```
    function area = calcrectarea(length, width)
    % This function calculates the area of a rectangle
    % Format of call: calcrectarea(length, width)
    % Returns the area

    area = length * width;
    end
% This script calculates the volume of a hollow sphere

% Assign values for the inner and outer radii
ri = 5.1
ro = 6.8

% Calculate the volume
vol = (4*pi)/3*(ro^3-ri^3)
x=input('Enter in the x value: ');
y=input('Enter in the y value: ');
z=input('Enter in the z value: ');
u=[x y z]/sqrt(x^2 + y^2 + z^2)
% Calculates the wing loading for an airplane

% Prompt the user for the weight and wing area of the plane
plane_weight = input('Enter the weight of the airplane: ');
wing_area = input('Enter the wing area: ');

% Calculate and print the wing loading
fprintf('The wing loading is %.2f\n', plane_weight/wing_area)

% Plots sin(x) with 10 points and 100 points in range 0 to pi

x = linspace(0,pi,10);
y = sin(x);
clf
figure(1)
plot(x,y,'k*')
title('sin(x) with 10 points')
figure(2)
x = linspace(0,pi);
y = sin(x);
plot(x,y,'k*')
title('sin(x) with 100 points')
% Read altitudes and temperatures from a file and plot

load alttemps.dat
altitudes = alttemps(:,1);
temps = alttemps(:,2);
plot(altitudes,temps,'k*')
xlabel('Altitudes')
ylabel('Temperatures')
title('Atmospheric Data')
% Create a vector of integers 1:2:n where n is random
% square them and plot the squares
```

```
n = randi([1,50])
vec = 1:2:n;
vecsq = vec .^ 2;
plot(vecsq,'k*')
title('Squares of integers')
% Display rainfall intensities for a day

load raindata.dat
bar(raindata)
xlabel('Hour')
ylabel('Inches of Rainfall')
title('Hyetograph')
% Calculates the molecular weight of hydrogen peroxide

% Initialize the atomic weights for oxygen and hydrogen
atWtOxygen = 15.9994;
atWtHydrogen = 1.0079;

% Hydrogen peroxide is 2 atoms of hydrogen and 2 of oxygen
molWtHydrogenPeroxide = 2*atWtHydrogen + 2 * atWtOxygen
% Read from a data file the diameter of a part every 10 minutes
% as it is turned on a lathe and plot this data

load partdiam.dat
mins = partdiam(:,1);
diams = partdiam(:,2);
plot(mins,diams,'k*')
xlabel('minutes')
ylabel('part diameter')
% Reads in float numbers stored column-wise from a file,
% rounds them to integers and writes row-wise to a new file

load floatnums.dat
inums = round(floatnums)';
save intnums.dat inums -ascii
% Solve the Y2K problem!

load hightemp.dat

newmat = hightemp;
newmat(:,1) = newmat(:,1) + 1900;

save y2ktemp.dat newmat -ascii% Prompt the user for a number of inches
of rain
% and call a function to calculate the
% equivalent amount of snow

rain = input('How much rain in inches: ');
snow = rainToSnow(rain);
fprintf('%.1f inches of rain would be ', rain)
fprintf('%.1f inches of snow\n', snow)
```

```
% Calculates and prints the volume of a regular tetrahedron
% Prompts user for length of sides, calls a function to
% calculate the volume, and prints the result

len = input('Enter the length of the sides: ');
volume = tetVol(len);
fprintf('The volume of a regular tetrahedron with ')
fprintf(' sides of length %.1f is %.2f\n', len, volume)A = 1;

tau1 = input('Enter a time constant: ');
tau2 = input('Enter another time constant: ');

tstart = input('Enter the beginning t: ');
tend = input('Enter the end of t: ');

t = linspace(tstart,tend);

y1 = expfn(A, t, tau1);
y2 = expfn(A, t, tau2);

plot(t,y1,'r*',t,y2,'go')
xlabel('x')
ylabel('y')
title('Exp function')
legend('tau1','tau2')
F = input('Enter the impulse force: ');
x = -2*pi:0.1:2*pi;
% ethyl alcohol
a = 0.246;
b = 0.806;
y = expdecay(F,x,a,b);
plot(x,y,'r*')
hold on
% water
a = 0.250;
b = 1;
y = expdecay(F,x,a,b);
plot(x,y,'g+')
% oil
a = 0.643;
b = 1.213;
y = expdecay(F,x,a,b);
plot(x,y,'bo')
legend('Ethyl Alc','Water','Oil')
xlabel('x')
ylabel('y')
title('Decaying sinusoids')
function velmps = convertVel(velmph)
% Convert the velocity of an aircraft from
% miles per hour to meters per second
% Format of call: convertVel(mph)
% Returns velocity in mps
```

```
velmps = velmph/3600*5280*.3048;
end
function out = costn(n)
% Calculates the cost of manufacturing n
%  units of a product
% Format of call: costn(n)
% Returns cost as 5*n^2-44*n+11

out = 5*n^2 - 44*n + 11;
end% Prompt the user and print a string in quotes

str = input('Enter your string: ', 's');
fprintf('Your string was: ''%s''\n',str)
function y = expdecay(F,x,a,b)
y = F * exp(-a*x).* sin(b*x);
