**Date: 21-08-2021**

**Experiment 3**

**Aim:** To work with branching statements.

**Apparatus:** MATLAB Software

**Objective:** To learn how to control the order in which statements are executed using if-else and switch-case.

**Problems:**

**Q-1.** The following statements are intended to alert a user to dangerously high

oral thermometer readings (values are in degrees Fahrenheit). Are they correct

or incorrect? If they are incorrect, explain why and correct them.

if temp < 97.5

disp('Temperature below normal');

elseif temp > 97.5

disp('Temperature normal');

elseif temp > 99.5

disp('Temperature slightly high');

elseif temp > 103.0

disp('Temperature dangerously high');

end

**Reason:**

As an when the input exceeds 97.5, it enters into second condition and skips the rest conditions.

**Code:**

clc;

clear all;

close all;

s = 'Enter temp value';

temp = input(s);

if temp <= 97.5

disp("Below Normal");

elseif temp > 97.5 && temp < 99.5

disp("Normal");

elseif temp >= 99.5 && temp < 103.0

disp("Slightly High");

elseif temp >= 103.0

disp("Dangerously High");

end

**Output:**





**Q-2.** Write a MATLAB program to evaluate the function:

for any user-specified value of *x*, where *x* is a number <1.0 (note that ln is the natural logarithm, the logarithm to the base *e*). Use an if structure to verify that the value passed to the program is legal. If the value of *x* is legal, calculate *y*(*x*). If not, write a suitable error message and quit.

**Code:**

clc;

clear all;

close all;

while(true)

s = 'Enter x value';

x = input(s);

if x < 1.0

y = log(1/(1-x));

fprintf("Y is: %f",y)

break

else

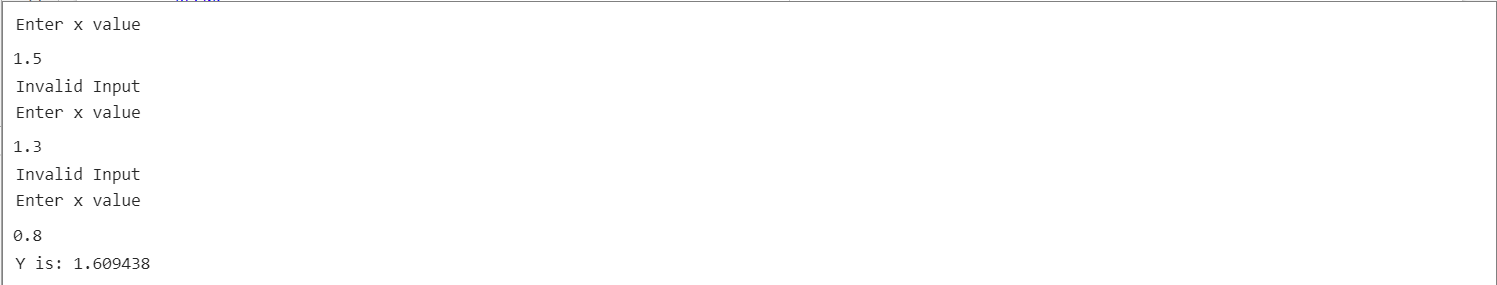
disp("Invalid Input")

end

end

**Output:**





**Q-3.** Examine the following MATLAB statements. Are they correct or incorrect?

If they are correct, what do they output? If they are incorrect, what

is wrong with them?

1. if volts > 125

disp('WARNING: High voltage on line.');

if volts < 105

disp('WARNING: Low voltage on line.');

else

disp('Line voltage is within tolerances.');

end

**Reason:**

Here as we have 2 **if** statements thus, we need 2 **end** statements also.

**Code:**

clc;

clear all;

close all;

s = 'Enter your volts value';

volts = input(s);

if volts < 105

disp('WARNING: Low voltage on line.');

if volts > 125

disp('WARNING: High voltage on line.');

else

disp('Line voltage is within tolerances.');

end

end

**Output:**



1. color = 'yellow';

switch ( color )

case 'red',

disp('Stop now!');

case 'yellow',

disp('Prepare to stop.');

case 'green',

disp('Proceed through intersection.');

otherwise,

disp('Illegal color encountered.');

end

**Code:**

clc;

clear all;

close all;

s = 'Enter your color value';

color = input(s);

switch ( color )

case 'red',

disp('Stop now!');

case 'yellow',

disp('Prepare to stop.');

case 'green',

disp('Proceed through intersection.');

otherwise,

disp('Illegal color encountered.');

end

**Output:**



1. if temperature > 37

disp('Human body temperature exceeded.');

elseif temperature > 100

disp('Boiling point of water exceeded.');

end

**Reason:**

As the input that is temperature won’t enter into the **elseif** condition, as and when the value is greater than 37, because in that case it will execute the **if** condition

**Code:**

clc;

clear all;

close all;

s = 'Enter temp value';

temp = input(s);

if temp > 37 && temp <100

disp('Human body temperature exceeded');

elseif temp >= 100

disp('Boiling point of water exceeded');

end

**Output:**





**Q-4** Write a program that allows a user to enter a string containing a day of the week (“Sunday,” “Monday,” “Tuesday,” etc.) and uses a switch construct to convert the day to its corresponding number, where Sunday is considered the first day of the week and Saturday is considered the last day of the week. Print out the resulting day number. Also, be sure to handle the case of an illegal day name! (*Note:* Be sure to use the 's' option on function input so that the input is treated as a string.)

