```
% VectorOfIntegers.m
% Patrick Utz, 2/9/18, 5.1

% Problem: Write a MATLAB script file to create and return a vector of
% integers from m to n (where m is the first input argument and n is the
% second), regardless of whether m is less than n or greater than n. If
% m is equal to n, the ?vector? will just be 1 x 1 or a scalar. Test your
% program using three different cases, m>n, m<n and m=n, respectively.
% Attach the results.

% Variables: m = first integer of the vector, n = last integer of the
% vector, intVector = vector created using m and n

% Algorithm:
% input m as starting value and n as ending value
% check to see if m > n, or m < n or m = n
% output the appropriate vector
% stop


clear
m = input('Please enter the first integer of the vector: ');
n = input('Please enter the last integer of the vector: ');
if m > n
    intVector = m:-1:n;
    disp(intVector)
elseif m < n
    intVector = m:1:n;
    disp(intVector)
else
    intVector = m
    fprintf('%d',intVector)
end
```

```
>> VectorOfIntegers
Please enter the first integer of the vector: 1
Please enter the last integer of the vector: 9
     1     2     3     4     5     6     7     8     9
>> VectorOfIntegers
Please enter the first integer of the vector: 9
Please enter the last integer of the vector: 1
     9     8     7     6     5     4     3     2     1
>> VectorOfIntegers
Please enter the first integer of the vector: 2
Please enter the last integer of the vector: 2
intVector =
     2
```

```
% RandomIntOddEvenChecker.m
% Patrick Utz, 2/9/18, 5.2

% Problem: Write a script that will generate one random integer, and
% will print whether the random integer is an even or an odd number.
% (Hint: an even number is divisible by 2, whereas an odd number is not;
% so check the remainder after dividing by 2). Test your program by
% running it at least twice, one with an even number, and another with
% an odd number. Attach the results.

% Variables: randInt = random integer created in the range between
% 1 and 100

% Algorithm:
% create random integer
% check to see if the integer is odd or even by dividing by two
% output whether the integer is odd or even
% stop


clear
randInt = randi([1,100],1,1);
if rem(randInt,2) == 0
    fprintf('The random integer %d is an even number \n', randInt);
else
    fprintf('The random integer %d is an odd number \n', randInt);
end
```

