Assignment Part 6 – Smalltalk

# Program Testing Procedure

Run program an observe output. Every 3rd line should be "Fizz", every 5th line "Buzz", every 15th line "Fizz buzz". The other lines should contain the line number.

# Weekly Question

The readability and writability of the Smalltalk version are better than the Fortran version, but worse than the Algol version. Fortran’s syntax is rather lacking in regularity and has unintuitive elements such as column-based syntax and output formatting, making the language difficult to learn, write, and understand. Smalltalk, while not having the most obvious nor intuitive syntax, is at least regular.  
Algol is a quite structured, regular procedural language, lending itself to the simple task of fizz buzz. Anyone familiar with basic procedural programming concepts such as loops and if statements would surely be able to write and understand the program. On the other hand, Smalltalk’s strong focus on object orientation is not particularly well suited to fizz buzz. A simple if-elseif-else chain must be built via object orientation abstractions, resulting in clumsy and less readable syntax (although there is probably a more advanced, cleaner way to do it that I did not think of). Smalltalk’s syntax is also not intuitive, meaning one will likely struggle to understand how the program works without being familiar with the language.

# Reflection

Smalltalk’s adherence to the regularity and orthogonality principles improves readability and writability. Smalltalk is based on the idea of sending “messages” to objects, from which functionality is achieved. The messages may have arguments, including code blocks. Despite this concept and the associated syntax not being particularly intuitive, it exhibits very high regularity and orthogonality, which does aid the readability of the program.