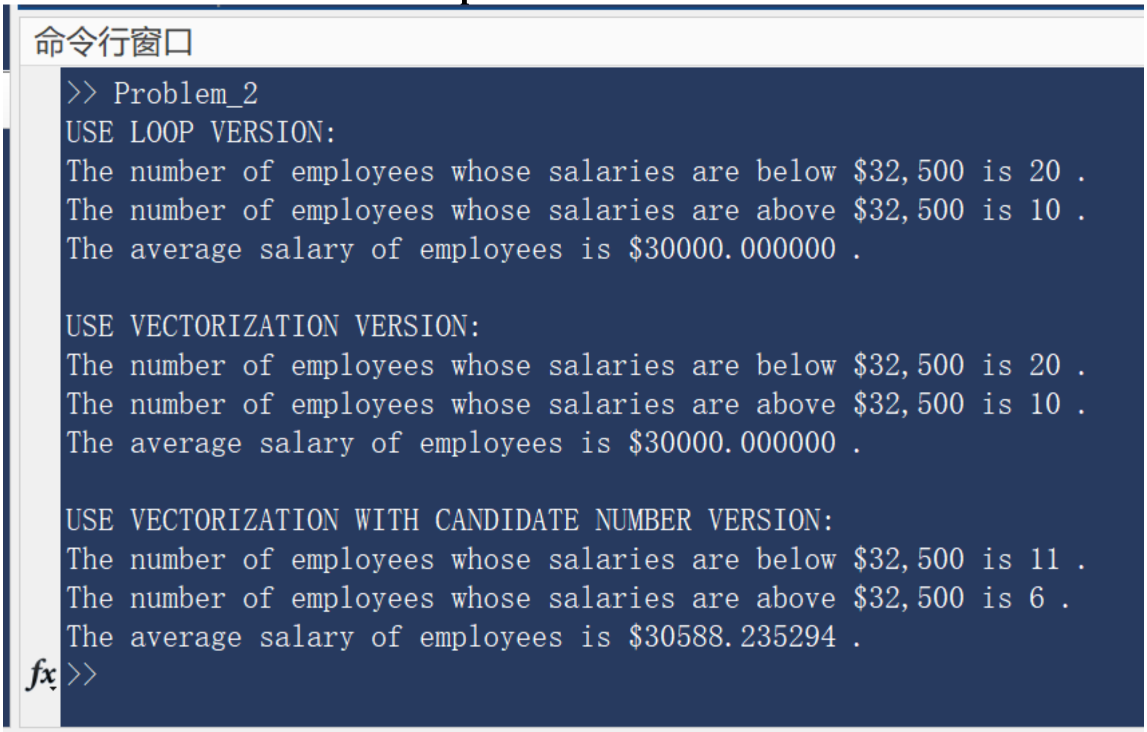
Answer:

2.1 Matlab script with discussion comments (The script M file also submitted)

```
Levels=[20000,25000,30000,35000,40000,50000];
num_employees=[8,6,6,4,3,3];

%for loop version
fprintf("USE LOOP VERSION:\n");
below=0;    %The number of employees below $32,500 recorded in below.
above=0;    %The number of employees above $32,500 recorded in above.
summ=0;     %The total amount of wages issued is recorded in summ.
n_sum=0;    %The total number of people is recorded in n sum.
for i=1:6
    if Levels(i)<32500
        below =below+num_employees(i);
    else
        above =above+num_employees(i);
    end
    summ=summ+Levels(i).*num_employees(i);
    n_sum=n_sum+num_employees(i);
end
ave=summ./n_sum;    %The average salary of employees is recorded in ave.
fprintf("The number of employees whose salaries are below $32,500
is %d .\n",below);
fprintf("The number of employees whose salaries are above $32,500
is %d .\n",above);
fprintf("The average salary of employees is $%f .\n",ave);

fprintf("\n");

%vectorization version
fprintf("USE VECTORIZATION VERSION:\n");
V_below=sum((Levels<32500).*num_employees);%The number of employees below
$32,500 recorded in below.
V_above=sum((Levels>=32500).*num_employees);%The number of employees above
$32,500 recorded in above.
V_ave=sum(Levels.*num_employees)./sum(num_employees);%The average salary of
employees is recorded in ave.
fprintf("The number of employees whose salaries are below $32,500
is %d .\n",V_below);
fprintf("The number of employees whose salaries are above $32,500
is %d .\n",V_above);
fprintf("The average salary of employees is $%f .\n",V_ave);

fprintf("\n");
```

```
%vectorization with candidate nmuber verstion
fprintf("USE VECTORIZATION WITH CANDIDATE NUMBER VERSION:\n");
num_candidate=[2,5,4,3,2,1];
V_below=sum((Levels<32500).*num_candidate);%The number of employees below
$32,500 recorded in below.
V_above=sum((Levels>=32500).*num_candidate);%The number of employees above
$32,500 recorded in above.
V_ave=sum(Levels.*num_candidate)./sum(num_candidate);%The average salary of
employees is recorded in ave.
fprintf("The number of employees whose salaries are below $32,500
is %d .\n",V_below);
fprintf("The number of employees whose salaries are above $32,500
is %d .\n",V_above);
fprintf("The average salary of employees is $%f .\n",V_ave);
```

