1. Which statement assigns the value 3 to the variable `myNum` and prints the value of `myNum` to the command window?
   (a) `myNum == 2 + 1;`
   (b) `myNum = 2 + 1;`
   (c) `myNum == 3`
   (d) `myNum = 2 + 1 <=`
   (e) `myNum = 3;`

2. What is the value of oranges after the following code executes?
   ```
   apples = 8;
   oranges = 2;
   oranges = oranges + apples;
   apples = apples - 1;
   ```
   (a) 10
   (b) 9
   (c) 8
   (d) 7
   (e) 2

3. Which of the following is not a valid Matlab identifier?
   (a) `MyNumber`
   (b) `my_array2`
   (c) `myArray`
   (d) `start`
   (e) `end`

4. Indicate whether each statement is true or false.
   (T/F) A double precision number’s range is about 100 times greater than a single precision number’s range.
   (T/F) Following the assignment `myNum = 4`, the variable `myNum` will be of type `int32`.
   (T/F) Floating point numbers can represent larger numbers than unsigned integers.
   (T/F) Due to rounding error, it is not necessarily true that `(a + b) + c = a + (b + c)` for some floating points values a, b, c.
   (T/F) `eps(number)` will give the relative error associated with representing that number in floating point.
   (T/F) Computing `0/0` will yield the value `Inf`.
   (T/F) Computing `Inf - Inf` will yield the value `NaN`.

5. Using the precedence rules for arithmetic operators, determine which statement is true.
   (a) `2 / 2 * 3 == 2 / 6`
   (b) `2 ^ 2 * 2 == 16`
   (c) `1 + 2 * 3 == 9`
   (d) `4 / 2 + 2 == 4 <=`
6. Consider the function definition,

```matlab
function [ apples, oranges ] = MyFunction( pears )
    apples = pears * 2;
    oranges = pears - 1;
end
```

and the following code that calls this function:

```matlab
apples = 1;
oranges = 2;
pears = 3;
[ oranges, pears ] = MyFunction( apples );
```

What will be the value of **apples**, **oranges**, and **pears** after the code executes?

(a) apples = 1, oranges = 0, and pears = 2
(b) apples = 6, oranges = 2, and pears = 3
(c) apples = 1, oranges = 2, and pears = 0
(d) apples = 6, oranges = 2, and pears = 3
(e) apples = 1, oranges = 2, and pears = 3

7. Which statement regarding functions and scripts is false?

(a) By default, variables defined in a script will be visible in the main workspace.
(b) By default, variables defined in a function will not be visible in the main workspace.
(c) Functions can be called with multiple input parameters.
(d) Scripts are terminated with the `end` keyword. ⇐
(e) Scripts are not called with input parameters.

8. For each `fprintf` statement in the left column, draw a line to the corresponding output in the right column.

<table>
<thead>
<tr>
<th><code>fprintf('--%4.2f--', 3.1415926)</code></th>
<th><code>--3.14--</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>fprintf('--%5.2f--', 3.1415926)</code></td>
<td><code>-- 3.14--</code></td>
</tr>
<tr>
<td><code>fprintf('--%4.2e--', 3.1415926)</code></td>
<td><code>--3.14e+00--</code></td>
</tr>
<tr>
<td><code>fprintf('--%f--', 3.1415926)</code></td>
<td><code>--3.141593--</code></td>
</tr>
</tbody>
</table>

9. Which statement regarding 1D arrays is true?

(a) `myArray = [ 1, 2, 3, 4 ];` creates a row array of size 4 × 1.
(b) `myArray = [ 1, 2, 3, 4 ];` creates a row array of size 4 × 1.
(c) `myArray = [ 3:-1:0 ];` creates a row array of size 1 × 3.
(d) `myArray = [ 1:2:4 ];` creates a row array of size 1 × 3.
(e) `myArray = [ 1:7 ];` `myArray(6) = [];` results in an array of size 6 × 1. ⇐
10. Consider the following code:

```
myArray = [ 1:10 ];
smallNumbers = myArray < 5;
smallNumbersArray = myArray(smallNumbers);
```

Indicate whether each statement is true or false.
- (T/F) `smallNumbersArray` has length 5.
- (T/F) `smallNumbers` is of class logical.
- (T/F) The code above generates an error.
- (T/F) The relational operation in the second line of code is evaluated elementwise.

11. Given `myArray = [ 10, 20, 30, 40, 50 ]`, match the statement in the left column with the resulting array in the right column.

