

CS243 HW 3 Solutions

Winter 2017

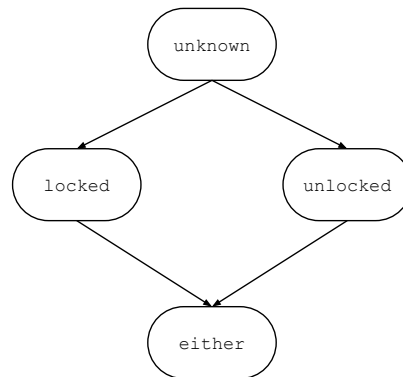
Note: these solutions are slightly more detailed than we would expect you to hand in so that you can learn from any questions you missed.

Problem 1: Lock Analysis – 10 points

A correct single analysis got full points. We accepted solutions different from the one below, but points were deducted for solutions with a separate analysis for each warning, if warnings were generated incorrectly, if there were errors in the specified analysis or for incorrect reasoning.

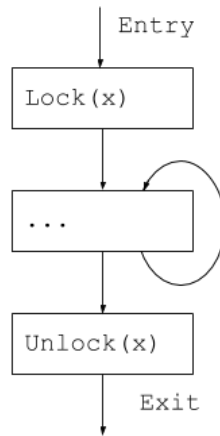
1. Analysis is as follows:

- Lattice elements: set of all variables, each in one of *unknown*, *locked*, *unlocked* or *either* state (product lattice)



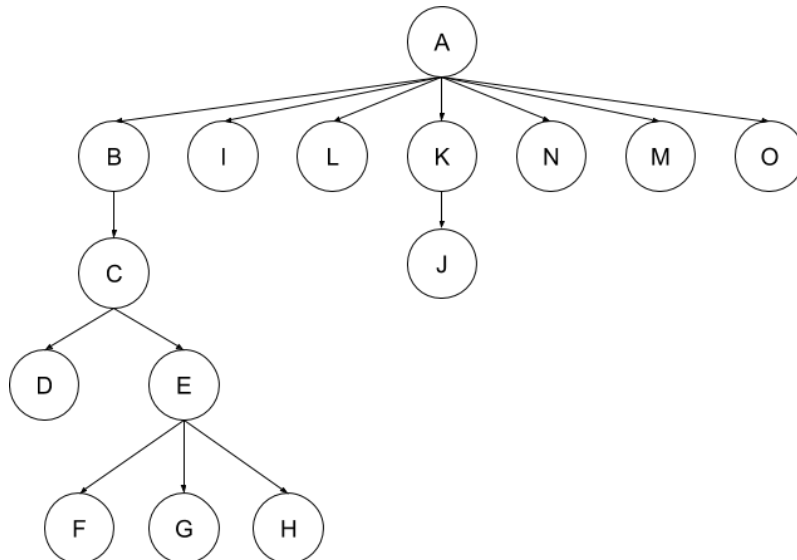
- Meet is a point-wise meet for each variable (as described by the lattice).
 - Direction: forward
 - Initialization: $\text{OUT}[B] = \text{unknown}$ for all variables
 - Boundary condition: $\text{OUT}[\text{Entry}] = \text{unlocked}$ for all variables
 - Transfer function f :
 - Lock(v): v becomes *locked* (constant), all others unchanged (identity),
 - Unlock(v): v becomes *unlocked* (constant), all others unchanged (identity)
2. It will converge, since it is a monotone framework, with a product lattice with finite descending chains.
3. It is monotone, since the transfer function defined above gives identity or constant, each of which is monotone.
4. It is distributive, since $f(v_1 \wedge v_2) = f(v_1) \wedge f(v_2)$, for the transfer function f defined above.
5. Warnings are issued as follows:
- Warning I: if $\text{IN}[\text{Lock}(v)]$ has v in *locked* or *either* state
 - Warning II: if $\text{IN}[\text{Exit}]$ has any v in *locked* or *either* state
 - Warning III: if $\text{IN}[\text{Unlock}(v)]$ has v in *unlocked* or *either*

Note, in the above lattice, top element, *unknown* is required for correct initialization. Initializing $\text{OUT}[B]$ to *unlocked* incorrectly generates a warning for the example shown in the figure.



Problem 2: Dominators and Natural Loops – 6 points

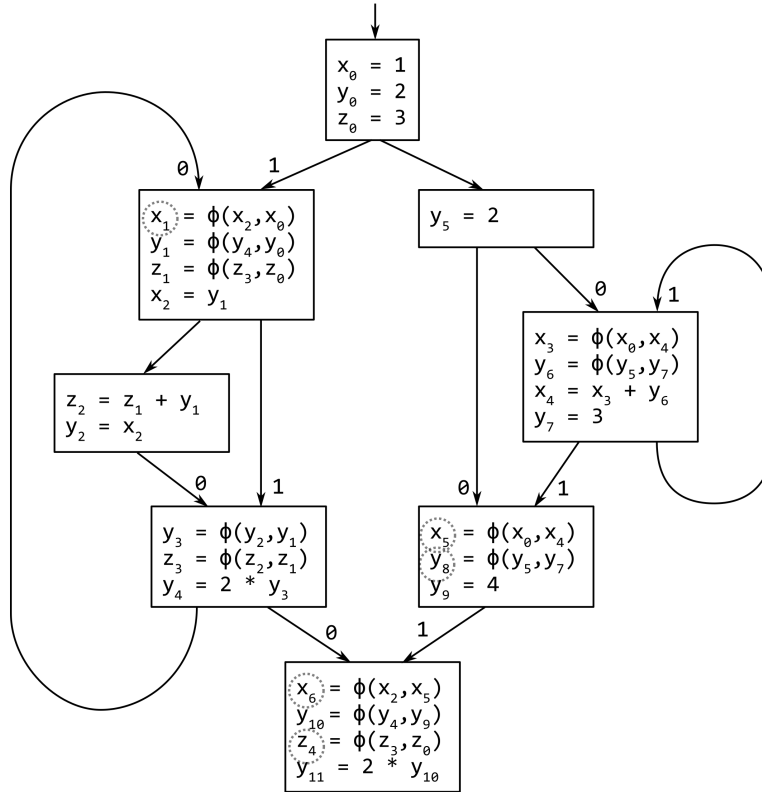
1. The dominator tree for the graph is given below.



