OC 512 MATLAB *Class 2 Assignment*

Julia Marks Peterson

1. Use a method of your choice to create the row vector x having 100 regularly spaced values starting exactly at 6 and ending exactly at 39.

```
x=[6:0.333:39]
```

```
x = 1×100
    6.0000    6.3330    6.6660    6.9990    7.3320    7.6650    7.9980    8.3310 ···
```

```
x = linspace(6,39,100)
```

```
x = 1×100
    6.0000    6.3333    6.6667    7.0000    7.3333    7.6667    8.0000    8.3333 ···
```

2. Use a method of your choice to create the column vector y having a regular spacing of 0.25 starting at -3 and ending at 12.

```
y = [-3:0.25:12]
```

```
y = 1×61
   -3.0000   -2.7500   -2.5000   -2.2500   -2.0000   -1.7500   -1.5000   -1.2500 ···
```

```
size(y)
```

```
ans = 1×2
     1    61
```

3. Create a vector x having six values starting at 0 and ending at 5. Create a matrix A whose first row is x, second row is 2x and third row is 3x + 10.

```
x = [0:5]
```

```
x = 1×6
     0     1     2     3     4     5
```

```
size(x)
```

```
ans = 1×2
     1     6
```

```
A = [x; 2*x; 3*x+10]
```

```
A = 3×6
     0     1     2     3     4     5
     0     2     4     6     8    10
    10    13    16    19    22    25
```

4. Create the matrix Create the vector c that consists of the third row of a. Create the vector d that consists of the second column of A. Create a 1x2 array e that consists of the first and second rows of

A. Create a 2x2 array that consists of the 4 corner elements of A. Do all this by using indexing. Do NOT simply type in the numbers!

```
A = [3 5 9; 6 37 1; 2 8 6]
```

```
A = 3×3
     3     5     9
     6    37     1
     2     8     6
```

```
c = [A(3,:)]
```

```
c = 1×3
     2     8     6
```

```
d = [A(:,2)]
```

```
d = 3×1
     5
    37
     8
```

```
e = [A(1) A(2)]
```

```
e = 1×2
     3     6
```

```
f = [A(1) A(7); A(3) A(9)]
```

```
f = 2×2
     3     9
     2     6
```

5. For the above matrix A, use the 'sort' function to create two new matrices; one with each column sorted and one with each row sorted.

```
A
```

```
A = 3×3
     3     5     9
     6    37     1
     2     8     6
```

```
Arow = sort(A,2)
```

```
Arow = 3×3
     3     5     9
     1     6    37
     2     6     8
```

```
Acol = sort(A,1)
```

```
Acol = 3×3
     2     5     1
     3     8     6
     6    37     9
```

6. Given the two matrices = [6 7; 2 9] and = [-9 3; 7 5], use element-by-element math to add, subtract, multiply, and divide C and D. Next, create new matrices by vertically and horizontally concatenating C and D.

```
C = [6 7; 2 9]
```

```
C = 2×2
     6     7
     2     9
```

```
D = [-9 3; 7 5]
```

```
D = 2×2
    -9     3
     7     5
```

```
Sum = C + D
```

```
Sum = 2×2
    -3    10
     9    14
```

```
Sub = C - D
```

```
Sub = 2×2
    15     4
    -5     4
```

```
Prod = C*D
```

```
Prod = 2×2
    -5    53
    45    51
```

```
Quot = C/D
```

```
Quot = 2×2
    0.2879    1.2273
    0.8030    1.3182
```

```
E = [C D]
```

```
E = 2×4
     6     7    -9     3
     2     9     7     5
```

```
F = [C; D]
```

```
F = 4×2
     6     7
     2     9
    -9     3
     7     5
```

7. Using a method of your choice, create the following 2x2 cell array

a. How would you access the "l" in "Matlab" as a character?

b. How would you access the word "Simulink" as a character array? As a cell?

c. How would you access the 8 in each of the two numeric arrays?

```
A = {'Matlab' 'Simulink'; [3 9; 8 2] [2;8;5]}
```

```
A = 2×2 cell
```

	1	2
1	'Matlab'	'Simulink'
2	[3,9;8,2]	[2;8;5]

```
b = A{1,1}(4)
```

```
b =
'l'
```

```
c = A{1,2} %call a cell
```

```
c =
'Simulink'
```

```
c2 = char(A{3}) %call a character
```

```
c2 =
'Simulink'
```

```
d = [A{2,1}(2) A{2,2}(2)]
```

```
d = 1×2
     8     8
```

8. Let us go back to the idea of structures. Let us put together some data for CE 640 and OC 512. We could do this:

```
oc512student(1).firstname='Greg';
oc512student(2).firstname='Mary';
oc512student(1).lastname='Jones';
oc512student(2).lastname='Smith';
oc512student(1).examscore=[96 95];
oc512student(2).examscore=[87 75];

ce640student(1).firstname = 'Lucille';
ce640student(2).firstname = 'Bob';
ce640student(3).firstname = 'Didi';
ce640student(1).lastname = 'Bluthe';
ce640student(2).lastname = 'Belcher';
ce640student(3).lastname = 'Pickles';
ce640student(1).examscore = [50 50];
ce640student(2).examscore = [70 90];
ce640student(3).examscore = [89 78];


oc512student
```

oc512student = 1×2 struct

Fields	firstname	lastname	examscore
1	'Greg'	'Jones'	[96,95]
2	'Mary'	'Smith'	[87,75]

```
ce640student
```

ce640student = 1×3 struct

Fields	firstname	lastname	examscore
1	'Lucille'	'Bluthe'	[50,50]
2	'Bob'	'Belcher'	[70,90]
3	'Didi'	'Pickles'	[89,78]

```
combined_students = [oc512student ce640student]
```

combined_students = 1×5 struct

Fields	firstname	lastname	examscore
1	'Greg'	'Jones'	[96,95]
2	'Mary'	'Smith'	[87,75]
3	'Lucille'	'Bluthe'	[50,50]
4	'Bob'	'Belcher'	[70,90]
5	'Didi'	'Pickles'	[89,78]