

**Armstrong State University**  
**Engineering Studies**  
**MATLAB Marina – Exception Handling Exercises**

1. What is an exception?
2. How can a program or function handle exceptions?
3. What is a wrapper function? What type of functions should use wrapper functions for exception handling?
4. Write a MATLAB function named `geometricMean` to compute the geometric mean of a set of numbers. You may assume the set of numbers is a row vector. The geometric mean of a set of numbers  $x = (x_1, x_2, x_3, \dots, x_n)$  is  $G = (x_1 \cdot x_2 \cdot x_3 \cdot \dots \cdot x_n)^{\frac{1}{n}}$ ; in other words the product of all the elements raised to the power of one over the number of elements.
5. Determine an appropriate set of test cases and write a test program that verifies the correct operation of the `geometricMean` function written for Exercise 4.
6. Modify the `geometricMean` function written for Exercise 4 to include inline error handling for the test cases that generated run-time errors or erroneous results. Test the modified function using the same test program written for Exercise 5.
7. Write a function named `impulseResponse` that determines the M+1 point approximation of the impulse response of a discrete-time system. The formula for the impulse response approximation of the system is:  $h[n] = \frac{\sin(\hat{\omega}_c n)}{\pi n}, -\frac{M}{2} \leq n \leq \frac{M}{2}$  where n can only take on integer values. The function should take two arguments `whc` and `M` and should return the M+1 point impulse response. The function definition is given in Figure 1.

```
function hh = impulseResponse(whc, M)
```

Figure 1, `impulseResponse` Function Definition

8. Write a test program for the `impulseResponse` function written for Exercise 7. The test program should include the following test cases: `M = 10`, `M = 2`, `M = 7`, `M = -10`, and `M = 0`. Use `whc = pi/4` for each test case. Determine which test cases generate run-time errors or erroneous results.
9. Modify the `impulseResponse` function written for Exercise 7 to include inline error handling for the test cases that generated run-time errors or erroneous results. Test the modified function using the same test program written for Exercise 8.
10. Modify the `impulseResponse` function written for Exercise 7 to throw an exception and exit the function for the test cases that generated run-time errors or erroneous results. Test the modified function using the same test program written for Exercise 8.

Last modified Thursday, October 02, 2014



This work by Thomas Murphy is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported License](https://creativecommons.org/licenses/by-nc-nd/3.0/).