LEARNING OBJECTIVES:
1. Matrix Multiplication. What’s it used for?
2. While and for loops
3. Working with scripts and functions

REMINDER: You may ask the course TA and Tutors for assistance. You must complete this assignment without looking at other student’s code or copying solutions from any source.

Description:
This assignment consists of following a series of instructions and reporting on outcomes.

Lab Instructions:
"Login and set up same environment as Lab#1 (cs7wXX ieng6 home directory, Notepad++, MATLAB). Refer to Lab#1".

PART ONE: WORKING WITH SIMPLE MATRIX MULTIPLICATION
1. MATLAB stands for MATrix LABoratory so at some point, I’m sure you guessed we’d be learning more about matrix operations. This week, we’re going to focus on MATRIX multiplication. Two obvious questions are. How does it work (how’s is it different from scalar multiplication) and when do we use it?

2. Let’s start by looking at how matrix multiplication works. Let’s start by simply considering the dimensions necessary in order for matrix multiplication to work correctly. Consider the figure below.

3. Notice that the inner dimensions must be equal. In the example above, the number of columns in the first matrix is 8 and that must be equal to the number of rows in the second matrix (also 8).

4. Also, notice that the output matrix, C, has the same number of rows as the first matrix and the same number of columns as the second matrix.

5. Now let’s look a simple example. Type the following into a new script called mat_multiply.m.

   ```matlab
   A = [1 2 3; 2 3 4; 3 4 5]
   B = [1; 0; 0]
   ```

6. QUESTION #1. What happens when you type C=A*B? What are the dimensions of the resulting matrix and what are the values?
7. So what's actually going on? Let's think of matrix $A$ in terms of its sub-components

![Matrix A](image)

$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$

$B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$

$C = \begin{bmatrix} \sum_{k=1}^{n} a_{ik} b_{kj} \end{bmatrix}$

8. The above diagram shows what's going on. Each row of matrix $A$ is multiplied by each column of matrix $B$.

9. In our case, because $b_2$ and $b_3$ are both set to zero, we simply get $[a_{11} \; a_{21} \; a_{31}]$ as the result of our matrix.

10. Reset $B$ so that it now is equal to $[1; 1; 1]$

11. **QUESTION #2** Now what happens when you multiply $A$ and $B$? Can you explain why? What is in each row of the output matrix $C$?

12. **QUESTION #3** In regular scalar multiplication $x*y = y*x$; this is called the **commutative property**. For example, $9*5 = 5*9 = 45$. Is it the same with matrices? Try executing $B*A$ in MATLAB?

13. Now reset $B$ one more time so that it is equal to:

   1. $B = [1 \; 0 \; 0; 0 \; 1 \; 0; 0 \; 0 \; 1]$

14. **QUESTION #4** What is the resulting output of $A*B$? How about $B*A$? What is special about this particular $B$ matrix?

15. Before leaving, type the following at the command prompt and read:

   ```
   >>doc eye
   ```

16. Congratulations, you've just used the following complicated matrix multiplication equation for multiplying two matrices $A$ and $B$ when performing $C = A*B$. All it means is that you are multiplying each element in the row $i$ of Matrix $A$ with every element in column $j$ of Matrix $B$. And then sum then goes in position $i,j$ of Matrix $C$.

   $$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

17. The output from the diagram would go in row 1, column 1 of matrix $C$.
PART TWO: SHOPPING WITH MATRICES

1. So we answer the first part about how matrix multiplication works, but it is still not immediately clear why it is useful? To see why, let’s use an example. Suppose we are at a fashion store where they sell shoes for $75, skirts for $30, blouses for $23 and handbags for $200.

2. Now, imagine that we purchase 1 of each item. What is the total amount of money that we spend?

3. We could simply do $1 \times 75 + 1 \times 30 + 1 \times 23 + 1 \times 200 = $328

4. Alternatively, we can create two matrices as follows:
   \[ A = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}; \] % this represents the number of items we purchased
   \[ B = \begin{bmatrix} 75; 30; 23; 200 \end{bmatrix}% this represents the cost of each item\]

5. Now, we can simply multiple \( A \times B \) to get the same total. Did it work?

6. QUESTION #5 How would you modify one of the matrices so that if another customer came along and purchased 5 pair of shoes, 3 skirts, 4 blouses and 3 handbags? we could also calculate their total order? Do it. HINT: Which matrix contains the number of items?

7. QUESTION #6. Using the magic command in MATLAB to set \( A \) to magic(4) and then complete the multiplication one more time. Report both the matrix generated by the magic(4) command as well as the output of the multiplication.

