Date: 1/10/2021

Experiment 7

Aim: To work with MATLAB strings and additional data plots.

Apparatus: MATLAB Software

Objective:

1. To learn about different string operations.
2. To be aware of additional features of the simple two-dimensional plots.
3. To learn how to plot in three dimensions.

Problems:

Q-1. Write a program that accepts an input string from the user and determines the how many times a user-specified character appears within the string. (*Hint:* Look up the 's' option of the input function using the MATLAB Help Browser.)

Code:

```
clc;
clear all;
close all;

str = input('Enter String : ','s');

f = input('Enter character to be find : ','s');
count = 0;

for i = str
    if i == f
        count = count + 1;
    end
end

count
```

Output:

```
Enter String :
This is Matlab
Enter character to be find :
i

count =

     2

>>
```

Q-2. Modify the previous program so that it determines how many times a user-specified character appears within the string without regard to the case of the character.

Code:

```
clc;
clear all;
close all;

str = input('Enter String : ','s');

f = input('Enter character to be find : ','s');
count = 0;

for i = str
    if strcmpi(f,i)
        count = count + 1;
    end
end

count
```

Output:

```
Enter String :
Hello how are you
Enter character to be find :
h

count =

     2

>>
```

Q-3. Write a program that accepts a string from a user with the input function, chops that string into a series of tokens, sorts the tokens into ascending order, and prints them out.

Code:

```
clc;
clear all;
close all;

str = input('Enter String : ','s');

sort(str)
```

Output:

```
Enter String :
harsh

ans =

    'ahhrs'

>>
```

Q-4. MATLAB includes functions upper and lower, which shift a string to upper case and lower case respectively. Create a new function called caps, which capitalizes the first letter in each word and forces all other letters to be lower case. (*Hint:* Take advantage of functions upper, lower, and strtok.)

Code:

```
clc;
clear all;
close all;

str = input('Enter String : ','s');
capitalize(str)


function res = capitalize(str)
   res = lower(str);
   res(1,1) = upper(res(1,1));
end
```

Output:

```
Enter String :
hello

ans =

    'Hello'

>>
```

Q-5. Write a function that accepts a character string and returns a logical array with true values corresponding to each printable character that is *not* alphanumeric or whitespace (for example, $, %, #, etc.) and false values everywhere else.

Code:

```
clc;
clear all;
close all;

a = ' ED#$22cD?? '
is_alpha = [];
count = 1;
for i=a
   if regexp(i,'^[A-Za-z0-9]+$')
       is_alpha(count)=1;
    else
        is_alpha(count)=0;
    end
    count = count + 1;
end
is_alpha
```

Output:

```
a =

    ' ED#$22cD?? '


is_alpha =

     0     1     1     0     0     1     1     1     1     0     0     0
```

Q-6. Write a function that accepts a character string and returns a logical array with true values corresponding to each vowel and false values everywhere else. Be sure that the function works properly for both lowercase and uppercase characters.

Code:

```
clc;
clear all;
close all;

str = 'hello how are you'
a = str == 'a';
e = str == 'e';
i = str == 'i';
o = str == 'o';
u = str == 'u';


final = bitor(a,e);
final = bitor(final,i);
final  = bitor(final,o);
final  = bitor(final,u);

final
```

Output:

```
str =

    'hello how are you'


final =

  1×17 logical array

   0   1   0   0   1   0   0   1   0   0   1   0   1   0   0   1   1
```

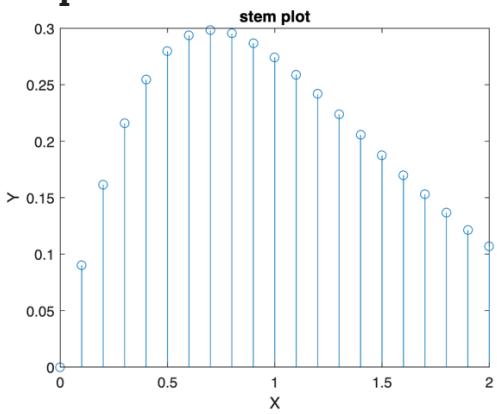
Q-7. Plot the function $y = e^{-x} \sin x$ for x between 0 and 2 in steps of 0.1. Create the following plot types: *(a)* stem plot; *(b)* stair plot; *(c)* bar plot; *(d)* compass plot. Be sure to include titles and axis labels on all plots.