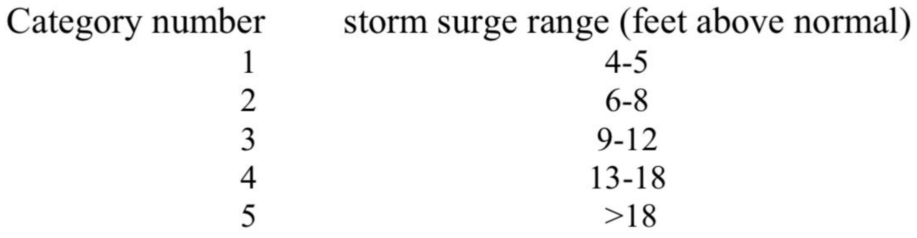
```
>> RandomIntOddEvenChecker
The random integer 66 is an even number

>> RandomIntOddEvenChecker
The random integer 41 is an odd number
```

```
% pHScaleTester.m
% Patrick Utz, 2/9/18, 5.3

% Problem: In chemistry, the pH of an aqueous solution
% is a measure of its acidity. The pH scale ranges from 0 to 14,
% inclusive. A solution with a pH of 7 is said to be neutral, a solution
% with a pH greater than 7 is basic, and a solution with a pH less than 7
% is acidic. Write a script that will prompt the user for the pH of a
% solution, and will print whether it is neutral, basic, or acidic. If
% the user enters an invalid pH, an error message will be printed. Test
% your program by running it five times, with the following inputs: -1;
% 3; 7; 10; and 18. Attach the results.

% Variables: pH = input value of pH

% Algorithm:
% input value of pH from user
% check to see if pH is within range
% check to see what the pH value corresponds to on the scale
% output what the pH value is on the scale
% stop

clear
pH = input('Hello! Please enter a valid pH value to see where it lies on the scale: ');

if pH < 0 || pH > 14
    fprintf('%d is an invalid pH value please enter a valid one. \n', pH);
elseif pH >= 0 && pH < 7
    fprintf('A pH of %d is considered acidic. \n', pH);
elseif pH > 7 && pH <= 14
    fprintf('A pH of %d is considered basic. \n', pH);
else
    fprintf('A pH of %d is considered neutral. \n', pH);
end
```

```
>> pHScaleTester
Hello! Please enter a valid pH value to see where it lies on the scale: -1
-1 is an invalid pH value please enter a valid one.
>> pHScaleTester
Hello! Please enter a valid pH value to see where it lies on the scale: 3
A pH of 3 is considered acidic.
```

```
>> pHScaleTester
Hello! Please enter a valid pH value to see where it lies on the scale: 7
A pH of 7 is considered neutral.
>> pHScaleTester
Hello! Please enter a valid pH value to see where it lies on the scale: 10
A pH of 10 is considered basic.
>> pHScaleTester
Hello! Please enter a valid pH value to see where it lies on the scale: 18
18 is an invalid pH value please enter a valid one.
```

```
% StormIdentifier.m
% Patrick Utz, 2/9/18, 5.4

% Whether a storm is a tropical depression, tropical storm,
% or hurricane is determined by the average sustained wind speed.
% In miles per hour, a storm is a tropical depression if the winds
% are less than 38 mph. It is a tropical storm if the winds are between
% 39 and 73 mph, and it is a hurricane if the wind speeds are >= 74 mph.
% Furthermore, hurricanes can be categorized based on storm surges. Write
% a script that will prompt the user for the wind speed, and will print
% what type of storm it is. If it is a hurricane, then your program
% should prompt the user to enter the storm surge, and will print which
% category the hurricane is. Your program should print an error message
% if the user inputs are not in the correct range. Test your program by
% running it through all possible cases, including an invalid wind speed,
% and an invalid storm surge. Attach the results.

% Variables: windSpeed = wind speed of storm from the user, surgeRange =
% storm surge range inputted by user

% Algorithm:
% input value wind speed
% check to see if it is a valid speed and then return what type of storm it is
% if it is a hurricane input the surge range, else output the type of storm and end the program
% check to see what category the hurricane is
% output the appropriate category of the hurricane
% stop
```



Category number	storm surge range (feet above normal)
1	4-5
2	6-8
3	9-12
4	13-18
5	>18

```
clear
windSpeed = input('Please enter the average sustained wind speed: ');
if windSpeed < 0 || isletter(windSpeed)
   fprintf('This is not a valid wind speed. Please enter a valid speed. \n');
else if windSpeed >= 0 && windSpeed < 38
   fprintf('You are currently in a tropical depression. \n');
else if windSpeed >= 39 && windSpeed <= 73
   fprintf('You are currently in a tropical storm. \n');
else if windSpeed >= 74
```

```
    fprintf('You are currently in a hurricane. \n \n');
    surgeRange = input('Please enter the storm surge: ');
    if surgeRange < 4
        fprintf('Invalid number. Please enter a valid storm surge. \n');
    elseif surgeRange >= 4 && surgeRange <= 5
        fprintf('This is a category 1 hurricane. \n');
    elseif surgeRange >= 6 && surgeRange <= 8
        fprintf('This is a category 2 hurricane. \n');
    elseif surgeRange >= 9 && surgeRange <= 12
        fprintf('This is a category 3 hurricane. \n');
    elseif surgeRange >= 13 && surgeRange <= 18
        fprintf('This is a category 4 hurricane. \n');
    else
        fprintf('This is a category 5 hurricane. \n');
    end
    end
end
end
end
```