2.2 Screenshots of the evaluation process and results



```
>> Problem_2
USE LOOP VERSION:
The number of employees whose salaries are below $32,500 is 20 .
The number of employees whose salaries are above $32,500 is 10 .
The average salary of employees is $30000.000000 .

USE VECTORIZATION VERSION:
The number of employees whose salaries are below $32,500 is 20 .
The number of employees whose salaries are above $32,500 is 10 .
The average salary of employees is $30000.000000 .

USE VECTORIZATION WITH CANDIDATE NUMBER VERSION:
The number of employees whose salaries are below $32,500 is 11 .
The number of employees whose salaries are above $32,500 is 6 .
The average salary of employees is $30588.235294 .
>>
```

Problem 3 [15 Marks]

Write a **function M-file** to implement a function that computes the roots of the quadratic equation $a\,x^2 + b\,x + c = 0$. This function has three input arguments a, b, and c which are the coefficients of the quadratic equation. This function also has three output arguments x1, x2, and flag. The x1 and x2 are the two roots (if they exist), which may be equal. The output argument flag must return the following values, according to the number and type of roots:

0: any x is a solution ($a = b = c = 0$);
1: no solution ($a = b = 0$, $c \neq 0$);
2: one real root ($a = 0$, $b \neq 0$, so the root is $-c/b$);
3: two real roots ($b^2 - 4ac \geq 0$);
4: two complex roots ($b^2 - 4ac < 0$);

Write a **script M-file** to test your function by printing the coefficients, flag, and roots on the data below.

(a) 0, 0, 0
(b) 0, 0, 1
(c) 0, 2, 2
(d) 3, 6, 3
(e) 2, 2, -12
(f) 12, 34, 56 (if your **candidate number** is 123456)

Hint: The quadratic formula is $x = \frac{-b\pm\sqrt{b^2-4ac}}{2a}$

Answer:

3.1 Matlab script with comments (The function and script M-files also submitted)

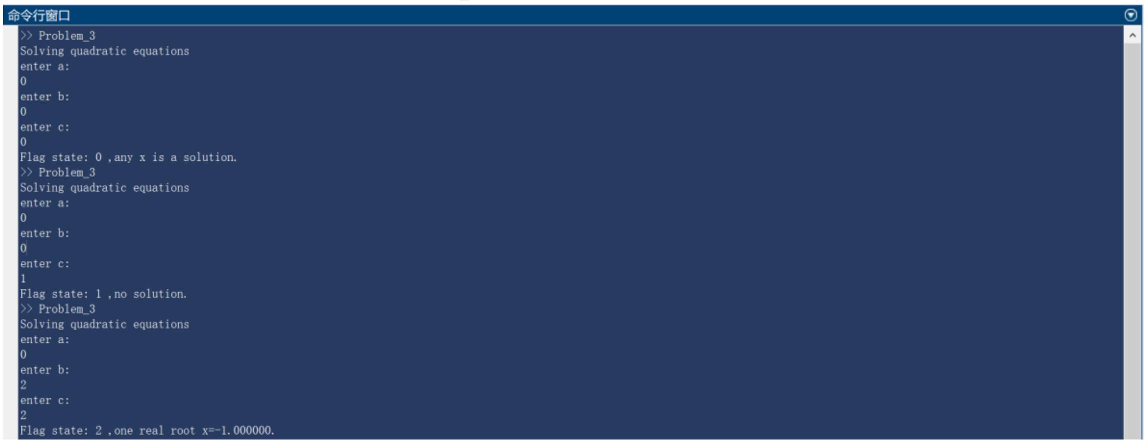
Problem_3.m:

```
clear;
fprintf("Solving quadratic equations\n");
a=input("enter a:\n");
b=input("enter b:\n");
c=input("enter c:\n");
[x1,x2,flag]=slove(a,b,c);
switch flag
    case 0
        fprintf("Flag state: 0 ,any x is a solution.\n");
    case 1
        fprintf("Flag state: 1 ,no solution.\n");
    case 2
        fprintf("Flag state: 2 ,one real root x=%f.\n",x1);
    case 3
        fprintf("Flag state: 3 ,two real roots x1=%f x2=%f.\n",x1,x2);
    case 4
        fprintf("Flag state: 4 , two complex roots.\n");%Output virtual root
        fprintf("x1=\n");
        disp(x1);
        fprintf("x2=\n");
        disp(x2);
end
```