PART THREE: MORE PRACTICE WITH IF/ELSE

1. As practice, let’s write a simple function that uses if/else to print your top three places to live.
2. By now you should know how to write and save a function by looking at the function header:

   \[
   \text{function print_preferences(pref)}
   \]

3. Just like before, we use the keyword function, followed by the output arguments followed by the function name and then the input arguments in parentheses.

4. QUESTION #7: What is the function name? What is the input? What do you notice about the output argument for this function (is there one)?

5. Let’s use numbers to represent our preferences.

6. Now let’s create some local variables representing our preferences. You may use our example as given or you can use your actual preferences.
   \[
   \begin{align*}
   \text{first} &= \text{‘San Diego’}; \\
   \text{second} &= \text{‘New York’}; \\
   \text{third} &= \text{‘Portland’};
   \end{align*}
   \]

7. Local variables mean they exist only within the function, and you CANNOT use them in your regular workspace.
8. I am going to give you the pseudo code for this function. If you copy and paste the following into Matlab, it will not work so try your best to implement it yourself. Recall the correct syntax for writing functions and if-else statements. Raise your hand if you have no clue how or where to start…

```matlab
function print_preferences(pref)
    create local variable first = 'San Diego';
    create local variables for second and third as well

    if pref is 1
        display 'My first choice is: ' 
        display variable first
    
    else if pref is 2
        display 'My second choice is: ' 
        display variable second
    
    else if pref is 3
        display 'My third choice is: ' 
        display variable third
    
    else
        display 'I do not recognize that choice!'
```

9. QUESTION #8: Copy and paste your print_preferences function into your text file. Now let’s test it. Call the function with 1, 2, and 3 as inputs in the command prompt and explain what happens for each input.

PART FOUR: A FUNCTION WITH A SWITCH

1. Switch statements are similar to if/elseif statements in that they allow us to check a condition and only evaluate certain parts of code depending on the result of the condition (whether it is 0 or 1, aka true or false)

2. Let’s write a simple function that uses the switch statement instead of if-else to print out our food preferences depending on what is given as input to the function (as a string this time!)

3. The function header for this function is given below

```matlab
function print_preferences_case(prefString)
```

4. Same as the above, let’s create some local variables representing our food preferences. You may use our example as given or you can use your actual preferences.

```matlab
first = 'San Diego';
second = 'New York';
third = 'Portland';
```
5. Now for the case-switch part. We are going to use case-switch to determine which preference to print.

6. Once again I am going to give you the pseudo code for this function. If you copy and paste the following into Matlab, **it will not work so try your best to implement it yourself.** Recall the correct syntax for writing functions and the switch statement.

```matlab
function print_preferences_case(prefString)
    create local variable first = 'San Diego';
    create local variables for second and third as well

    switch prefString
        'First' case
            display 'My first choice is: '
            display variable first

        'Second' case
            display 'My second choice is: '
            display variable second

        'Third' case
            display 'My third choice is: '
            display variable third

        otherwise
            display 'I do not recognize that choice!'
    end
end
```

7. Otherwise is optional and only executes when no other case is true. What case might we want to include in the otherwise statement? Should we prompt the user to try again?

8. **QUESTION #9:** Copy and paste your print_preferences_case function into the text file. Now let's test it. Enter each of the following cases in the command prompt and explain what happens

```
>>print_preferences_case('First')
>>print_preferences_case('Second')
>>print_preferences_case('Third')
>>print_preferences_case('first')
```

9. **QUESTION #10.** What happened with the last one? And why? How can we improve our case method (you don’t have to do it, but just make a suggestion).

10. **QUESTION #11.** Also, think about what happens if some programmer who had no idea what our function does typed print_preferences_case('january') or print_preferences_case('turkey'). Clearly, he has no idea how to use our function, but what does our function do in these cases?

**PART FIVE: SAVING and SUBMITTING**
1. SUBMIT *FOUR* THINGS: Save your Lab7_Lastname, and then also save a copy of your mat_multiply.m, your function file for print_preferences file and print_preferences_case file. You are welcome to stay and work on the homework assignment portion.

**PART SIX: LAB #7 CHECKOFF CHECKLIST**

To receive credit for this lab you need to:

- Write your name on the whiteboard, when you are finished.
- Show your TA/Tutor your Notepad++ document in your ieng6 cs7wXX home directory folder.
- Be prepared to show the TA/Tutor all questions answered in this document.
- Be able to answer questions about the MATLAB environment, case statements, matrix multiplication and more

Do not leave until you have seen a TA/Tutor mark your name down in autograder.

*It is your responsibility to make sure you get credit for each lab!*