

Báo cáo thực hành

Bài thực hành số 1

Môn: Truyền dữ liệu

Lớp: NT105.K21.MMCL

Họ tên sinh viên: Thái Công Sinh

MSSV: 18521340

1) Sử dụng lệnh help hoặc lookfor để tìm kiếm thông tin cho các câu hỏi sau:

- Hãy tìm lệnh thể hiện phép toán hàm cosin .

```
>> lookfor cosin
acos                  - Inverse cosine, result in radians.
acosd                 - Inverse cosine, result in degrees.
acosh                 - Inverse hyperbolic cosine.
cos                   - Cosine of argument in radians.
cosd                 - Cosine of argument in degrees.
cosh                 - Hyperbolic cosine.
s1sinicos            - This is a private mask helper file for sine and cosine blocks in
s1sinicoslut          - This is a private mask helper file for sine and cosine blocks in
angle2dcm             - Create direction cosine matrix from rotation angles.
dcm2alphabeta         - Convert direction cosine matrix to angle of attack and sideslip angle.
dcm2angle              - Create rotation angles from direction cosine matrix.
dcm2latlon             - Convert direction cosine matrix to geodetic latitude and longitude.
dcm2quat               - Convert direction cosine matrix to quaternion.
dcm2rod                - Convert direction cosine matrix to Rodrigues vector.
dcmbody2wind           - Convert angle of attack and sideslip angle to direction cosine matrix.
dcmecef2ned            - Convert geodetic latitude and longitude to direction cosine matrix.
quat2dcm               - Convert quaternion to direction cosine matrix.
rod2dcm                - Convert Rodrigues vector to direction cosine matrix.
commblkagctransform   - Raised cosine filter blocks transform function
commblkrcfiltrx        - Raised cosine Receiver FIR filter blocks helper function.
commblkrcfilttx        - Raised cosine filter blocks transform function
commblkrcfilttx        - Raised cosine Transmit filter blocks helper function.
rcfiltgaincompat       - Backwards compatible raised cosine filter gain
rcosfir               - Design a raised cosine FIR filter.
rcosflt                - Filter the input signal using a raised cosine filter.
rcosfir                - Design a raised cosine IIR filter.
rcosflt                - Design raised cosine filter.
```

- Tìm thông tin về hàm logarithms

```
>> lookfor logarithms
logarithms not found.
>> help logarithms

logarithms not found.
```

2) Cho hai số phức bất kỳ, ví dụ $-3 + 2i$ và $5 - 7i$. Hãy thực hiện các phép toán để cộng, trừ, nhân và chia hai số phức này với nhau sao cho đơn giản, nhanh nhất

>> a=-3+2i

a =

$$-3.0000 + 2.0000i$$

>> b=5-7i

b =

$$5.0000 - 7.0000i$$

>> a+b

ans =

$$2.0000 - 5.0000i$$

>> a-b

ans =

$$-8.0000 + 9.0000i$$

>> a*b

ans =

$$-1.0000 + 31.0000i$$

>> a/b

ans =

$$-0.3919 - 0.1486i$$

3) Hãy thực hiện các bài tập sau đây:

- Tạo một vector bao gồm những số lẻ trong khoảng từ 21 đến 47.

>> A=[21:2:47]

- Cho $x = [4 \ 5 \ 9 \ 6]$.
- Trừ đi 3 ở mỗi thành phần của vector.

>> x=[4 5 9 6]

x =

4 5 9 6

>> x=x-3

x =

1 2 6 3

- Cộng 11 vào các thành phần có vị trí lẻ

>> x(1:2:end)=x(1:2:end)+11

x =

12 2 17 3

4) Hãy xóa tất cả các biến (lệnh clear). Định nghĩa ma trận A = [1:4; 5:8; 1 1 1 1].

Hãy thực thi và kiểm tra kết quả của các phép tính sau:

>> A=[1:4;5:8;1 1 1 1]

A =

1 2 3 4

5 6 7 8

1 1 1 1

- $x = A(:, 3)$

>> x=A(:,3)

x =

3

7

1

- $y = A(3 : 3, 1 : 4)$

>> y=A(3:3,1:4)

y =

1 1 1 1

- $B = A(1 : 3, 2 : 2)$

>> B=A(1:3,2:2)

B =

2

6

1

- $A = [A; 2 \ 1 \ 7 \ 7; 7 \ 7 \ 4 \ 5]$

>> A=[A;2 1 7 7;7 7 4 5]

A =

1 2 3 4

5 6 7 8

1 1 1 1

2 1 7 7

7 7 4 5

- $A(1, 1) = 9 + A(2, 3)$

>> A(1,1)=9+A(2,3)

