Analysis and Verification of Software
Homework 3

due by March 2, 2015

Tainting Analysis

- To deal with security issues, several programming languages have a notion of “tainted” data.
- The idea is that any value read from the outside environment is marked as being tainted, i.e., potentially dangerous. The results of any operations that use tainted data are also marked tainted.
- For this problem, assume that the available operations in our intermediate language are:
  - x = y op z — binary operation: x is tainted if either y or z or both are tainted
  - x = y — assignment: x is tainted if y is tainted
  - x = read() — read input: x is tainted
  - x = clean(y) — clean: assign y to x, but mark x as not tainted

Now we would like to use dataflow analysis on this low-level code to discover tainted variables. A variable that might contain a tainted value is marked as tainted.

To discover tainted variables we define the following sets for each basic block b:

- IN[b] = set of all possibly tainted variables on entry to block b
- OUT[b] = set of all possibly tainted variable on exit from block b
- GEN[b] = set of all variables marked tainted in block b and not cleaned before exit
- CLEAN[b]= set of all variables cleaned in block b and not later tainted before exit

The sets GEN[b] and CLEAN[b] can be computed once based on the static contents of each block b.

The IN[b] and OUT[b] sets need to be computed iteratively during the analysis.

Exercise 1

- Give appropriate dataflow equations for the IN and OUT sets for a block b in terms of the IN, OUT, GEN, and CLEAN sets.
- As usual, these equations will involve some combination of local information about the block itself as well as information about the block's predecessors and successors.

Exercise 2

- Using an iterative dataflow analysis identify the tainted variables in the IN and OUT sets for each block in the above graph.
- You should first fill in the GEN and CLEAN entries for each block, then iteratively solve for IN and OUT. Choose whichever direction (forward or backward) you wish to solve the equations.
- You should assume that there are no tainted variables in the IN set for block b1.