

**QUIZ I (1/27/12)**

Name \_\_\_\_\_

Partial credit will be given if you show all your work.

Consider the following MATLAB program entered at the Command Window,

```
>> t=0:0.5:20;  
>> f1=sin(2*pi*0.1*t);  
>> f2=exp(-0.25*t);  
>> f=f1.*f2;  
>> plot(t,f)
```

What would be the maximum value of the function f?

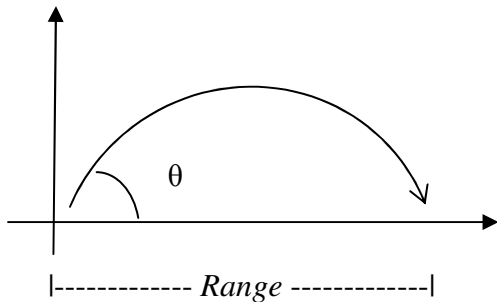
**QUIZ II (2/3/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work.

Follow steps 1-5 to solve this problem: 1) State the problem, 2) Describe inputs/outputs, 3) Develop hand example, 4) Develop MATLAB solution, 5) Verify solution. Label your plots.

The range of an object shot at an angle  $\theta$  with respect to the x-axis and an initial velocity  $v_0$  is given by the trajectory shown below. The *Range* can be obtained by the following equation,



$$Range = \frac{v_0^2}{g} \sin(2\theta)$$

The above equation neglects air resistance and uses standard gravity of  $g = 9.81 \text{ m/s}^2$ . Plot the *Range* (in a 3-D plot) for velocity range  $0 \leq v_0 \leq 100 \text{ m/s}$  and shooting angle range  $0 \leq \theta \leq 90^\circ$ . Find the angle that gives the maximum range.

```
>> [v,t]=meshgrid(0:100,0:0.01:pi/2); g=9.81; r=(v.^2/g).*sin(2*t);
>> mesh(v,t*180/pi,r); title('Ranges for vel. vs. angle'); xlabel('vel'); ylabel('angle'); zlabel('Range');
```

The function is quadratic with same maximum angle for all velocity, so pick for example  $v_0=50 \text{ m/s}$ :

```
>> v=100; t=0:0.01:pi/2; g=9.81; r=(v^2/g)*sin(2*t); [mx,idx]=max(r); t(idx)*180/pi
```

ans =

45.2637

**QUIZ III (2/10/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

**(20%)** Follow the problem solving in engineering and science approach: 1) State the problem, 2) Describe inputs/outputs, 3) Develop hand example, 4) Develop MATLAB solution, 5) Verify solution.

**(80%)** Create a MATLAB function called *free\_fall*, with time  $t$  and acceleration due to gravity  $g$  as inputs, that computes and returns values for distance  $x$  and velocity  $v$ . The inputs should be read from the keyboard while the program is running in the main program.

The distance a freely falling object travels is

$$x = \frac{1}{2}gt^2$$

Where  $g = 9.8 \text{ m/s}^2$

We can find the velocity of the object by taking the derivative of the preceding equation,

$$\frac{dx}{dt} = v = gt$$

Test your function with a time vector that ranges from 0 to 20 seconds, and compare it to your hand example.

Display the results in the main program using the following format:

*Distance and velocity in free fall*

*Time Distance Velocity*

*0.0 0.0 0.0*

*1.0 4.9 9.8*

*.*

*.*

*.*

```
% This program reads t and g from the user, and computes x, v, a
% Written by: HE
% Purpose: Quiz
% Date: 2/10/12
```

```
t = input('input time vector: ');
g = input('input gravity: ');
[d, v] = free_fall(t,g);
disp('Distance and velocity in free fall');
disp(' Time Distance Velocity');
table=[t;d;v];
fprintf('%8.1f %8.1f %8.1f\n',table);
```

```
function [x,v] = free_fall(t,g)
% This function requires time and acceleration
% and returns distance and velocity
x = 1/2.*g.*t.^2;
v = g.*t;
```

**QUIZ IV (2/17/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

**(20%)** Follow the problem solving in engineering and science approach: 1) State the problem, 2) Describe inputs/outputs, 3) Develop hand example, 4) Develop MATLAB solution, 5) Verify solution.

