
**Homework # 10**

**01286131 Object-Oriented Programming**

**Software Engineering Program,**

**Department of Computer Engineering,**

**School of Engineering, KMITL**

By

65011277 Chanasorn Howattanakulphong

**Object-Oriented Programming Homework #10** **Apr 7th, 2023** **Objects and Programs (Part 2)**

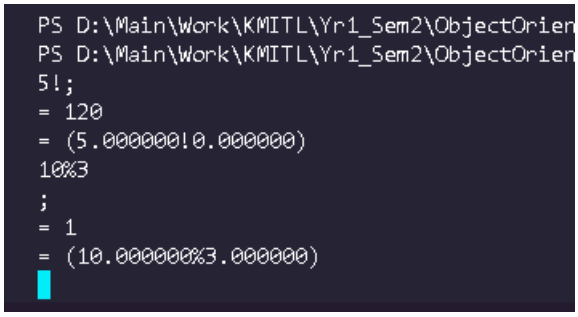
**1.** Use the supplement "Simple Calculator" project as a starting point for extending the calculator program.

**1.1)** Suppose that we want to support modulo (`%`) operator in the calculation expression. Modify grammar to allow expressions with modulo (`%`) operator.

**1.2)** Use the grammar in **1.1)** and modify the calculator program to allow expressions with modulo (`%`) operator and its calculation.

**1.3)** Suppose that we want to add a factorial operator by using a suffix `!` operator to represent "factorial". For example, the expression `7!` means `7 * 6 * 5 * 4 * 3 * 2 * 1`. Make `!` bind tighter than `*` and `/`; that is, `7*8!` means `7*(8!)` rather than `(7*8)!`. Write a new grammar to account for a higher-level operator.

**1.4)** Use the grammar in **1.3)** and modify the calculator program to support factorial calculation. To agree with the standard mathematical definition of factorial, let `0!` evaluate to `1`. As we deal with `double`s but factorial is defined only for `int`s, so just for `int` assign the `x` to an `int` and calculate the factorial of that `int`.



**2.** Create a program for evaluating bitwise logical expressions.

**2.1)** Write a grammar for bitwise logical expressions. A bitwise logical expression is much like an arithmetic expression except that the operators are ~ (complement), & (and), | (or), and ^ (exclusive or). Each operator does its operation to each bit of its integer operands. ~ is a prefix unary operator. A ^ binds tighter than a | (just as * binds tighter than +) so that `x|y^z` means `x|(y^z)` rather than `(x|y)^z`. The & operator binds tighter than ^ so that `x^y&z` means `x^(y&z)`.

**2.2)** Use the grammar in **2.1)** to create a program for evaluating bitwise logical expressions.

```
PS D:\Main\Work\KMITL\Yr1_Sem2\ObjectOrient
4\debugAdapters\bin\WindowsDebugLauncher.ex
e-Error-0homo2vp.ixs' '--pid=Microsoft-MIEn
Enter a bitwise logical expression: 1&0
Result: 0
PS D:\Main\Work\KMITL\Yr1_Sem2\ObjectOrient
4\debugAdapters\bin\WindowsDebugLauncher.ex
e-Error-zdrl2fo1.kq5' '--pid=Microsoft-MIEn
Enter a bitwise logical expression: 0|0
Result: 0
PS D:\Main\Work\KMITL\Yr1_Sem2\ObjectOrient
4\debugAdapters\bin\WindowsDebugLauncher.ex
e-Error-mxadxb25.0ft' '--pid=Microsoft-MIEn
Enter a bitwise logical expression: 1^1
Result: 0
```