A =

16 2 3 4

5 6 7 8

1 1 1 1

2 1 7 7

7 7 4 5

- $C = A([1, 3], 2)$

>> C=A([1,3],2)

C =

2

1

- $A(2 : 3, 1 : 3) = [0 \ 0 \ 0; 0 \ 0 \ 0]$

>> A(2:3,1:3)=[0 0 0;0 0 0]

A =

16 2 3 4

0 0 0 8

0 0 0 1

2 1 7 7

7 7 4 5

- $D = A([2, 3, 5], [1, 3, 4])$

>> D=A([2,3,5],[1,3,4])

D =

0 0 8

0 0 1

7 4 5

- $D(2, :) = []$

>> D(2,:)=[]

D =

0 0 8

7 4 5

5) Cho ma trận $A = [2 \ 4 \ 1; 6 \ 7 \ 2; 3 \ 5 \ 9]$. Thực thi các phép toán sau đối với A:

- Gán hàng thứ 1 của A cho vector x

>> A=[2 4 1; 6 7 2; 3 5 9]

A =

2 4 1

6 7 2

3 5 9

>> x=A(1,:)

x =

2 4 1

- Gán 2 hàng cuối của A cho y.

```
>> y=A(3,:)
```

```
y =
```

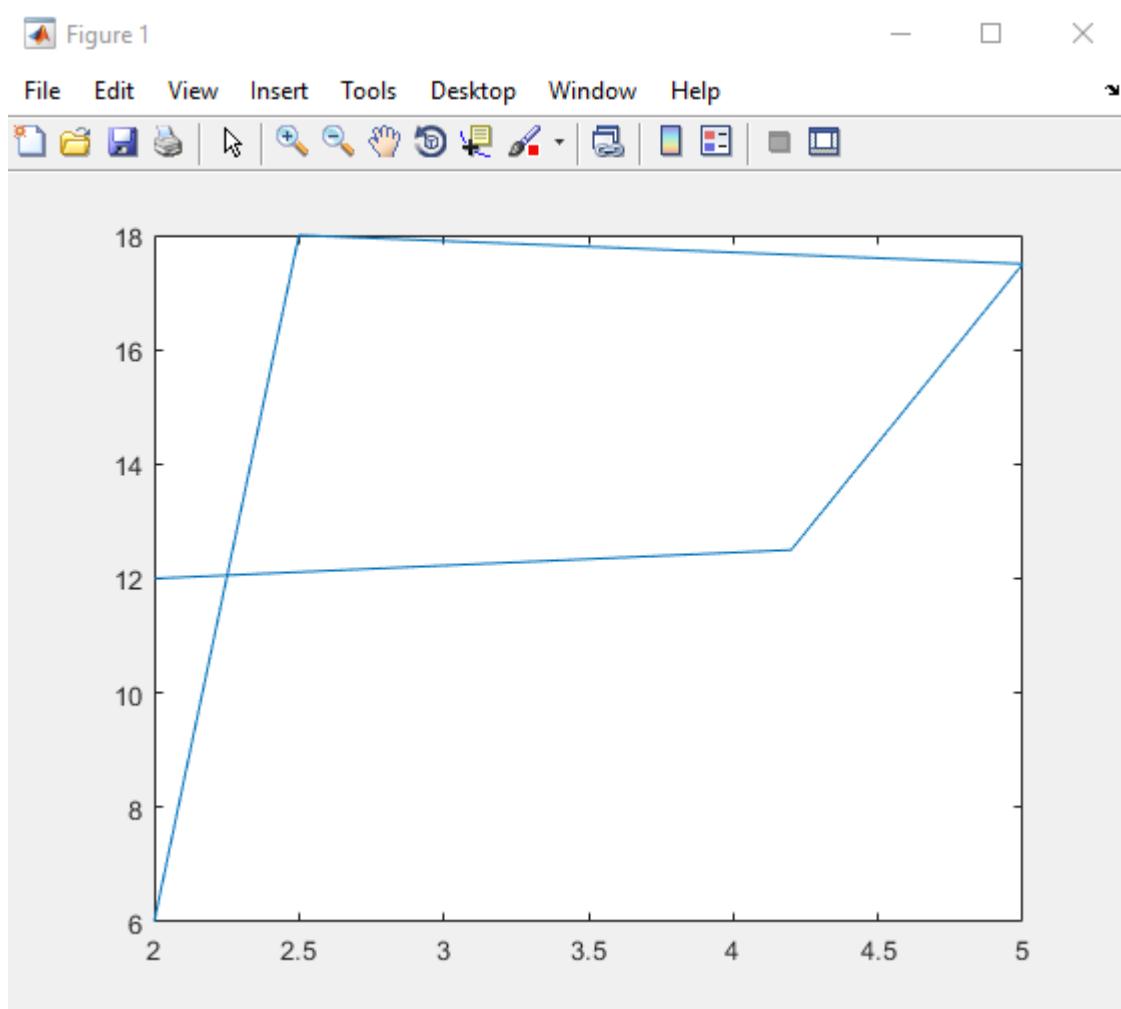
```
3 5 9
```

6) Vẽ một đường bằng nét gạch ngắn nối các điểm sau lại với nhau: (2, 6), (2.5, 18),