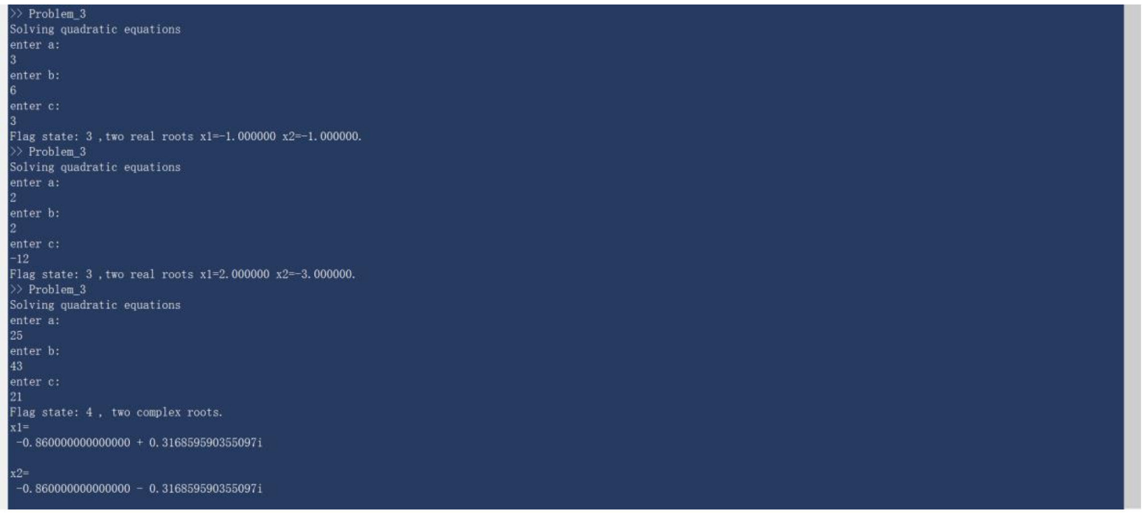
slove.m:

```
function [x1,x2,flag]=slove(a,b,c)
format long;
d=sqrt(b*b-4*a*c);
if d>0
    if a==0
        x1=-c/b;
        x2=-c/b;
        flag=2;
    else
        x1=(-b+d)/(2*a);
        x2=(-b-d)/(2*a);
        flag=3;
    end
elseif d==0
    if a==0&&b==0&&c==0
        flag=0;
        x1=0;
        x2=0;
    elseif a==0&&b==0
        flag=1;
        x1=0;
        x2=0;
    else
        x1=(-b+d)/(2*a);
        x2=(-b-d)/(2*a);
        flag=3;
    end
else
    x1=(-b+d)/(2*a);
    x2=(-b-d)/(2*a);
    flag=4;
end
end
```

3.2 Screenshots of the evaluation process and results



```
>> Problem_3
Solving quadratic equations
enter a:
0
enter b:
0
enter c:
0
Flag state: 0 ,any x is a solution.
>> Problem_3
Solving quadratic equations
enter a:
0
enter b:
0
enter c:
1
Flag state: 1 ,no solution.
>> Problem_3
Solving quadratic equations
enter a:
0
enter b:
2
enter c:
2
Flag state: 2 ,one real root x=-1.000000.
```



```
>> Problem_3
Solving quadratic equations
enter a:
3
enter b:
6
enter c:
3
Flag state: 3 ,two real roots x1=-1.000000 x2=-1.000000.
>> Problem_3
Solving quadratic equations
enter a:
2
enter b:
2
enter c:
-12
Flag state: 3 ,two real roots x1=2.000000 x2=-3.000000.
>> Problem_3
Solving quadratic equations
enter a:
25
enter b:
43
enter c:
21
Flag state: 4 , two complex roots.
x1=
 -0.860000000000000 + 0.316859590355097i

x2=
 -0.860000000000000 - 0.316859590355097i
```

Problem 4 [15 Marks]

Write a **script M-file** to solve the following system of linear equations:

$$2x - y + z = 4$$

$$x + y + z = 3$$

$$3x - y - z = 1$$

by expressing the equations in matrix form $\boldsymbol{Ax} = \boldsymbol{b}$ and using the left division operator.

(a) The script first asks the user to input the coefficient matrix $\boldsymbol{A}$ and vector $\boldsymbol{b}$.

(b) The script then checks whether the determinant of $\boldsymbol{A}$ is not zero and whether the dimensions of $\boldsymbol{A}$ and $\boldsymbol{b}$ are compatible before calculating the roots.

(c) The script displays the values of *x*, *y* and *z*.

(d) After doing (a)(b)(c), the script interchanges columns 1 and 2 of matrix $\boldsymbol{A}$ and replaces the vector $\boldsymbol{b}$ according to your **candidate number.** If your **candidate number** is 123456, then the elements of $\boldsymbol{b}$ should be 12, 34, and 56, respectively. Display the revised matrix $\boldsymbol{A}$ and vector $\boldsymbol{b}$.