**(80%)** At  $t=0$ , when a rocket's engine shuts down, the rocket has reached an altitude of 500 m and is rising a velocity of 125 m/s. At this point, gravity takes over. The height of the rocket as a function of time is

$$h(t) = -\frac{9.8}{2}t^2 + 125t + 500 \quad \text{for } t > 0$$

- Use a flowchart to design your MATLAB program.
- Plot the height of the rocket from 0:30 seconds.
- Use the *ginput* function to estimate the maximum height the rocket reaches, and the time when the rocket hits the ground.
- Use the *disp* command to report your results to the command window.

```
% Purpose: Quiz 4 (Plot rocket trajectory)
% Written by: HE
% Input(s): t
% Output(s): Max height, and time to hit ground
% Date: 2/12

t = 0:30;
h = -9.8/2*t.^2 + 125*t + 500;
plot(t,h)
disp('Position the cursor over the maximum height on the graph')
[a,b] = ginput(1)
disp(['The maximum height reached by the rocket is approximately ' num2str(b) '
meters'])
disp(['Position the cursor over the position where the rocket hits the ground'])
[a,b] = ginput(1)
disp(['The time the rocket hits the ground is ' num2str(a) ' seconds'])
```

**QUIZ V (2/24/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

**(20%)** Follow the problem solving in engineering and science approach: 1) State the problem, 2) Describe inputs/outputs, 3) Develop hand example, 4) Develop MATLAB solution, 5) Verify solution.

**(80%)** Create a program that runs an for infinite number of iterations, and which uses a function called fact2 to find the factorial of a number, N!. Include an if statement in the main program to check for negative numbers, and use a while or a for loop to compute the factorial number in the fact2 function. Exit the program if the user inputs a negative number (which will be a way of terminating the program). Develop a flowchart to help you plan your code.

Hint: The factorial of a non-negative integer N, denoted by N!, is the product of all positive integers less than or equal to N. For example,

$$5! = 1 \times 2 \times 3 \times 4 \times 5 = 120$$

```
% Purpose: find factorial of a number
% Written by: HE
% Date: 2/12

while(1)
    N=input('input N ');
    if(N<0)
        disp('invalid input ');
        break;
    end
    nf=fact2(N)
end

function nf=fact2(N)
% fact2: user defined factorial function
%
nf=1;
for k=1:N
    nf=nf*k;
end
```

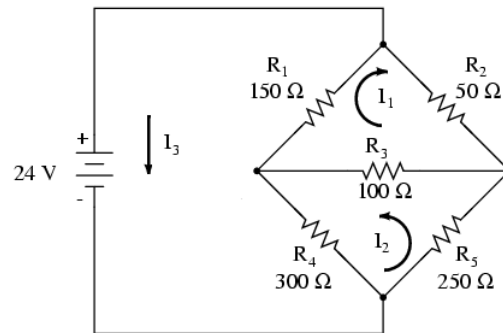
**QUIZ VI (3/2/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

(20%) Follow the problem solving in engineering and science approach: 1) State the problem, 2) Describe inputs/outputs, 3) Develop hand example, 4) Develop MATLAB solution, 5) Verify solution.

(80%) Consider the following electrical circuit.



Use Kirchhoff Voltage Law (KVL): sum of voltages around a circuit must be zero, to solve for the currents  $I_1$ ,  $I_2$ , and  $I_3$ .