<table>
<thead>
<tr>
<th>Left Statement</th>
<th>Right Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>myArray(end-1:-1:1)</td>
<td>[40, 30, 20, 10]</td>
</tr>
<tr>
<td>myArray(2,5)</td>
<td>error</td>
</tr>
<tr>
<td>myArray(myArray&lt;1)</td>
<td>empty matrix</td>
</tr>
<tr>
<td>myArray(6)</td>
<td>error</td>
</tr>
</tbody>
</table>

12. Consider the following code:

```
apples = 3;
oranges = 2;
pears = 1;
```

Indicate whether each statement is true or false.
- (T/F) `apples <= oranges+1`
- (T/F) `apples = oranges+1`
- (T/F) `(apples > oranges) & (pears < 2)`
- (T/F) `(apples < oranges) | (pears < 2)`
- (T/F) `(pears & apples) < oranges`

13. Consider two floating point numbers, x and y, whose values lie between 1 and 10. Which of the following might be a good way to test for equality of x and y?

(a) `abs(x-y) < 1`
(b) `x == y`
(c) `y-x < .00001`
(d) `abs(x+y) < .00001`
(e) None of the above.

14. Fill in the following truth tables.

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>and(X,Y)</th>
<th>X</th>
<th>Y</th>
<th>xor(X,Y)</th>
<th>X</th>
<th>Y</th>
<th>or(not(X),Y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
15. Let

\[ x = [1, 2, 3, 4]; \]
\[ y = [-1, 0, 1, 2]; \]

For each expression below, use the space provided to write the value of the expression or **error** if the expression is not valid Matlab code.

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x .* y )</td>
<td>([-1, 0, 3, 8])</td>
</tr>
<tr>
<td>( x .^ y )</td>
<td>([1, 1, 3, 16])</td>
</tr>
<tr>
<td>( x * y )</td>
<td><strong>error</strong></td>
</tr>
<tr>
<td>( x * y' )</td>
<td>10</td>
</tr>
<tr>
<td>( x ./ y )</td>
<td>([-1.0000, 0, .3333, .5000])</td>
</tr>
<tr>
<td>( x .+ y )</td>
<td><strong>error</strong></td>
</tr>
<tr>
<td>( x - 2 )</td>
<td>([-1, 0, 1, 2])</td>
</tr>
</tbody>
</table>

16. For each expression below, use the space provided to write the value of the expression or **error** if the expression is not valid Matlab code.

<table>
<thead>
<tr>
<th>Expression</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>not([0, 1, 1, 0])</code></td>
<td>([1, 0, 0, 1])</td>
</tr>
<tr>
<td><code>[0, 1, 0] &amp; [1, 0, 0]</code></td>
<td>([0, 0, 0])</td>
</tr>
<tr>
<td>`[0, 1, 0]</td>
<td>1`</td>
</tr>
<tr>
<td><code>not([0, 1, 0]) &amp; 1</code></td>
<td>([1, 0, 1])</td>
</tr>
</tbody>
</table>

17. Given the array `myArray = [-1, 2, 0, 4, -5]`, which statement sets all of the negative elements of `myArray` to 0?

(a) `myArray = (myArray >= 0) .* myArray` ⇐
(b) `myArray = (myArray < 0) .* myArray`
(c) `myArray = (myArray ~= 0) .* myArray`
(d) `myArray(myArray<0) = myArray & 0`
(e) None of the above.
18. Which code generated the figure shown?

(a) \[ x = [0:1:2\pi]; \]
    \[ y = \sin(x); \]
    \[ z = \cos(x); \]
    \[ \text{hold on} \]
    \[ \text{plot}(x,y,'-k'); \]
    \[ \text{plot}(x,z,'-k'); \]
(b) \[ x = [0:.01:2\pi]; \]
    \[ y = \sin(x); \]
    \[ z = \cos(x); \]
    \[ \text{plot}(x,y,'-k'); \]
    \[ \text{plot}(x,z,'-k'); \]
(c) \[ x = [0:.05:2\pi]; \]
    \[ y = \sin(x); \]
    \[ z = \cos(x); \]
    \[ \text{plot}(x,y,'-k'); \]
    \[ \text{plot}(x,z,'-.k'); \]
(d) \[ x = [0:.05:2\pi]; \]
    \[ y = \sin(x); \]
    \[ z = \cos(x); \]
    \[ \text{hold on} \]
    \[ \text{plot}(x,y,'-k'); \]
    \[ \text{plot}(x,z,'-.k'); \]

19. Consider the following code.

\[ x = [1, 2, 3, 4]; \]
\[ y = 7; \]
\[ z = 1; \]
\[ \text{save } \text{MyFile} \ y \ z; \]

Which statement is true?