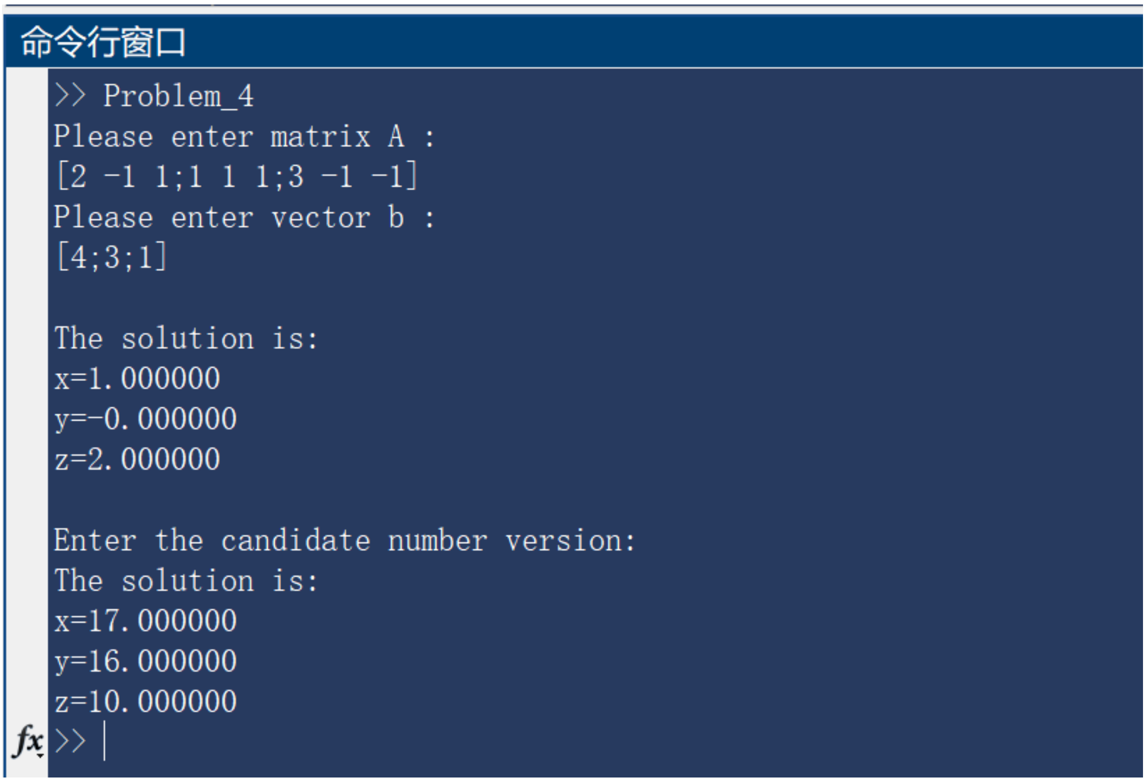
(e) The script finally displays the values of *x*, *y* and *z* after doing (d).

Answer：

4.1 Matlab script with discussion comments (The script M file also submitted)

```
A=input("Please enter matrix A :\n");
b=input("Please enter vector b :\n");
D=det(A);
s_A=size(A);
s_b=size(b);
if D ~=0 && s_A(2)==s_b(1)
    out=A\b;
end
fprintf("\n");
fprintf("The solution is:\n");
fprintf("x=%f\ny=%f\nz=%f\n",out(1),out(2),out(3));

fprintf("\n");
fprintf("Enter the candidate number version:\n");
A(:,[1 2])=A(:,[2 1]);
b=[25;43;21];
out=A\b;
fprintf("The solution is:\n");
fprintf("x=%f\ny=%f\nz=%f\n",out(1),out(2),out(3));
```

4.2 Screenshots of evaluation process and results



```
>> Problem_4
Please enter matrix A :
[2 -1 1;1 1 1;3 -1 -1]
Please enter vector b :
[4;3;1]

The solution is:
x=1.000000
y=-0.000000
z=2.000000

Enter the candidate number version:
The solution is:
x=17.000000
y=16.000000
z=10.000000
>> 
```

Problem 5 [15 Marks]

Please solve the following system of differential equations where ω is your **candidate number** and *t* is time.

$$\frac{dx}{dt} = 0.5xy - x - e^{\frac{-\omega}{10000}t}$$

$$\frac{dy}{dt} = -0.4xy + z + 2e^{\frac{-\omega}{20000}t}$$

$$\frac{dz}{dt} = -0.3x + y + 3e^{\frac{-\omega}{30000}t}$$

(a) Write a **script M-file** that uses the **ode45** function to solve the equations with initial values *x*(0)=0.1, *y*(0)=0.2, and *z*(0)=0.3 for $0 \le t \le 20$ seconds.

(b) Write a **function M-file** to define the differential function.

(c) The script should display the maximum values of *x*(*t*), *y*(*t*), and *z*(*t*) with their corresponding time *t*.

(d) The script should also plot *x*(*t*), *y*(*t*), and *z*(*t*) on a single figure with correct labels and legends.

Answer:

5.1 Matlab script with comments (The function and script M-files also submitted)

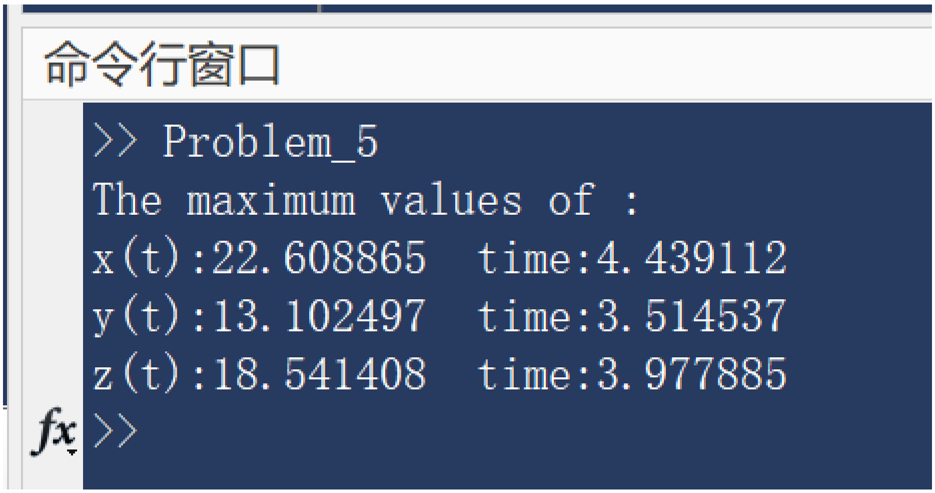
Problem_5.m:

```
tspan=[0 20];
u0=[0.1;0.2;0.3];
[t,u] = ode45(@odefun,tspan,u0);
[U,T]=max(u);
fprintf("The maximum values of :\n");
fprintf("x(t):%f\ttime:%f\n",U(1),t(T(1)));
fprintf("y(t):%f\ttime:%f\n",U(2),t(T(2)));
fprintf("z(t):%f\ttime:%f\n",U(3),t(T(3)));
plot(t,u);
grid on;
title('Value of x(t)/y(t)/z(t) with t');
xlabel('t');
ylabel('u(t)');
legend('x(t)','y(t)','z(t)');
```

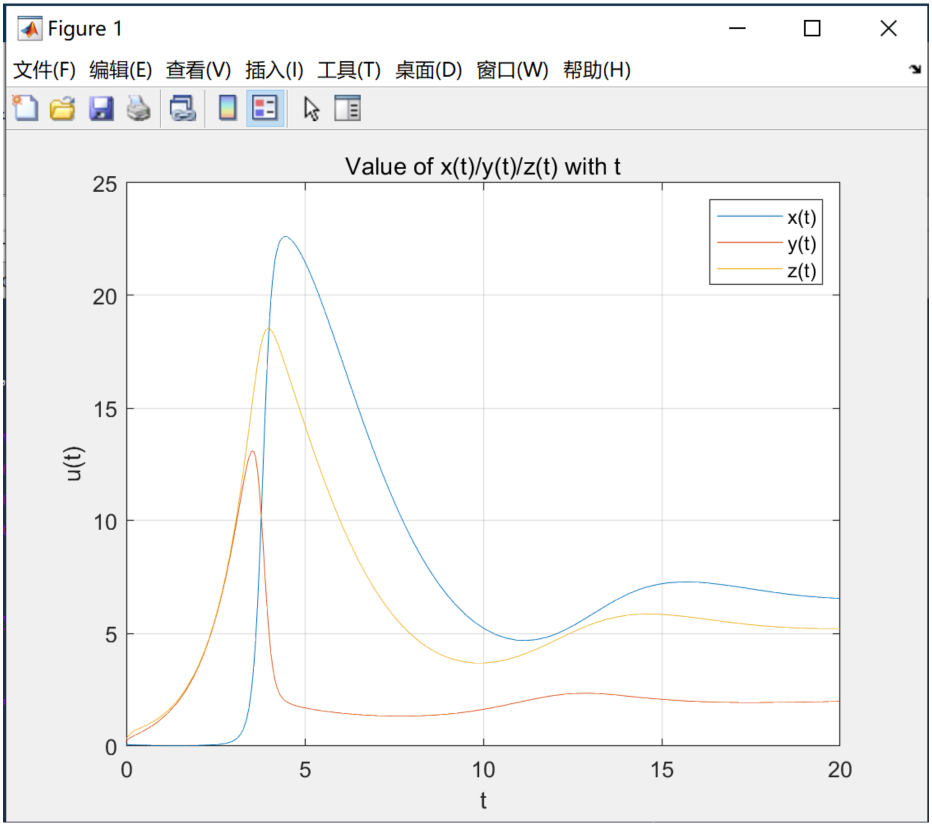
odefun.m:

```
function dudt=odefun(t,u)
Omega=254321;
dudt=[0.5.*u(1).*u(2)-u(1)-exp(-Omega/10000*t);
    -0.4.*u(1).*u(2)+u(3)+2.*exp(-Omega/20000*t);
    -0.3.*u(1)+u(2)+3.*exp(-Omega/30000*t)];
end
```

5.2 Screenshots of evaluation process and results



```
>> Problem_5
The maximum values of :
x(t):22.608865   time:4.439112
y(t):13.102497   time:3.514537
z(t):18.541408   time:3.977885
>>
```



Problem 6 [15 Marks]

A process adds signals A and B together to generate an output signal C.

