Mini Project #1 – Calculator Function

This mini-project requires you to create a MATLAB function that can perform various basic mathematical functions on two numbers. You should name your function **netIDcalculator**, where netID is your netID (as in aaa111). Your function should take three inputs in the following order:

`num1` – The first number in the calculation you want to perform.

`num2` – The second number in the calculation you want to perform

`indicator` – A variable, that has a data type of your choice[1], that the user will use to determine which operation to perform. **You should note in a comment how you are implementing the indicator variable.**

Your function should be able to perform the following operations:

- `num1 + num2` (addition)
- `num1 - num2` (subtraction)
- `num1 * num2` (multiplication)
- `num1 / num2` (division)
- `num1^num2` (exponent)
- $\log_{num2} num1$ (log base num2 of num1)[2]
- return the larger of `num1` and `num2`
- return the smaller of `num1` and `num2`

Note that every time this function is called, it should only perform ONE of these operations. However, the user should be able to choose from any of the operations with each call using the indicator input.

For example, if you were to use a string for your indicator[1] input, the call of:

`result = aaa111calculator(5,10, 'subtract');`

Would store the value of -5 in the variable `result`.

In another example, using a numerical variable for the indicator[1], and associating 6 with the logarithmic operation, a call of:

`result = aaa111calculator(125, 5, 6);`

Would store the value of 3 in the variable `result`.

Notes:

[1]It is your choice to determine how the indicator variable works, and there are quite a few ways to approach it and set up the associated conditional statements. Note that the above two examples are for two differently written functions: One that uses strings for the indicator, and one that uses numbers. If you are interested in strings, you should look into using the `strcmp` function.

[2]Note that `log(x)` in MATLAB takes the natural log, and `log10(x)` takes log base 10. For more information on changing the base of a logarithm, look here.