(a) All of the variables in the workspace are saved to a file \text{MyFile.mat}.
(b) \text{load MyFile} loads the value of \( x \) into the workspace.
(c) \text{load MyFile} \ z loads only the value of \( z \) into the workspace. \[ \text{⇐} \]
(d) The code above would generate an error.
(e) None of the above.

20. Indicate whether each statement is true or false.

(T/F) \( \text{myName} = 'Sam'; \text{myName}(4) = 'e' \) results in \( \text{myName} == 'Sam' \).

(T/F) The command \text{uint8('ABC')} will display the ASCII code for the letters A,B, and C.
(T/F) `char('Hello', 'my', 'name', 'is', 'Sam')` will create another string by concatenating the input strings.

(T/F) The code `'Sam' == 'sam'` generates `true`.

(T/F) The code `lower('Sam')` == `'sam'` generates `true`.

21. Consider the following code:

```matlab
myString = 'Hello, how are you doing 2day?'
isspaceArray = isspace(myString);
myString2 = myString(~isspaceArray)
myString3 = myString(isletter(myString))
```

Which statement is `false` after the code runs?

(a) `myString2` == `Hello, how are you doing 2day?`
(b) `myString3` has 22 elements.
(c) `strrep(myString2, '2', 'to')` results in the string `'Hello, how are you doing today?'`
(d) `find(myString == 'h')` returns the array `[1, 8]`. ⇐
(e) `lower(myString2)` == `'hello, how are you doing 2day?'`.

22. Match the statement in the left column with the resulting value in the right column.

| myArray = [ 1, 2, 3, 4; 5, 6, 7, 8 ] | [ 1 2 3 4 ]
|-----------------------------------|------------------|
| myArray = [ 1, 2, 3, 4; 5, 6, 7, 8]' | [ 1 5 ]
| myArray = [ [1, 2]; [4, 5, 6] ]   | [ 2 6 3 7 ]
| myArray = [ 1, 2, 3; 5, 6, 7 ]; myArray(2,1) = 4 | error |

23. For 2D arrays, which statement is `false`?

(a) Row-column indexing uses two numbers to index the array while linear indexing uses one number.
(b) If the linear index of element `(m,n)` is `k`, then the linear index of element `(m,n+1)` is `k+1`. ⇐
(c) The second column of a `3 x 2` array `myArray` would be accessed as `myArray(:,2)`.
(d) The first row of a `3 x 2` array `myArray` would be accessed as `myArray(1,:)`.
(e) `For x = [ 1, 2, 3 ]` and `y = [ 1, 2 ]`, `x' * y` yields a `3 x 2` array.

24. Let

```matlab
myArray = [ 1, 2, 3, 4; 5, 6, 7, 8; 9, 10, 11, 12]
```

Which statement is `false`?

(a) `myArray(:,3)` = `[]` deletes the third column of `myArray`.
(b) `myArray(1:2,3:4)` gives the array `[ 3, 4; 7, 8 ]`. 
(c) \( \text{myArray([3;-1:1],1)} \) gives the array \([ 9, 5, 1 ]\).

(d) \( \text{myArray([1, 1, 1],1)} \) gives an error. ⇐

(e) \( \text{myArray([1:2:3],[1:2])} \) is equivalent to \( \text{myArray(1:2:3,1:2)} \).

25. Let

\[
\text{myArray} = \begin{bmatrix}
10 & 20 & 30 & 40 \\
-5 & 6 & 0 & 1 \\
1 & -2 & 11 & 3
\end{bmatrix}
\]

Indicate whether each statement is true or false.

\begin{align*}
(\text{T/F}) & \quad \text{myArray2 = myArray(,:) results in myArray2 having size 12 × 1.} \\
(\text{T/F}) & \quad \text{myArray2 = reshape(myArray,4,3) results in myArray2 having size 4 × 3.} \\
(\text{T/F}) & \quad \text{myArray2 = reshape(myArray,2,2) results in myArray2 having size 2 × 2.} \\
(\text{T/F}) & \quad \text{The statement myArray(,:) = myArray(,:) + [1:12]'} \quad \text{will change the shape of myArray.} \\
(\text{T/F}) & \quad \text{reshape(myArray,12,1) yields a 1D column array.}
\end{align*}