(5, 17.5), (4.2, 12.5) và (2, 12). >> x=[2 2.5 5 4.2 2]

```
>>y=[6 18 17.5 12.5 12]
```

```
>> plot(x,y)
```



7) Vẽ đồ thị của hàm $y = \sin(x)$ trong khoảng $0 < x < 30$ thêm tiêu đề và mô tả của các trục vào đồ thị.

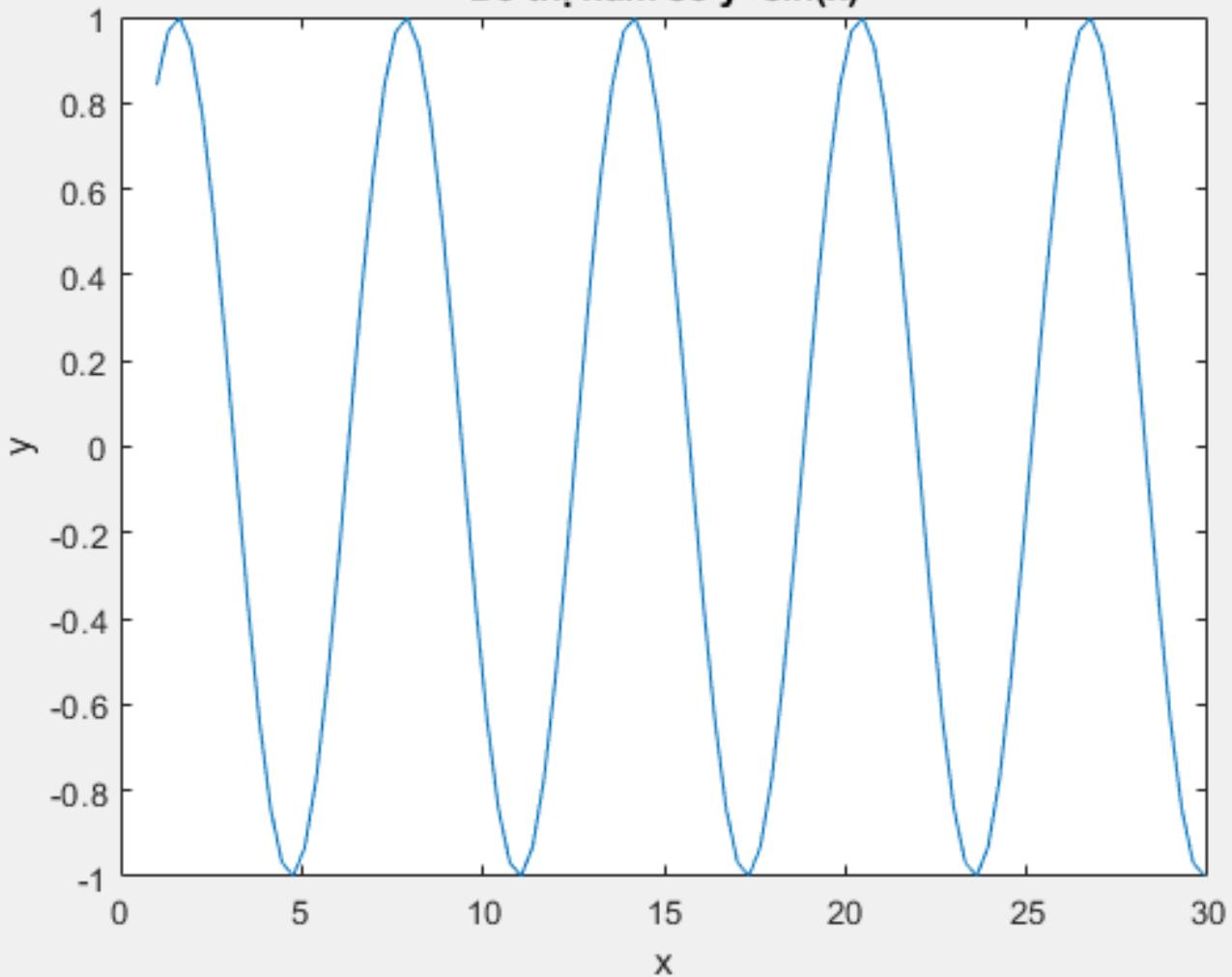
```
>> x=[1:pi/10:30];  
>> y=sin(x);  
>> plot(x,y);  
>> xlabel('x');  
>> ylabel('y');  
>> title('Đồ thị hàm số y=sin(x)');
```

