Recitation 5 Topics

• Solved Problems (Exam I Review)
  – Solution of ODEs
  – Stability

• MATLAB Program Flow Control
  – Relational Operators
  – Logical Operators
  – Conditional Statements
  – For Loop
  – While Loop
Problem 1

- A LTI system with input $f(t)$ and output $y(t)$ has the ODE representation

$$\ddot{y} + \dot{y} = f(t)$$

1. Determine the roots of the characteristic equation and sketch the roots in the $\lambda$–plane

2. State if the system is unstable, marginally stable, or unstable

3. Determine the zero-state unit-step response

4. Determine the zero-state response for $f(t) = \cos(t) \ u(t)$
Problem 1 Solution
Problem 2

A second-order LTI system has the following zero-state and zero-input responses for a given input $f(t)$ and set of initial conditions:

\[
y_{zi}(t) = -e^{-2t} + 2e^{-t} \quad t \geq 0
\]

\[
y_{zs}(t) = -4e^{-2t} + 2e^{-t} + 2e^{-3t} \quad t \geq 0
\]

• Determine
  1. The characteristic equation
  2. The total response
  3. The natural response (homogeneous solution)
  4. The forced response (particular solution)
  5. The initial conditions
Relational Operators

• Compare variable values and produce a result that is true (1) or false (0)

• Example

```matlab
>> x = 10; y = 20; % assign values to two variables
>> z = x < y;          % compare x < y
z = 1
```

• The syntax of a relational expression is

```
left expression relational operator right express
```
## List of Relational Operators

<table>
<thead>
<tr>
<th>Relational Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>==</td>
<td>Equal to</td>
</tr>
<tr>
<td>~=</td>
<td>Not equal to</td>
</tr>
</tbody>
</table>
Problem 3

• Determine the results of the following expressions
  – verify using MATLAB

```matlab
>> a = -1; b = 2; c = 3; d = 4;
>> e = a^2 + b^2 >= c^2 + d^2
e =
>> f = (b > a) + 2*(c < d)
f =
>> x = [0, 2, 4, 6, 8]; y = [1, -1, 8, 2, 8];
>> g = x > 3
g =
>> h = x >= y
h =
```
Logical Operators

- Logical operators work on logical variables and yield a logical value (0 or 1)
- Use parentheses to control the order of evaluation
- Logical operators work on numbers
  - The number 0 is used as logical 0
  - A nonzero number is used as logical 1

<table>
<thead>
<tr>
<th>Logical Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;</td>
<td>AND</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>xor(a,b)</td>
<td>Exclusive OR (XOR)</td>
</tr>
<tr>
<td>~</td>
<td>NOT</td>
</tr>
</tbody>
</table>
Short-Circuit Logical Operators

• Evaluate the second operand only when the result is not fully determined by the first operand

<table>
<thead>
<tr>
<th>Logical Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;&amp;</td>
<td>Short-circuit Logical AND</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• Examples
  – If $A = 0$, then MATLAB will not evaluate $B$ when determining $A \&\& B$
  – If $A = 1$, then MATLAB will not evaluate $B$ when determining $A \| B$
Problem 4

• Determine the results of the following expressions
  – verify using MATLAB

```matlab
>> a = 0; b = 1; c = pi;
>> b & c
ans =
>> a | b
ans =
>> xor(b,c)
ans =
>> ~c
ans =
```
The if-end Structure

• Simplest if structure

\[
\text{if expression} \\
\text{block of statements} \\
\text{end}
\]

• If the expression is true the block of statements is executed, otherwise the block of statements is skipped and execution continues after the end statement
The if-else-end Structure

• Provides a means for choosing among two blocks of statements to execute

```
if expression
  block #1 of statements
else
  block #2 of statements
end
```

• If the expression is true, statements in block #1 execute

• If the expression is false, statements in block #2 execute
The if-elseif-else-end Structure

- Most general if structure

```
if expression 1
  block #1 of statements
elseif expression 2
  block #2 of statements
else
  block #N of statements
end
```
Problem 5

- Determine x, verify using MATLAB

```matlab
>> x = -10;
>> if (x < -2) || (x > 2), x = 2*sign(x); end
x = 4;
>> if x > 0, x=sqrt(x); else x = x^2; end
>> x
x =
```

- Determine x, verify using MATLAB

```matlab
>> x = -10;
>> if (x < -2) || (x > 2), x = 2*sign(x); end
x =
>> x = 4;
>> if x > 0, x=sqrt(x); else x = x^2; end
>> x
x =
```
For Loop

- Repeats executing a block of statements a predetermined number of times

- Syntax

  for loop_index = first_index : index_increment : last_index
  block of statements
  end
Problem 6

• Consider the row vector \( x \) and column vector \( y \)

\[
>> x = \text{ones}(1,10000); \quad y = \text{ones}(10000,1)
\]

• Use a For Loop to determine the product \( xy \)
  – Measure the execution time using the tic and toc functions – use MATLAB help to determine the syntax

• Determine the product using a command line multiply
  – Measure the execution time

• Which approach is faster?
While Loop

• Repeats executing a block of statements until an expression is no longer true

• Syntax

```plaintext
while expression
  block of statements
end
```
Problem 7

• Use a While Loop to display $2^0, 2^1, \ldots, 2^N$, where the integer $N \geq 0$