$$A = 2sin\left(\frac{\gamma}{30000}t\right) + 3\cos(t)$$

$$B = 20e^{-0.5t}$$

Where γ is your **candidate number**, and t is time

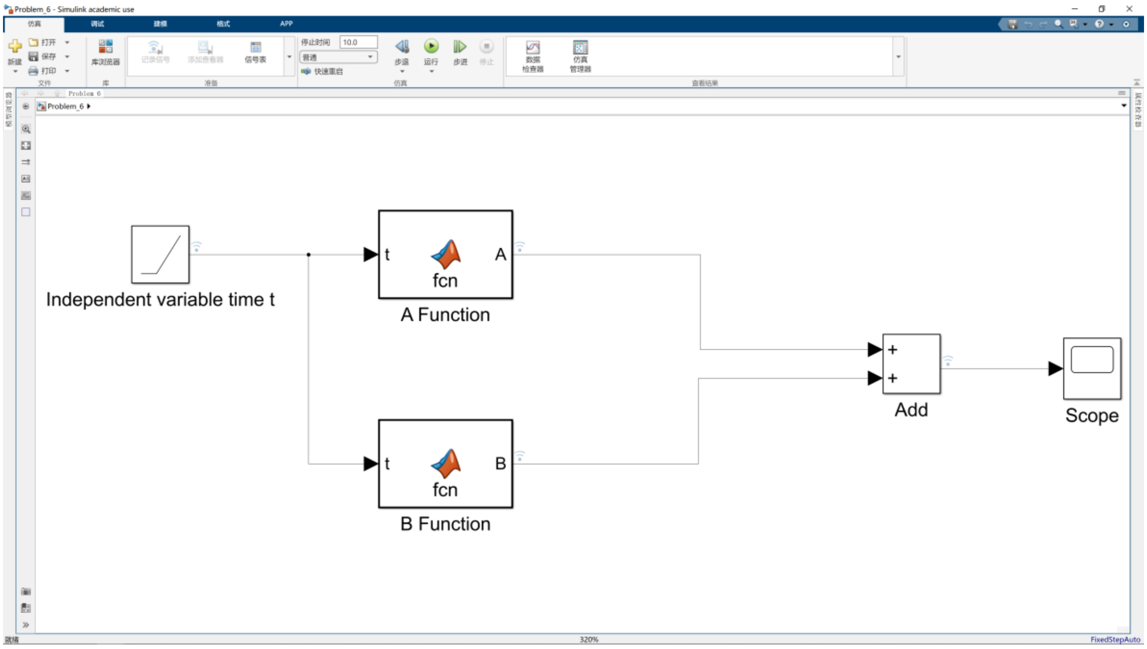
(a) Please design a **SimuLink** model (an **SLX file**) to describe this process for $0 \leq t \leq 20$ seconds, the model must be shown in the report.

(b) All signals in the system must be displayed on scopes and shown in the report. Please set proper styles for the scopes.

(c) Each block (including the scopes) should have a proper name. The Simulink model and the properties of blocks (excluding the scopes) must be shown in the report.

Answer:

6.1 Screenshot of the SimuLink model (The SLX file also submitted)

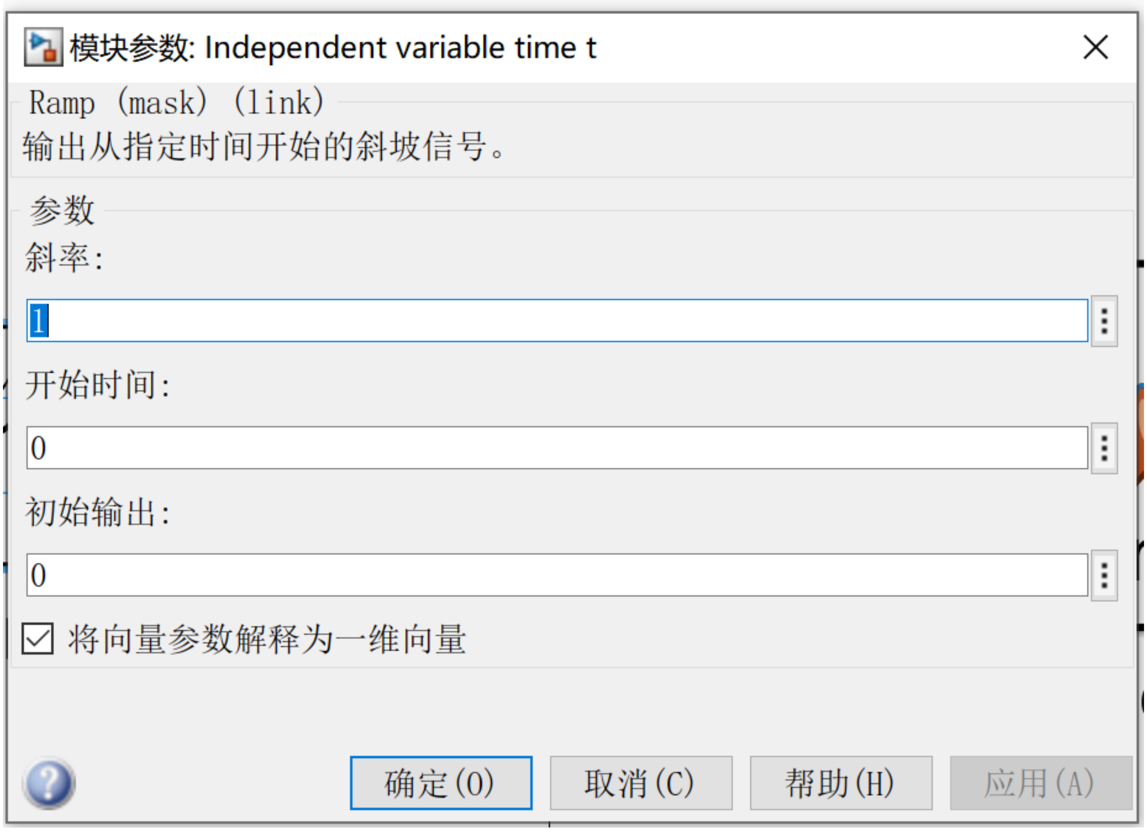


Problem_6 - Simulink academic use

停止时间 10.0

普通

6.2 Screenshots of the properties (parameters) of blocks



模块参数: Independent variable time t

Ramp (mask) (link)

输出从指定时间开始的斜坡信号。

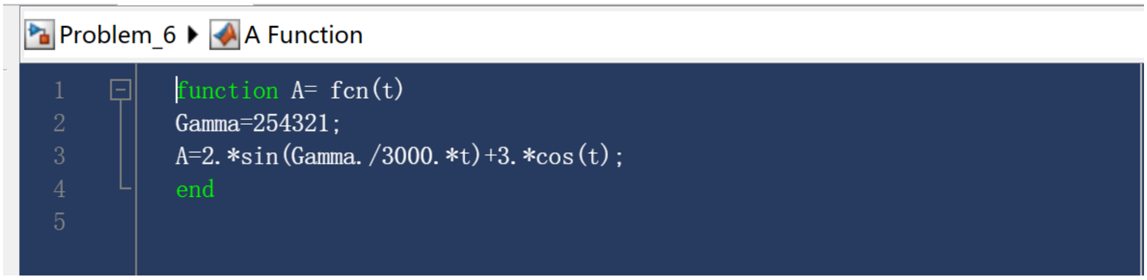
参数

斜率: 1

开始时间: 0

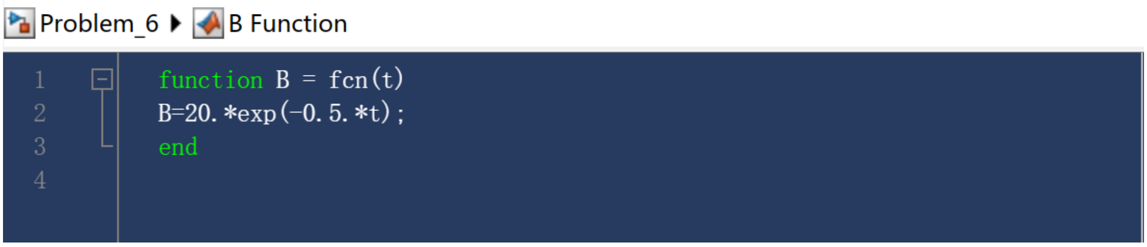
初始输出: 0

☑ 将向量参数解释为一维向量



Problem_6 ▸ A Function

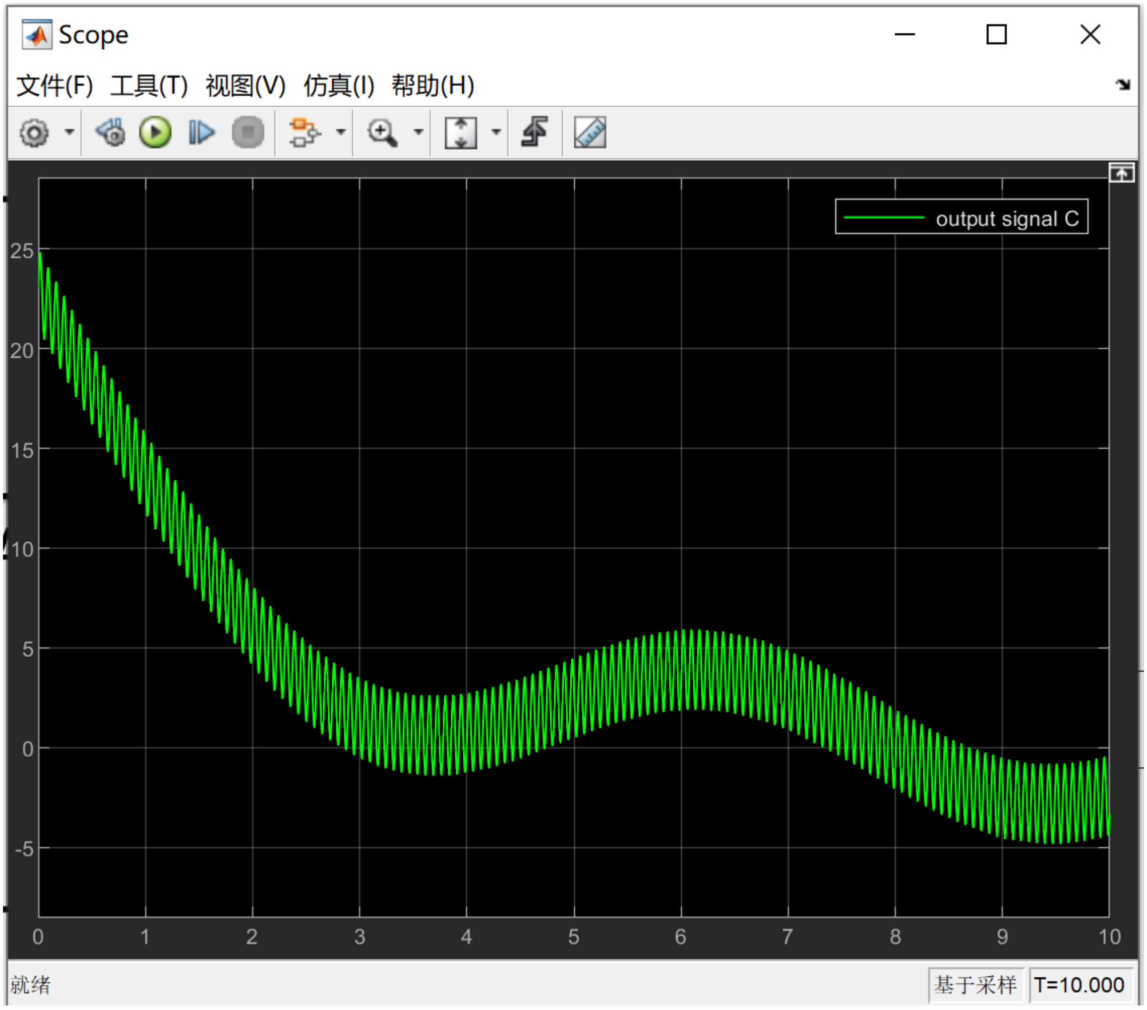
```
function A= fcn(t)
Gamma=254321;
A=2.*sin(Gamma./3000.*t)+3.*cos(t);
end
```

