Name: Jingsi Zhou
Date: 09/14/2020
Section: 17460
Assignment: Lab 1

Problem 1.1 For each of the names above, write down why they aren't valid in MATLAB

The names of variables must start with a letter (upper or lowercase) and can only contain numbers, letters, and underscores. Thus, 2more is not a valid variable because the name begins with a number. The variable i-p-freely contains dashes in its name, which are not in the list of allowed characters that a variable name can contain. Similarly, lucky# has a '#' in its name, which is also not in the list of allowable characters.

Problem 1.2 Assign 6.67×10^{-11} to the name grav_constant (in $N \cdot m^2/kg^2$).
Copy/paste what you typed at the prompt and MATLAB's response.

```
>> grav_constant = 6.67e-11

grav_constant =

  6.6700e-11
```

Problem 1.3 Assign a value with a real component of 4×10^{-5} and an imaginary component of 3×10^{-3} to a variable named x. Copy/paste what you typed at the prompt and MATLAB's response. Is it as accurate as you thought it would be? What command would you issue to make the output more accurate?

```
>> x = 4e-5 + 3e-3*i

x =

   0.0000 + 0.0030i
```

MATLAB's response was not as accurate as I thought it would be. The command I would use to make the output more accurate "format long" or, to have the number in a more readable form, "format shorte."

Problem 1.4 MATLAB uses double precision for this and all variables. Copy/paste the value of pi to 15 decimal places using the format command.

```
>> format long
>> pi

ans =

  3.141592653589793
```

Problem 1.5 Type in the lines above and then assign $y^x + w*w + v$ to t. Copy/paste your results.

```
>> t = y^x + w * w + v
```

```
t =

    4.993600000000000e+02
```

Problem 1.6 Using the variable definitions above, which two commands below result in the same value? (Try to answer before typing it into MATLAB!)

```
>> y-(57^1.1)/10+a
>> y-57^1.1/10+a
```

Problem 1.7 Using all that you have read till now, write the set of commands that gets two numbers from the user and displays the sum of the numbers in the following manner:
"You entered x and y which sum to z"
Where instead of x, y and z you will display the actual user input and result. Copy/paste the commands and corresponding outputs.

```
>> x = input('Enter your first number: ');
Enter your first number: 6
>> y = input('Enter your second number: ');
Enter your second number: 3
>> z = x + y;
>> str = ['You entered ', num2str(x),' and ', num2str(y), ' which sum to ', num2str(z)];
>> disp(str)
You entered 6 and 3 which sum to 9
```

Problem 1.8 Repeat the previous problem (number addition) but this time, store the user input in a mat file with a safe name and read it back to perform the addition and store the output (sum) in a different mat file. Copy/paste all the commands and corresponding output(s).

```
>> x = input('Enter your first number: ');
Enter your first number: 4
>> y = input('Enter your second number: ');
Enter your second number: 5
>> save('vars.mat')
>> clear
>> load('vars.mat')
>> z = x + y;
>> save('result.mat','z')
```