• Verify using MATLAB
Recitation 5 Topics

• Solved Problems (Exam I Review)
  – Solution of ODEs
  – Stability

• MATLAB Program Flow Control
  – Relational Operators
  – Logical Operators
  – Conditional Statements
  – For Loop
  – While Loop
Problem 1

• A LTI system with input $f(t)$ and output $y(t)$ has the ODE representation

\[ \ddot{y} + \dot{y} = f(t) \]

1. Determine the roots of the characteristic equation and sketch the roots in the $\lambda$–plane

2. State if the system is unstable, marginally stable, or unstable

3. Determine the zero-state unit-step response

4. Determine the zero-state response for $f(t) = \cos(t) \ u(t)$
Problem 1 Solution
Problem 1 Solution
Problem 2

• A second-order LTI system has the following zero-state and zero-input responses for a given input $f(t)$ and set of initial conditions

$$y_{zi}(t) = -e^{-2t} + 2e^{-t} \quad t \geq 0$$

$$y_{zs}(t) = -4e^{-2t} + 2e^{-t} + 2e^{-3t} \quad t \geq 0$$

• Determine
  1. The characteristic equation
  2. The total response
  3. The natural response (homogeneous solution)
  4. The forced response (particular solution)
  5. The initial conditions
Relational Operators

• Compare variable values and produce a result that is true (1) or false (0)

• Example

```plaintext
>> x = 10; y = 20; % assign values to two variables
>> z = x < y;       % compare x < y
z = 1
```

• The syntax of a relational expression is

```
left expression    relational operator    right express
```
## List of Relational Operators

<table>
<thead>
<tr>
<th>Relational Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>==</td>
<td>Equal to</td>
</tr>
<tr>
<td>~=</td>
<td>Not equal to</td>
</tr>
</tbody>
</table>
Problem 3

• Determine the results of the following expressions
  – verify using MATLAB

```matlab
>> a = -1; b = 2; c = 3; d = 4;
>> e = a^2 + b^2 >= c^2 + d^2
   e = 
>> f = (b > a) + 2*(c < d)
   f = 
>> x = [0, 2, 4, 6, 8]; y = [1, -1, 8, 2, 8];
>> g = x > 3
   g = 
>> h = x >= y
   h = 
```
Logical Operators

• Logical operators work on logical variables and yield a logical value (0 or 1)

• Use parentheses to control the order of evaluation

• Logical operators work on numbers
  – The number 0 is used as logical 0
  – A nonzero number is used as logical 1

<table>
<thead>
<tr>
<th>Logical Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;</td>
<td>AND</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>xor(a,b)</td>
<td>Exclusive OR (XOR)</td>
</tr>
<tr>
<td>~</td>
<td>NOT</td>
</tr>
</tbody>
</table>
Short-Circuit Logical Operators

• Evaluate the second operand only when the result is not fully determined by the first operand

<table>
<thead>
<tr>
<th>Logical Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;&amp;</td>
<td>Short-circuit Logical AND</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• Examples
  – If A = 0, then MATLAB will not evaluate B when determining A && B
  – If A = 1, then MATLAB will not evaluate B when determining A || B
Problem 4

• Determine the results of the following expressions
  – verify using MATLAB

```matlab
>> a = 0; b = 1; c = pi;
>> b & c
ans =

>> a | b
ans =

>> xor(b,c)
ans =

>> ~c
ans =
```
The if-end Structure

• Simplest if structure

```
if expression
block of statements
end
```

• If the expression is true the block of statements is executed, otherwise the block of statements is skipped and execution continues after the end statement
The if-else-end Structure

- Provides a means for choosing among two blocks of statements to execute

```plaintext
if expression
    block #1 of statements
else
    block #2 of statements
end
```

- If the expression is true, statements in block #1 execute
- If the expression is false, statements in block #2 execute
The if-elseif-else-end Structure

• Most general if structure

```plaintext
if expression 1
    block #1 of statements
elseif expression 2
    block #2 of statements
    :
else
    block #N of statements
end
```
Problem 5

• Determine $x$, verify using MATLAB

```matlab
>> x = -10;

>> if (x < -2) || (x > 2), x = 2*sign(x); end
```

```matlab
>> x
x = 

>> x = 4;

>> if x > 0, x=sqrt(x); else x = x^2; end
```

```matlab
>> x
x = 
```
For Loop

• Repeats executing a block of statements a predetermined number of times

• Syntax

   for loop_index = first_index : index_increment : last_index
       block of statements
   end
Problem 6

• Consider the row vector \( x \) and column vector \( y \)
  \[
  >> x = \text{ones}(1,10000); \ y = \text{ones}(10000,1)
  \]

• Use a For Loop to determine the product \( xy \)
  – Measure the execution time using the tic and toc functions – use MATLAB help to determine the syntax

• Determine the product using a command line multiply
  – Measure the execution time

• Which approach is faster?
While Loop

• Repeats executing a block of statements until an expression is no longer true

• Syntax

```
while expression
    block of statements
end
```
Problem 7

• Use a While Loop to display $2^0, 2^1, \ldots, 2^N$, where the integer $N \geq 0$

• Verify using MATLAB