2. The back edges and their natural loops are given below:

- $J \rightarrow K$ with natural loop $\{J, K\}$
- $D \rightarrow B$ with natural loop $\{B, C, D\}$
- $H \rightarrow E$ with natural loop $\{E, F, G, H\}$
- $F \rightarrow E$ with natural loop $\{E, F\}$
- $H \rightarrow B$ with natural loop $\{B, C, E, F, G, H\}$

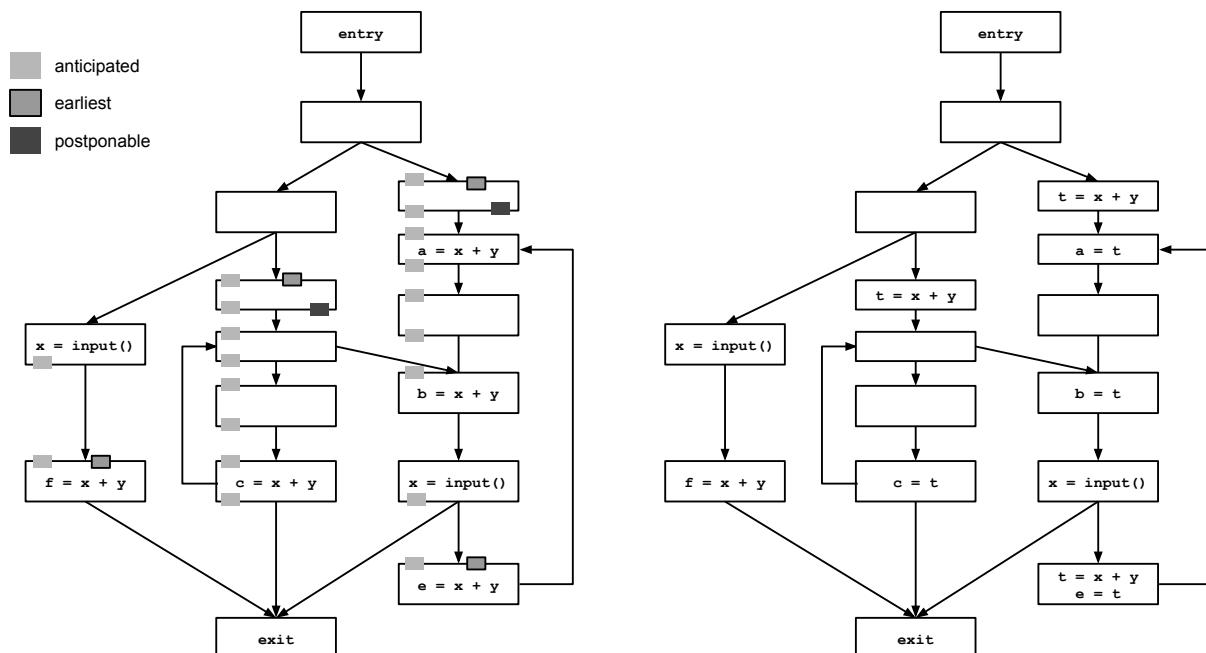
Problem 3: Static Single Assignment Form (SSA) – 10 points The transformed graph is given below. The phi nodes $x_1, x_5, y_8, x_6,$ and z_4 are all redundant (and circled below).



Problem 4: Partial Redundancy Elimination – 10 points

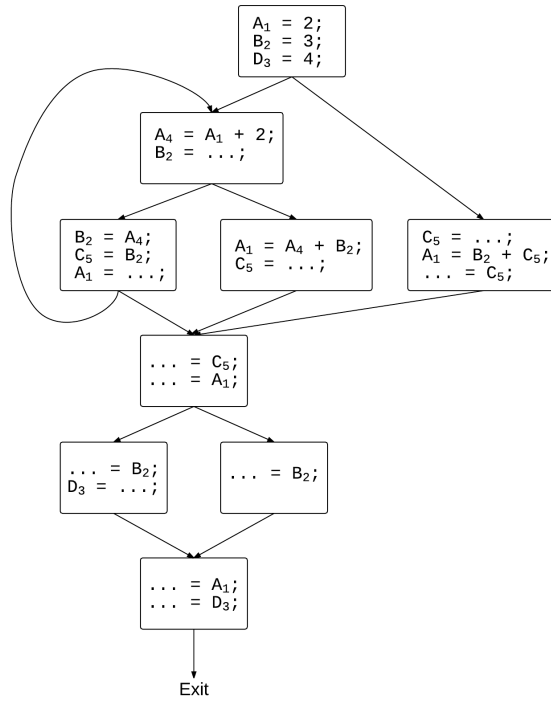
Points were deducted for not placing expressions correctly and for not cleaning up temporaries.

The left figure shows points where the expression $x + y$ is anticipated, earliest and postponable. The right figure places $x + y$ at the points where the expression must be computed latest.



Problem 5: Register Allocation – 10 points

In the control flow graph below, we have labeled each definition and use of a variable with a subscript indicating which live range it corresponds to. Note, the live range assignments given are after we have merged overlapping live ranges.



We now draw the interference graph and the assignment of registers to live ranges.