Figure 1

File Edit View Insert Tools Desktop Window Help



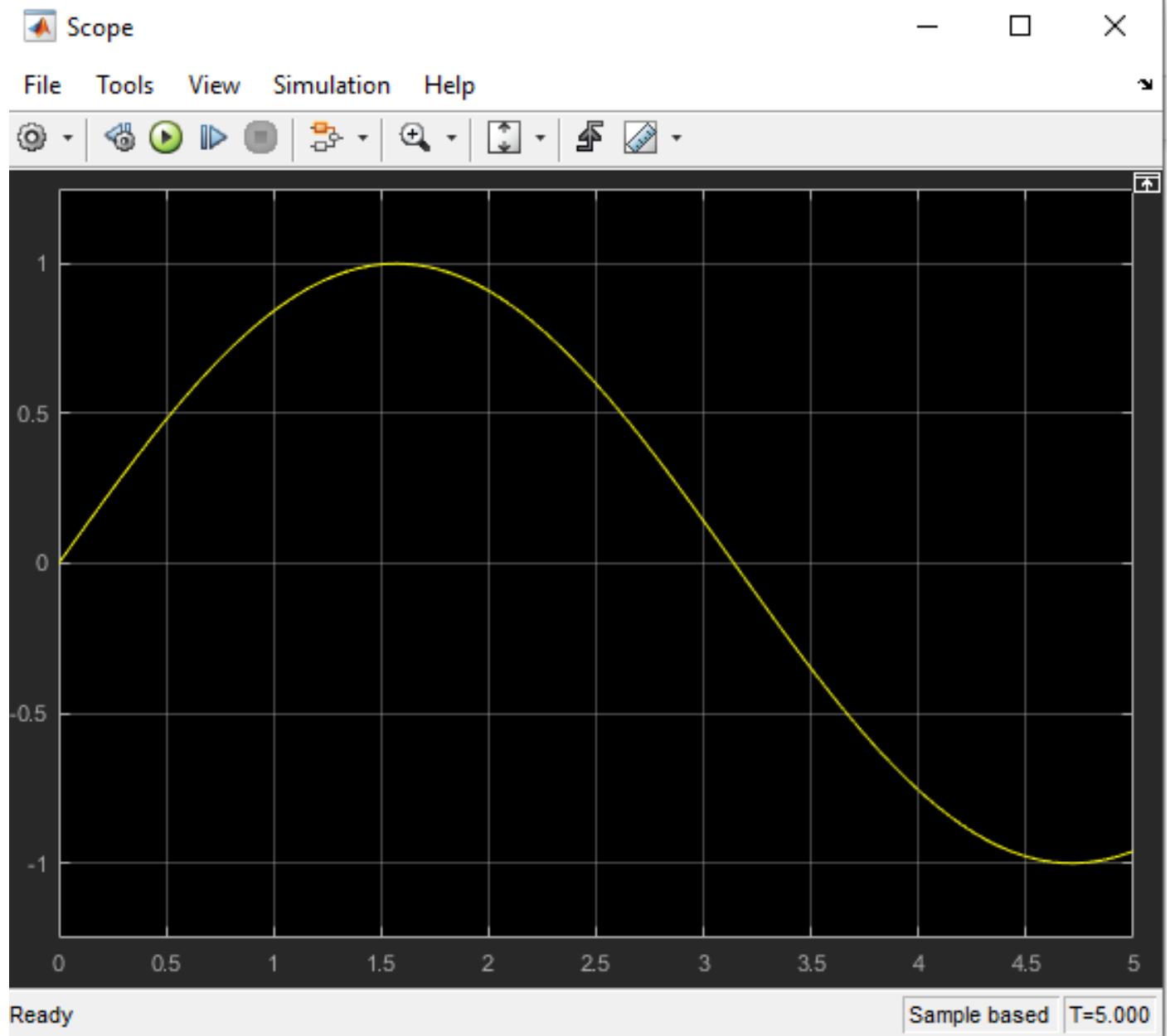
Đồ thị hàm số $y=\sin(x)$



Bài 8:

Vẽ 1 tín hiệu sin và biểu diễn bằng scope trong 5s, thay đổi số mẫu (sample) và

nhận xét:



Giá trị sample time thể hiện số lần vẽ sau (t) giây, khi giá trị này càng nhỏ thì số lần vẽ càng nhiều-> hình vẽ càng chi tiết.

Vẽ 3 tín hiệu sin khác nhau và biểu diễn trên cùng scope trong 5. (gợi ý: thêm số

lượng input trong setup scope).

