**Compilers Optimisation Coursework Report**

*Group Members: Galen Han, Alexander Xu, Jamie Law*

**Overview**

This report will deal with explaining the core algorithms and heuristics of our optimisation methods in ConstantFolder.java.

**Algorithms**

**Initalization**

Building upon the original provided code, we obtain the methods from the ClassGen. We then iterate through each method and call upon our optimise method function.

**Optimise method**

In this section, we perform our optimisation (by calling upon other functions) on each method (method one, method two etc. from each unoptimised java file).

Initially, the code is obtained from the method, following which we form the instruction list. A method generator is initialised with the original method as the baseline (in preparation for creating the optimised method).

We then use a counter and a while loop to perform arithmetic and comparison optimisations. This counter is incremented by the method return types; arithmetic optimisation and comparison optimisation both return an integer indicating how many optimisations have been made. This counter is reset every time the loop is re-entered. If the counter stays at 0 after one iteration, then the loop is exited as this indicates that no more optimisations can be made.