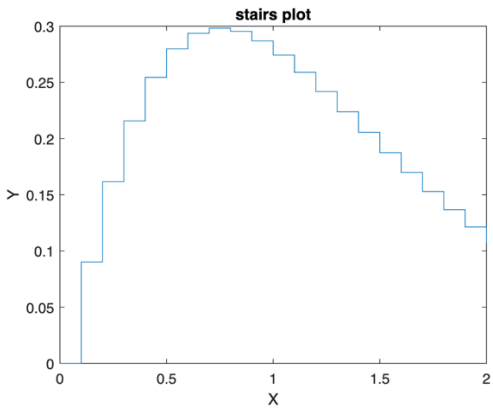
Code:

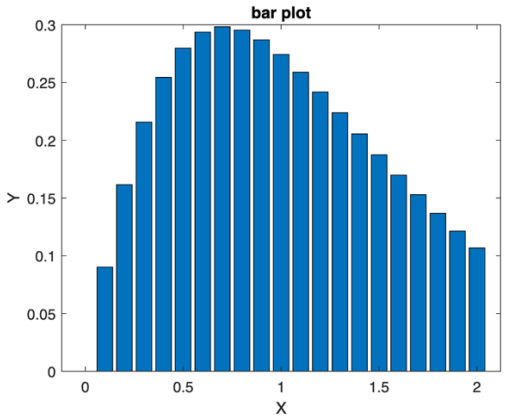
```
clc;
clear all;
close all;


x = 0:0.1:2;
y = exp(-x) .* sin(sin(x));

figure(1);
stem(x,y);
title('stem plot');
xlabel('X');
ylabel('Y');

figure(2);
stairs(x,y);
title('stairs plot');
xlabel('X');
ylabel('Y');

figure(3);
bar(x,y);
title('bar plot');
xlabel('X');
ylabel('Y');
```

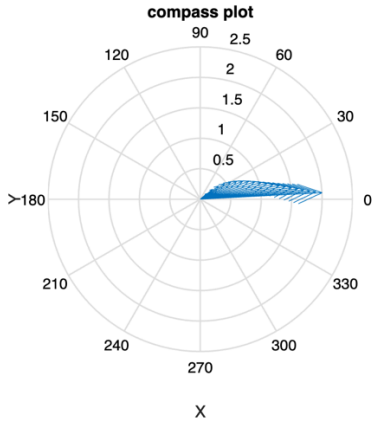
```
figure(4);
compass(x,y);
title('compass plot');
xlabel('X');
ylabel('Y');
```

Output:







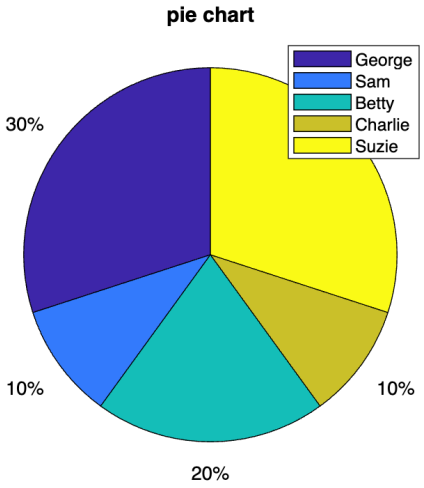


Q-8. Suppose that George, Sam, Betty, Charlie, and Suzie contributed $15, $5,$10, $5, and $15, respectively, to a colleague's going-away present. Create a pie chart of their contributions. What percentage of the cost was paid by Sam?