Create a MATLAB program to solve for the system of linear equations using **a)** the matrix inverse method, and **b)** the left-division approach. Which method takes more time to execute?

$$50I_1 + 100(I_1 + I_2) + 150(I_1 + I_3) = 0 \quad \text{so} \quad 300I_1 + 100I_2 + 150I_3 = 0$$

$$250I_2 + 100(I_2 + I_1) + 300(I_2 - I_3) = 0 \quad \text{so} \quad 100I_1 + 650I_2 - 300I_3 = 0$$

$$-24 + 150(I_3 + I_1) + 300(I_3 - I_2) = 0 \quad \text{so} \quad 150I_1 - 300I_2 + 450I_3 = 24$$

```
% Purpose: To find mesh currents
% Inputs: Voltages and Resistances
% Outputs: Mesh currents
% Written by: HE
% Date: 3/12
```

```
clear; clc;
A=[300 100 150;100 650 -300;150 -300 450];
B=[0; 0; 24];
```

```
% Compute currents using inverse method
tic
I=inv(A)*B;
```

toc

```
% Compute currents using left-division  
tic  
I=A\B;  
toc
```

Elapsed time is 0.000053 seconds.

Elapsed time is 0.000036 seconds.

Hand example

Substituting back:

```
>> 150*ans(1)-300*ans(2)+450*ans(3)
```

ans =

24.0000

This verifies that currents are correct.



**QUIZ VII (3/9/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

**(20%)** Follow the problem solving in engineering and science approach: 1) State the problem, 2) Describe inputs/outputs, 3) Develop hand example, 4) Develop MATLAB solution, 5) Verify solution.

**(60%)** Keeping information private in an electronic edge is becoming more and more difficult. One approach is to encode information, so that even if an unauthorized person sees the information, he or she won't be able to understand it. Modern coding techniques are extremely complicated, but we can create a simple code by taking advantage of the way character information is stored in MATLAB. If we add or subtract a constant integer value to character information, we can transform the string into something that is difficult to interpret.

Download the encoded message from the EE289 website and decode it. Because you may need to add and subtract many numbers, which produce different combinations, you may want to set up a loop to go through all the combinations.

**(20 %)** Develop a flowchart to help you plan your code.

```
% Purpose: Decode a message
% Input: encoded message
% Output: decoded message
% Written by: HE
% Date: 3/2012

clear; clc;
a='You have earned a 100 on this quiz!';
v=single(a);
coded=v+25;
Encoded_message=char(coded);
save('Message.mat','Encoded_message');

load Message.mat
message=single(Encoded_message);
for i=-100:100
    decoded=message+i;
    char(decoded)
    i
    pause(1)
end
```

**QUIZ VIII (3/23/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

(20 %) Develop a flowchart to help you plan your code for the following problem.

(80%) We want to use the symbolic math capabilities of MATLAB to explore the equations representing the path followed by an unpowered projectile, such as a flare that is used to identify a location when GPS may not be available. We know from elementary physics that the distance a projectile travels horizontally is

$$dx = v_0 t \cos(\theta)$$

and the distance traveled vertically is

$$dy = v_0 t \sin(\theta) - 1/2 g t^2$$

Use these equations to derive an equation for the distance the flare has traveled horizontally when it hits the ground (the range). What is the maximum range traveled for angles from 0 to  $\pi/2$ . Assume  $v_0=100$  m/s and  $g=9.8$  m/s<sup>2</sup>

```
>> clear, clc
>> syms v0 t theta g
>> dx=v0*t*cos(theta); dy=v0*t*sin(theta)-0.5*g*t^2;
>> impact_time=solve(dy,t)

impact_time =

      0
(2*v0*sin(theta))/g

>> impact_distance=subs(dx,t,impact_time(2))

impact_distance =

(2*v0^2*cos(theta)*sin(theta))/g

>>rge=subs(impact_distance,{v0 theta g},{100 0:(pi/2)/100:pi/2 9.8});
>> max(rge)

ans =

1.0204e+03
```

**QUIZ IX (3/30/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

(20 %) Develop a flowchart to help you plan your code for the following problem.