end
function y = expfn(A,t,tau)
y = A * exp(-tau*t);
end
function five = fives(r,c)
% Returns a matrix of fives of specified size
% Format of call: fives(rows, cols)
% Returns a rows by cols matrix of all fives

% Initialization
five = zeros(r,c) + 5;

end
% Converts a flow rate from cubic meters per second
%  to cubic feet per second

cubMperSec = input('Enter the flow in m^3/s :');
cubFperSec = cubMperSec / .028;
fprintf('A flow rate of %.3f meters per sec\n', cubMperSec)
fprintf('is equivalent to %.3f feet per sec\n', cubFperSec)function
waterVel = fluidvel(totp, statp)
% Calculates the velocity of water given the
% total and static pressures
% Format: fluidvel(total pressure, static pressure)
% Returns the velocity of the water

waterVel = 1.016 * sqrt(totp-statp);
end
function out = isdivby4(inarg)
% Returns 1 for true if the input argument is
%  divisible by 4 or 0 for false if not
% Format of call: isdivby4(input arg)
% Returns whether divisible by 4 or not

out = rem(inarg,4) == 0;
end
```

```
function out = isint(innum)
% Returns 1 for true if the argument is an integer
% Format of call: isint(number)
% Returns logical 1 iff number is an integer

out = innum == int32(innum);
endfunction out = ispythag(a,b,c)
% Determines whether a, b, c are a Pythagorean
%  triple or not
% Format of call: ispythag(a,b,c)
% Returns logical 1 if a Pythagorean triple

out = a^2 + b^2 == c^2;
end
% Calculates the luminosity of a star

disp('This script will calculate the luminosity of a star.')
disp('When prompted, enter the star''s distance from the sun')
fprintf('  in meters, and its brightness in W/meters squared.\n\n')

d = input('Enter the distance: ');
b = input('Enter the brightness: ');
L = 4*pi*d^2*b;

fprintf('The luminosity of this star is %.2f watts\n', L)
% Prompts the user for the number of units, calls a function
% to calculate the cost for producing that many units, and
% prints this result

n = input('Enter the number of units: ');
costPerN = costn(n);
fprintf('The cost for %d units will be ',n)
fprintf('$%.2f\n', costPerN)function out = mwh_to_gj(mwh)
% Converts from MWh to GJ

% Format of call: mwh_to_gj(mwh)
% Returns gigajoules

out = mwh * 3.6;
endfunction p = perim(radius)
% Calculates the perimeter of a circle

p = 2 * pi * radius;
end
function elem = pickone(invec)
% pickone(x) returns a random element from vector x
% Format of call: pickone(vector)
% Returns random element from the vector

len = length(invec);
ran = randi([1, len]);
elem = invec(ran);
```

```
end
function outsnow = rainToSnow(rain)
% Calculate equivalent amount of snow
% given rainfall in inches
% Format of call: rainToSnow(rain)
% Returns equivalent snowfall

outsnow = rain * 6.5;
end
function outvec = repvec(vec,n)
% Duplicates every number in a vector n times
% Format of call: repvec(vector, n)
% Returns a new vector

mat = repmat(vec,n,1);
outvec = reshape(mat,1,n*length(vec));
end
load costssales.dat

costs = costssales(:,1);
sales = costssales(:,2);

len = length(costs); % or sales

x = 1:len;  % Note: not necessary
plot(x,costs,'ko')
hold on
plot(x,sales,'k*')
legend('Costs', 'Sales')
fprintf('There were %d quarters in the file\n', len)
neworder = rot90(costssales)
% neworder = flipud(costssales');
save newfile.dat neworder -ascii
function a = side3(b,c,alpha)
a = sqrt(b^2 + c^2 -2*b*c*cos(alpha));
end
function vol = tetVol(len)
% Calculates the volume of a regular tetrahedron
% Format of call: tetVol(side length)
% Returns volume

vol = 1/12 * sqrt(2) * len^3;
end% Calculates the third side of a triangle, given
% the lengths of two sides and the angle between them

b = input('Enter the first side: ');
c = input('Enter the second side: ');
alpha = input('Enter the angle between them: ');
alpha = alpha * pi / 180;

a = sqrt(b^2 + c^2 -2*b*c*cos(alpha));
fprintf('\nThe third side is %.3f\n', a)
```

```
% Calculates the third side of a triangle, given
% the lengths of two sides and the angle between them

b = input('Enter the first side: ');
c = input('Enter the second side: ');
alpha = input('Enter the angle between them: ');
alpha = alpha * pi / 180;

a = side3(b,c,alpha);
fprintf('\nThe third side is %.3f\n', a)
function outvec = vecout(innum)
% Create a vector from innum to innum + 5
% Format of call: vecout(input number)
% Returns a vector input num : input num+5

outvec = innum:innum+5;
end
```