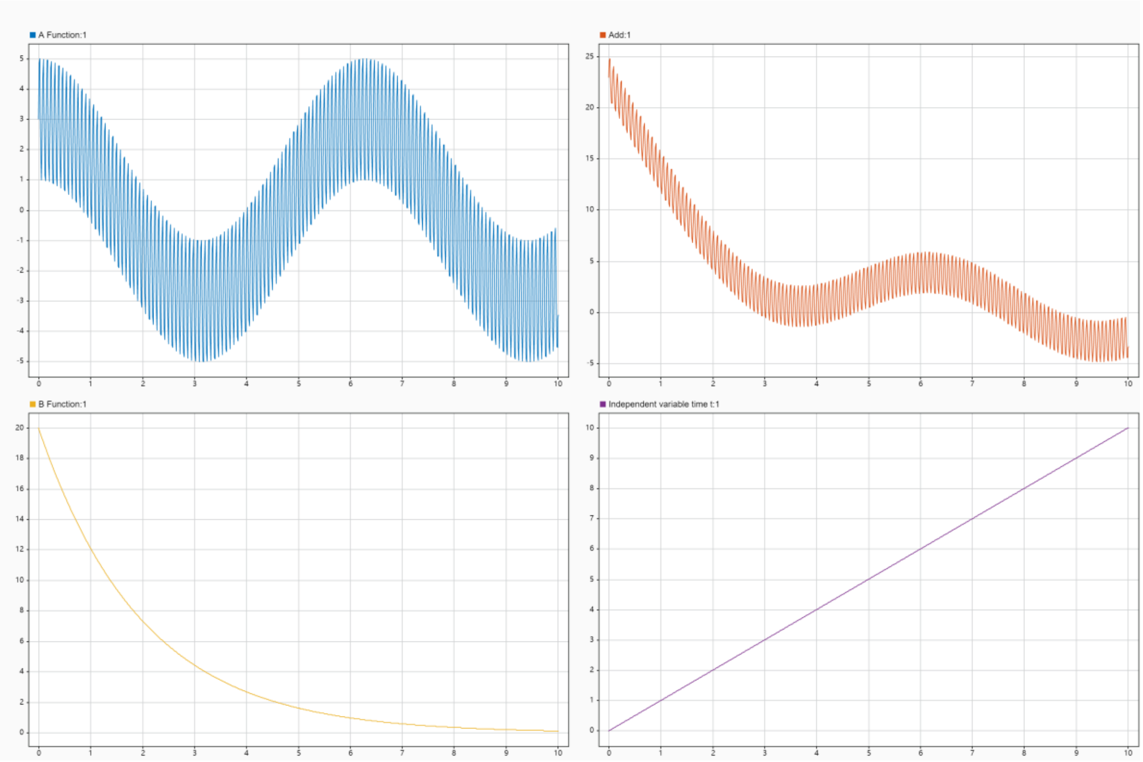


Problem_6 ▸ B Function

```
function B = fcn(t)
B=20.*exp(-0.5.*t);
end
```

6.3 Screenshots of the scopes on all the signals


Scope
文件(F)
工具(T)
视图(V)
仿真(I)
帮助(H)
output signal C
25
20
15
10
5
0
-5
0
1
2
3
4
5
6
7
8
9
10
就绪
基于采样
T=10.000


A Function:1
Add:1
B Function:1
Independent variable time t:1