Advanced Scripting   
Bitwise Operators

Last Updated: 4/19/2024 5:32 PM Version 1  
Document Prepared for: CYBER360 Student

# Name Click here to enter name ID Click here to enter id

# Instructions

Answer all questions directly in this document. You will save and upload this completed document as your homework submission.

# Overview

This exercise presents a few practice drills involving PowerShell bitwise arithmetic operators. In the future we will use bitwise operators to help us with some networking tasks.

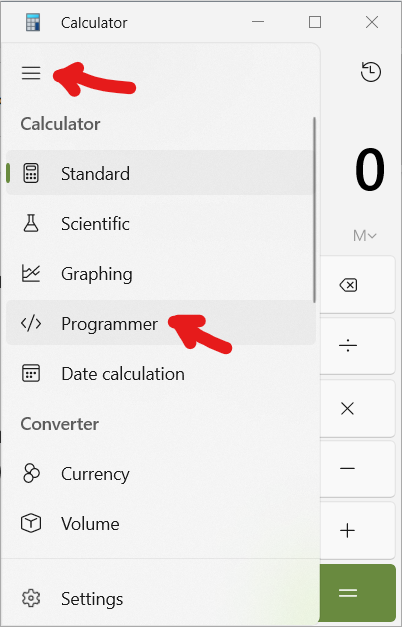
# Setup

## Requirements

* PowerShell
* The Windows Calculator app (or a similar advanced calculator app)

# Task —Bitwise Arithmetic Operators

In Microsoft Windows, launch the Calculator app. You can look for it in the start menu, or better yet, just enter **calc** at your PowerShell prompt. Then tap the **≡** (“hamburger” menu) in the top left corner, and change the calculator to “Programmer” mode. Of its four input methods HEX, DEC, OCT, and BIN, ensure that DEC (“decimal”) is activated, as illustrated here:

 A screenshot of a calculator

Description automatically generated

## Steps

1. Bitwise and
   1. At your powershell prompt, Enter  
      85 -band 170
   2. What was the output? Click or tap here to enter text.
   3. In your calculator app in Programmer mode, tap **CE** (clear entry) or **C** (clear), then enter **85**.
      1. What is 85 in hexadecimal, as shown beside HEX in the Calculator app? Click or tap here to enter text.
      2. What is 85 in binary, as shown beside BIN in the Calculator app? Click or tap here to enter text.
   4. In your calculator app in Programmer mode, tap **CE**, then enter **170**.
      1. What is 170 in hexadecimal, as shown beside HEX in the Calculator app? Click or tap here to enter text.
      2. What is 170 in binary, as shown beside BIN in the Calculator app? Click or tap here to enter text.
   5. Based on the binary numerals for 85 and 170, explain in your own words how the result you recorded in step 1.2 above was computed. Click or tap here to enter text.
   6. Verify that the Calculator app produces the same result:  
      tap **C**,  
      85,  
      tap **Bitwise,** **AND**,  
      170,  
      then tap **=**.   
      Your result: Click or tap here to enter text.
   7. If your result isn’t the same as your answer from step 1.2, check and do it again.
2. Bitwise or, bitwise xor
   1. At your powershell prompt, Enter  
      (204 -bor 170), (204 -bxor 170)
   2. What was the output? Click or tap here to enter text.
   3. Use the calculator app in programmer mode to find 204 in hex and binary. Based on the binary numerals for 204 and 170, explain in your own words how the results you recorded in step 2.2 above were computed. Click or tap here to enter text.
   4. Verify that the Calculator app produces the same results: tap **CE**, enter 204, tap **Bitwise,** **OR**, enter 170, then tap **=**. Repeat for Bitwise XOR.
3. Bitwise not
   1. At your powershell prompt, Enter  
      -bnot [long]3
   2. What was the output? Click or tap here to enter text.
   3. What datatype is the result? Click or tap here to enter text. *Hint: put the expression in parentheses, then call the* **GetType()** *method*: **(-bnot [long]3).GetType().name** . *Or you can save the result to a variable, and access the* **GetType()** *method on that variable.*
   4. Verify in your calculator app as follows: make sure the data type is set QWORD. (If it’s set to DWORD, WORD, or BYTE, tap that setting until it’s back to QWORD.) Then **CE**, 3, **Bitwise**, **NOT**. Examine the hex and binary numerals for your result. Explain in your own words how the result you recorded in step 3.2 above was computed. Click or tap here to enter text.
4. Bit shifting
   1. At your powershell prompt, Enter  
      (3 -shl 2), (170 -shr 1)   
      The expressions in this array are 3 shifted left twice and 170 shifted right once, respectively.
   2. Your output: Click or tap here to enter text.
   3. Verify in your calculator app:
      1. Tap the Bit Shift menu and make sure it’s set to either arithmetic or logical shift:   
         A screenshot of a computer

         Description automatically generated
      2. **CE**, **3**, **«**, **2**, **=**. Examine HEX and BIN numeral results.
      3. **CE**, **170**, **»**, **1**, **=**. Examine HEX and BIN numeral results.
   4. Based on the binary numerals for 3, 170, and the results you recorded in step 4.2 above, how were those bit-shifted results computed? Explain in your own words. Click or tap here to enter text.

# Deliverable

Upload this document with completed answers to I-Learn Canvas.