Homework 1
Posted Thursday January 16, Due Monday Thursday 30
50 points

**Problem 1 (6 pts).** (From Aho, Lam, Sethi, Ullman.) For the CFG below fill in *Reaching Definitions* gen and kill sets for each block, and in and out sets for each block. (The in and out sets must show the final solution, not an intermediate value.)

```
1. a = 1
2. b = 2
3. c = a+b
4. d = c-a
5. d = b+d
8. b = a+b
9. e = c-a
10. a = b*d
11. b = a-d
```
Problem 2 (7 pts). (From Aho, Lam, Sethi, Ullman.) For the same CFG fill in \textit{Available Expressions} $\text{gen}_{AE}$ and $\text{kill}_{AE}$ sets for each block, and $\text{in}_{AE}$ and $\text{out}_{AE}$ sets for each block.

\begin{itemize}
  \item 1. $a = 1$
  \item 2. $b = 2$
  \item 3. $c = a + b$
  \item 4. $d = c - a$
  \item 5. $d = b + d$
  \item 6. $d = a + b$
  \item 7. $e = e + 1$
  \item 8. $b = a + b$
  \item 9. $e = c - a$
  \item 10. $a = b \times d$
  \item 11. $b = a - d$
\end{itemize}

Problem 3 (7 pts). (From Aho, Lam, Sethi, Ullman.) Now fill in Live Variables $\text{gen}_{LV}$ and $\text{kill}_{LV}$ sets for each block, and $\text{in}_{LV}$ and $\text{out}_{LV}$ sets for each block.

$\text{gen}_{LV}(B_1) =$
$\text{kill}_{LV}(B_1) =$
$\text{in}_{LV}(B_1) =$
$\text{out}_{LV}(B_1) =$

1. $a = 1$
2. $b = 2$

$\text{gen}_{LV}(B_2) =$
$\text{kill}_{LV}(B_2) =$
$\text{in}_{AE}(B_2) =$
$\text{out}_{AE}(B_2) =$

3. $c = a + b$
4. $d = c - a$

$\text{gen}_{LV}(B_3) =$
$\text{kill}_{LV}(B_3) =$
$\text{in}_{LV}(B_3) =$
$\text{out}_{LV}(B_3) =$

5. $d = b + d$

$\text{gen}_{LV}(B_4) =$
$\text{kill}_{LV}(B_4) =$
$\text{in}_{LV}(B_4) =$
$\text{out}_{LV}(B_4) =$

6. $d = a + b$
7. $e = e + 1$

$\text{gen}_{LV}(B_5) =$
$\text{kill}_{LV}(B_5) =$
$\text{in}_{LV}(B_5) =$
$\text{out}_{LV}(B_5) =$

8. $b = a + b$
9. $e = c - a$

$\text{gen}_{LV}(B_6) =$
$\text{kill}_{LV}(B_6) =$
$\text{in}_{LV}(B_6) =$
$\text{out}_{LV}(B_6) =$

10. $a = b * d$
11. $b = a - d$
Problem 4 (10 pts, 2.5 pts each). (From Aho, Lam, Sethi, Ullman.) Let $V$ be the set of complex numbers. Which of the following operations can serve as the meet operation for a lattice over $V$? For each of the choices below, if your answer is NO, explain why not. If your answer is YES, leave it at that.

a) Addition: $(a + ib) \land (c + id) = (a + c) + i(b + d)$

b) Multiplication: $(a + ib) \land (c + id) = (ac - bd) + i(ad + bc)$

c) Component-wise minimum: $(a + ib) \land (c + id) = \min(a, c) + i \min(b, d)$

d) Component-wise maximum: $(a + ib) \land (c + id) = \max(a, c) + i \max(b, d)$

Problem 5 (10 pts). The intraprocedural Must-be-modified problem is a backward dataflow problem solvable by fixpoint iteration. A variable is in the must-be-modified set on exit of CFG node $n$, if it is modified on all paths from $n$ to exit. The problem statement is as follows: for each node $n$ compute the set of variables that are in the must-be-modified set on exit from $n$.

a) (5 pts) Define the analysis as an instance of the dataflow framework. Specify

Lattice $L$, $\leq$:

Merge operator:

Transfer functions:

b) (5 pts) Are the functions for this problem distributive or monotone? Show your proof.
Problem 6 (10 pts). (Modified from Nielson, Nielson and Hankin) A bit vector dataflow analysis is a special case of a monotone dataflow analysis where

I. The property space $L$ is the lattice of the subsets over some finite set $D$, and $\leq$ is either $\subseteq$ or $\supseteq$ and

II. The transfer function space is $F = \{f : \mathcal{P}(D) \to \mathcal{P}(D) \mid f(Y) = (Y \cap Y_1) \cup Y_2 \text{ where } Y_1 \subseteq D \text{ and } Y_2 \subseteq D \text{ are constants}\}$

Note: $\mathcal{P}(D)$ denotes the powerset of $D$ (the powerset is also frequently denoted by $2^D$). The above condition states that every transfer function $f$ can be written as $f(Y) = (Y \cap Y_1) \cup Y_2$ where $Y$ is the argument of the function (the in(j) set in a forward problem), and $Y_1$ and $Y_2$ are constants that do not depend on $Y$.

a) (5 pts) Briefly argue that the four classical dataflow analyses are bit vector dataflow analyses.

b) (5 pts) Devise a distributive analysis that is not a bit vector analysis. Hint: Consider one of the non-distributive examples we studied in class, and drop one of the statements from the syntax.