Code:

```
clc;
clear all;
close all;

g = 15;
s = 5;
b = 10;
c = 5;
su = 15;

labels = {'George','Sam','Betty','Charlie','Suzie'};

figure(1);
pie([g,s,b,c,su]);
legend(labels)
title('pie chart');
```

Output:

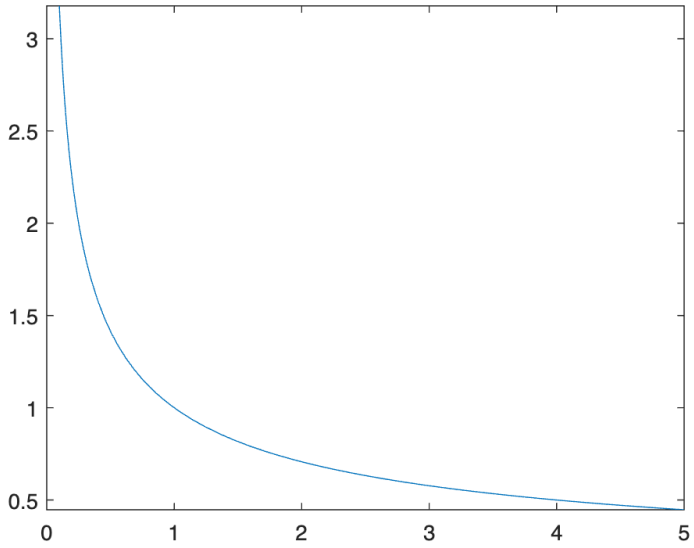


Q-9. Plot the function f(x)=1/$\sqrt{x}$ over the range 0.1<=x<=10 using the function fplot. Be sure to label your plot properly.

Code:

```
clc;
clear all;
close all;

t = 0.1:10;
fplot(@(t) 1/sqrt(t))
```

Output:



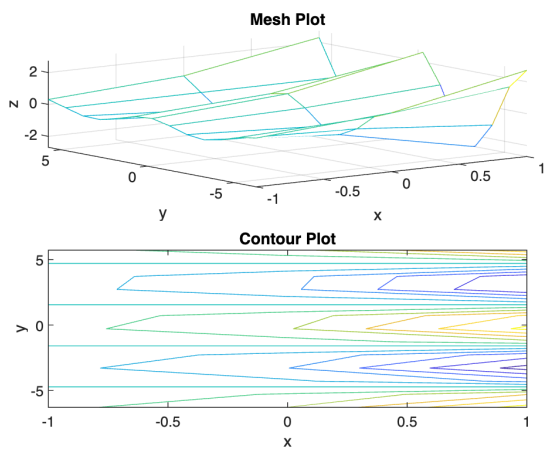
Q-10. Create a mesh, surface plot, and contour plot of the function $z = e^{x+iy}$ for the interval -1<=x<=1 and -2π<=y<=2π. In each case, plot the real part of z versus x and y.

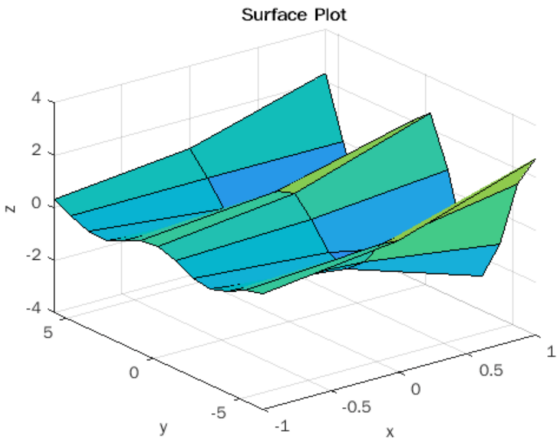
Code:

```
clc;
clear all;
close all;


x = -1:1;
y = -2*pi:2*pi;
[A, B] = meshgrid(x, y);
z = exp(A + i .* B);
z_real = real(z);

figure(1);

subplot(2, 1, 1);
mesh(A, B, z_real);
title("Mesh Plot");
xlabel("x");
ylabel("y");
zlabel("z");

subplot(2, 1, 2);
contour(A, B, z_real);
title("Contour Plot");
xlabel("x");
ylabel("y");
zlabel("z");
```

```
figure(2);
surf(A, B, z_real);
title("Surface Plot");
xlabel("x");
ylabel("y");
zlabel("z");
```

Output:





Conclusion:

In this practical, I learnt how to use different functions related to String, how these function works and also learnt about additional data plots provided from MATLAB.