**Code:**

clc;

clear all;

close all;

s = 'Enter your day';

day = input(s);

switch(day)

case 'Sunday'

disp("1");

case 'Monday'

disp("2");

case 'Tuesday'

disp("3");

case 'Wednesday'

disp("4");

case 'Thursday'

disp("5");

case 'Friday'

disp("6");

case 'Saturday'

disp("7");

otherwise

disp("Invalid Day");

end

**Output:**





**Q-5.** The gain *G* of a certain microwave dish antenna can be expressed as a function of angle by the equation



where θis measured in radians from the boresite of the dish, and sinc *x*=sin *x*/*x*. Plot this gain function on a polar plot, with the title “**Antenna** **Gain vs**” in boldface.

**Code:**

clc;

clear all;

close all;

theta = -pi/2:pi/30:pi/2;

y = abs(sin(4.\*theta)./(4.\*theta));

polarplot(theta,y,'-b')

title("Antenna Gain Vs \theta")

**Output:**

**Q-6.** The author of this book now lives in Australia. Australia is a great place to live, but it is also a land of high taxes. In 2002, individual citizens and residents of Australia paid the following income taxes:



In addition, a flat 1.5% Medicare levy is charged on all income. Write a program to calculate how much income tax a person will owe based on this information. The program should accept a total income figure from the user and calculate the income tax, Medicare levy, and total tax payable by the individual.

**Code:**

clc;

clear all;

close all;

s = 'Enter your Total Income';

inc = input(s);

if inc > 0 && inc <= 6000

tx = 0;

elseif inc>= 6001 && inc <= 20000

tx = (inc - 6000) \* 17;

elseif inc>= 20001 && inc <= 50000

tx = 2380 + (inc - 20000)\*42;

elseif inc> 50001 && inc < 60000

tx = (inc - 50000) \* 42 + 11380;

elseif inc> 60000

tx = (inc - 60000) \* 47 + 15580;

end

ml = 0.015 \* inc ;

fprintf("Income Tax: %.3f \n",tx)

fprintf("Medicare Levy: %.3f \n",ml)

fprintf("Total Tax: %.3f \n",tx + ml)

**Output:**





**Q-7.** Below figure shows a simple high-pass filter consisting of a resistor and a capacitor. The ratio of the output voltage *Vo* to the input voltage *Vi* is given by the equation



Assume that *R* =16 kΩ and *C* =1µF. Calculate and plot the amplitude and phase response of this filter as a function of frequency.

**Code:**

clc;

clear all;

close all;

vD = -1.0:0.1:0.6;

I0 = 2\*10^(-6);

q = 1.602 \* 10^(-19);

k = 1.38 \* 10^(-23);

T = ((75 - 32) \* (5/9)) + 273.15;

iD = I0 \* ( exp( (q.\*vD)./ k.\*T ) - 1);

plot(vD,iD);

hold on;

T = ((100 - 32) \* (5/9)) + 273.15;

iD = I0 \* (exp((q.\*vD)./k.\*T) - 1);

plot(vD,iD);

hold on;

T = ((125 - 32) \* (5/9)) + 273.15;

iD = I0 \* (exp((q.\*vD)./k.\*T) - 1);

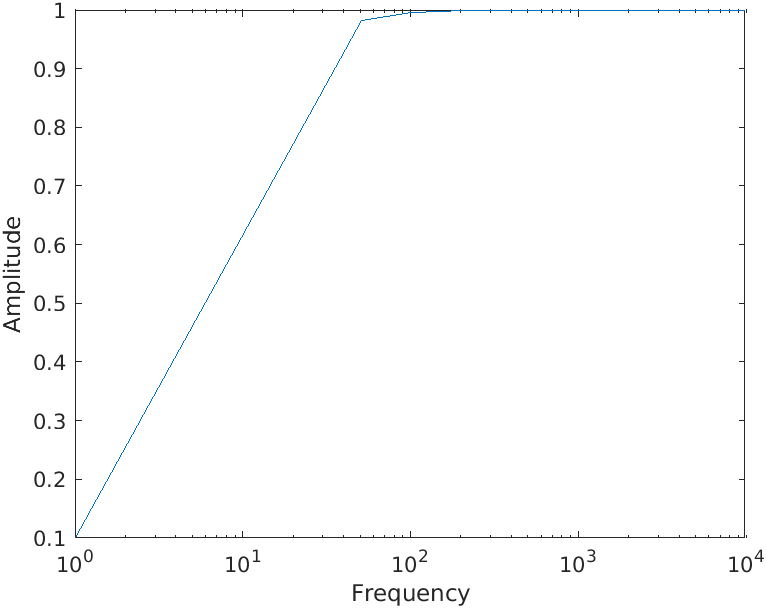
plot(vD,iD);