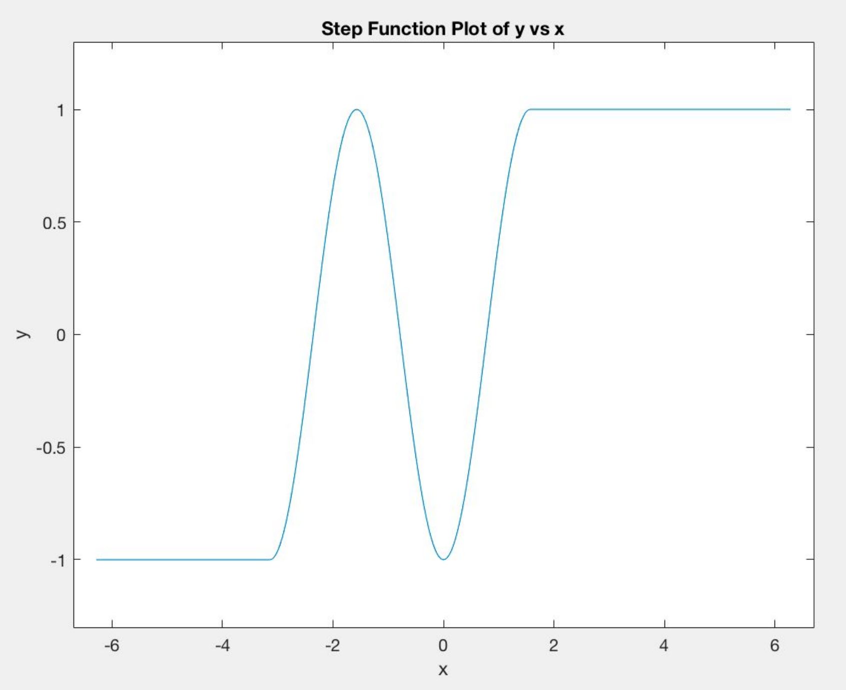
```
>> StormIdentifier
Please enter the average sustained wind speed: 37
You are currently in a tropical depression.
>> StormIdentifier
Please enter the average sustained wind speed: 42
You are currently in a tropical storm.
>> StormIdentifier
Please enter the average sustained wind speed: 76
You are currently in a hurricane.
Please enter the storm surge: 4.5
This is a category 1 hurricane.
>> StormIdentifier
Please enter the average sustained wind speed: 76
You are currently in a hurricane.
Please enter the storm surge: 7
This is a category 2 hurricane.
>> StormIdentifier
Please enter the average sustained wind speed: 78
You are currently in a hurricane.
Please enter the storm surge: 10
This is a category 3 hurricane.
```

```
>> StormIdentifier
Please enter the average sustained wind speed: 78
You are currently in a hurricane.
Please enter the storm surge: 17
This is a category 4 hurricane.

>> StormIdentifier
Please enter the average sustained wind speed: 78
You are currently in a hurricane.
Please enter the storm surge: 20
This is a category 5 hurricane.
>> StormIdentifier
Please enter the average sustained wind speed: -2
This is not a valid wind speed. Please enter a valid speed.
>> StormIdentifier
Please enter the average sustained wind speed: 78
You are currently in a hurricane.
Please enter the storm surge: 2
Invalid number. Please enter a valid storm surge.
```

```
% PlottingFunction.m
% Patrick Utz, 2/9/18, 5.5

% Create and run a MATLAB script file to plot the following function
% for x over the range of [-2?, 2?] with step size of 0.01?.
% Show your program and result figure.
```

$$y(x) = \begin{cases} -1, & x \le -\pi \\ 2\sin^2(x) - 1, & -\pi < x \le \frac{\pi}{2} \\ 1, & x > \frac{\pi}{2} \end{cases}$$

```
% Variables: x = the x range, y = the step function split into three parts,
% k = a vector used to index through the x and y vectors to check for
% range and give the appropriate length to y for constant functions

% Algorithm:
% create range of x
% split the range into three parts using conditional statements and apply
% the appropriate y function for the respective range
% plot the function of y vs x
% give title and labels for x and y
% stop

clear
x = (-2*pi):(.01*pi):(2*pi);
y = -1*(x >= -2*pi & x <= -pi) + 1*(x > (pi/2) & x <= 2*pi) + (2*(sin(x)).^2 - 1).*(x > (-pi) & x <= (pi/2));
plot (x,y)
title('Step Function Plot of y vs x');
xlabel('x');
ylabel('y');
axis([-6.7 6.7 -1.3 1.3])
```


Step Function Plot of y vs x
y
x