(80%) Consider the following equation:

$$y = 12x^3 - 5x^2 + 3$$

- Define an x vector from -5 to +5, and use it together with the *diff* function to approximate the derivative of y with respect to x numerically.
- Obtain the derivative of y with respect to x analytically.
- Plot the two functions and see how they differ.

```
% Purpose: Quiz 8
% Input(s): x vector
% Output(s): plots of dx/dy obtained numerically and analytically
% Written by: He

x=-5:5;
y=12*x.^3-5*x.^2+3;
num_diff=diff(y)./diff(x)
dydx=36*x.^2-10*x
table=[ [num_diff,NaN] ', dydx']
plot(x,table)
title('A Comparison of Numerical Derivatives and Analytical Derivatives')
legend('Numerical Derivative','Analytical Derivative')
xlabel('x')
ylabel('dy')
```

**QUIZ IX (4/4/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your MATLAB code.

(20 %) Develop a flowchart to help you plan your code for the following problem.

(80%) Consider a gas in a piston-cylinder device in which the temperature is held constant. As the volume of the device was changed, the pressure was measured. The volume and pressure values are reported in the following table:

Volume, m <sup>3</sup>	Pressure, kPa
1	2494
2	1247
3	831
4	623
5	499
6	416

- Use linear interpolation to estimate the pressure when the volume is 3.8 m<sup>3</sup>.
- Use cubic spline interpolation to estimate the pressure when the volume is 3.8 m<sup>3</sup>.
- Which technique produces the best estimate?

```
% Purpose: Quiz 8
% Input(s): x vector
% Output(s): use of interpolation techniques
% Written by: He
%% Problem 13.1
```

```
V=1:6;
P=[2494, 1247, 831, 623, 499, 416];
interpl(V,P,3.8)
interpl(V,P,3.8,'spline')
```

The linear interpolation provides the best estimate.

**QUIZ X (4/11/12)**

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your C code.

**(20 %)** Develop a flowchart to help you plan your code for the following problem.

**(80%)** Write a C program to compute the area of a circle with radius  $r$ . (Recall that  $A=\pi r^2$ .)

```
// Purpose: Quiz X
// Inputs: radius
// Output(s): Area
// Written by: HE
// Date: 4/2012

#include <stdio.h>
#define PI 3.141593

int main(void)
{
    // Declare variables.
    double r, area;

    // Enter the radius.
    printf("Enter the radius of the circle: \n");
    scanf("%lf",&r);

    // Compute the area of the circle.
    area = PI*r*r;

    // Print the value of the area.
    printf("The area of a circle with radius %5.3f is %5.3f. \n",r,area);

    // Exit program.
    return 0;
}
```

QUIZ XI (4/20/12)

Name: \_\_\_\_\_

Partial credit will be given if you show all your work. Comment your C code.

(20 %) Develop a flowchart to help you plan your code for the following problem.

(80%) Write a C program to compute the sinusoidal function:

$$s(t) = 5 \cdot \sin(0.4 \cdot \pi \cdot t)$$

Compute the function for 10 seconds, in increments of 0.1 seconds. Save the  $t$  and  $s(t)$  values to a data file, and use MATLAB to plot the function.

```
// Purpose: compute a sinudoidal function
// Inputs: none
// Outputs: data file with t and s(t)
// Written by: HE
// Date: 4/12

#include <stdio.h>
#include <math.h>
#define PI 3.1416
FILE *F1;

int main(void)
{
    // declare vars
    double deltat, k, sfunc;

    // open data file
    F1=fopen("data.txt", "w");

    // loop through values
    for(k=0;k<=100;k++)
    {
        // Compute deltat, s(1), and write to a file
        deltat=k*0.1; sfunc = 5*sin(0.4*PI*deltat);
        fprintf(F1, "%8.3f%8.3f\n",deltat,sfunc);
    }
    fclose(F1);

    return(0);
}
```

```
>> d=textread('data.txt'); plot(d(:,1),d(:,2))
>> title('sine function'); ylabel('s(t)'); xlabel('t');
```