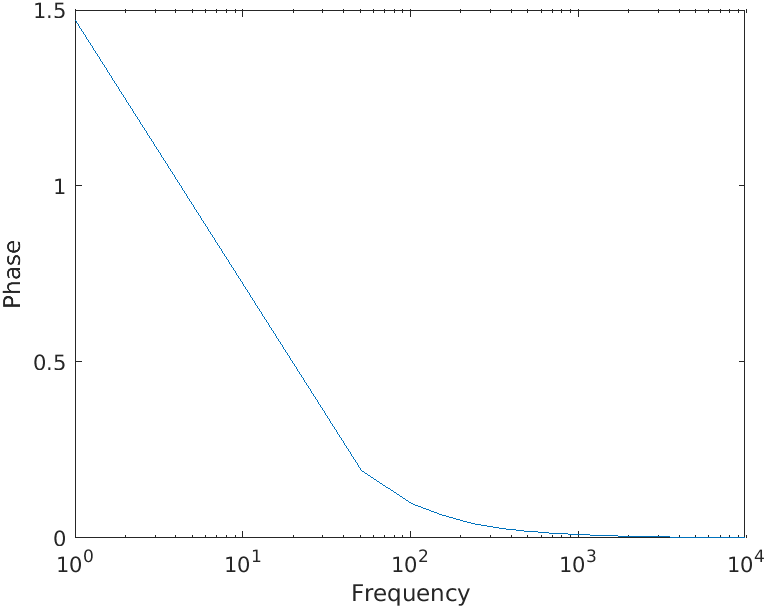
legend("T1 - 75","T2 - 100","T3 - 125");

xlabel("Voltage");

ylabel("Current");

**Output:**

****

****

**Q-8.** When a satellite orbits the Earth, the satellite’s orbit will form an ellipse with the Earth located at one of the focal points of the ellipse. The satellite’s orbit can be expressed in polar coordinates as



where *r* and θ are the distance and angle of the satellite from the center of the Earth, *p* is a parameter specifying the size of the orbit, and ε is a parameter representing the eccentricity of the orbit. A circular orbit has an eccentricity ε of 0. An elliptical orbit has an eccentricity of 0<= ε<=1. If ε >1, the satellite follows a hyperbolic path and escapes from the Earth’s gravitational field.

Consider a satellite with a size parameter *p* =1000 km. Plot the orbit

of this satellite if (*a*) ε =0; (*b*) ε =0.25; (*c*) ε =0.5. How close does each orbit come to the Earth? How far away does each orbit get from the Earth? Compare the three plots you created. Can you determine what the parameter *p* means from looking at the plots?

**Code:**

clc;

close all;

clear all;

p=1000000;

theta = 0:pi/180:2\*pi;

c = 0;

while c < 3

s = 'Enter your E(Eccentricity)';

e = input(s);

r=p./(1-e.\*cos(theta));

plot(theta,r);

hold on;

c = c+1;

end

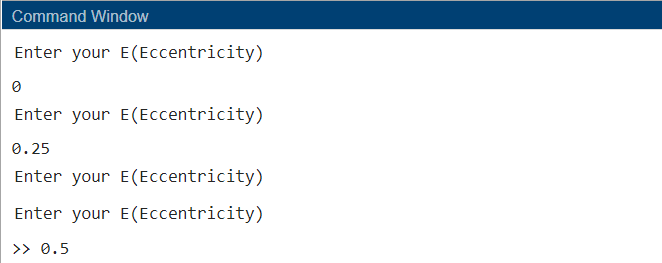
legend("e=0","e=0.25","e=0.5")

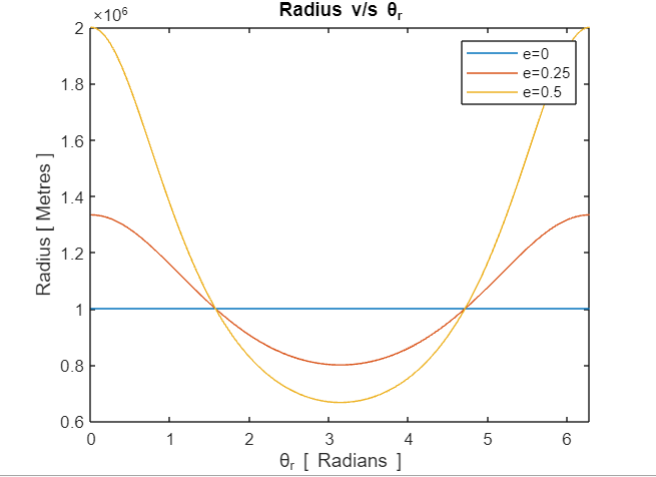
title("Radius v/s \theta\_{r}")

xlabel("\theta\_{r} [ Radians ]")

ylabel("Radius [ Metres ]")

**Output:**





**Reason:**

The one with eccentricity 0 is the closest to the earth and 0.25 a bit far and 0.5 more. As the distance increases the theta similarly modulates.

**Conclusion:**

From this lab we came to understand different conditions for if-else and as well as using while and for loops. The way we used for loops and while depends upon the application. We even got acquainted with the polar graphs and switch case. Where we used multiple statements in the same condition of switch case.