

Static Single Assignment Form

Last Time

- Introduction to SSA
- Inserting ϕ functions

Today

- Csmith paper
- Renaming variables for SSA form
- SSA and DFA

Next Time

- Reuse optimizations

February 16, 2015

SSA and DFA

1

LLVM Questions

Are you confused by LLVM?

February 16, 2015

SSA and DFA

2

Transformation to SSA Form

Two steps

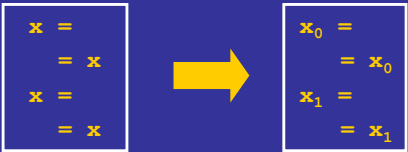
- Insert ϕ -functions
- Rename variables

Variable Renaming

Basic idea

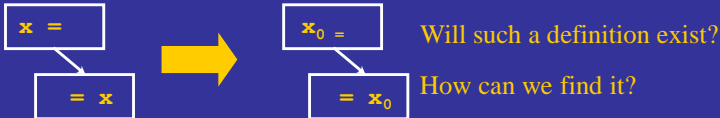
- When we see a variable on the LHS, create a new name for it
- When we see a variable on the RHS, use appropriate subscript

Easy for straightline code



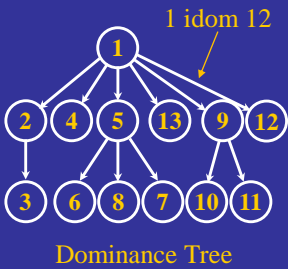
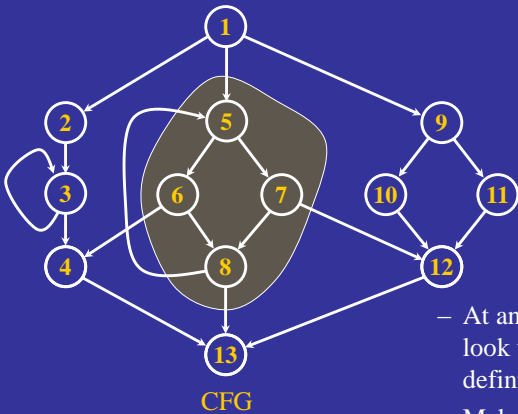
Harder when there's control flow

- For each use of x , find the definition of x that dominates it



Dominance Tree Example

The dominance tree shows the dominance relation



- At any point in the dominance tree, look up the tree, searching for the first definition that you find
- Make a depth-first traversal of the Dominance Tree to assign version numbers

February 16, 2015

SSA and DFA

5

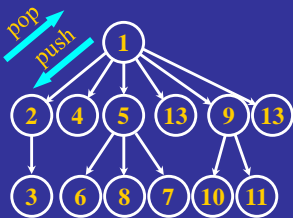
Variable Renaming (cont)

Data Structures

- $\text{Stacks}[v] \forall v$
Holds the subscript of most recent definition of variable v , initially empty
- $\text{Counters}[v] \forall v$
Holds the current number of assignments to variable v ; initially 0

Auxiliary Routine

procedure GenName(variable v)
 $i := \text{Counters}[v]$
 push i onto $\text{Stacks}[v]$
 $\text{Counters}[v] := i + 1$



Use the Dominance Tree to remember the most recent definition of each variable

February 16, 2015


SSA and DFA

6

Variable Renaming Algorithm

```
procedure Rename(block b)
  if b previously visited return
  for each  $\phi$ -function p in b
    GenName(LHS(p)) and replace v with  $v_i$ , where  $i = \text{Top}(\text{Stack}[v])$ 
  for each statement s in b (in order)
    for each variable  $v \in \text{RHS}(s)$ 
      replace v by  $v_i$ , where  $i = \text{Top}(\text{Stack}[v])$ 
    for each variable  $v \in \text{LHS}(s)$ 
      GenName(v) and replace v with  $v_i$ , where  $i = \text{Top}(\text{Stack}[v])$ 
  for each  $s \in \text{succ}(b)$  (in CFG)
     $j \leftarrow$  position in s's  $\phi$ -function corresponding to block b
    for each  $\phi$ -function p in s
      replace the  $j^{\text{th}}$  operand of  $\text{RHS}(p)$  by  $v_i$ , where  $i = \text{Top}(\text{Stack}[v])$ 
  for each  $s \in \text{child}(b)$  (in DT)
    Rename(s)
  for each  $\phi$ -function or statement t in b
    for each  $v_i \in \text{LHS}(t)$ 
      Pop(Stack[v])
```

Call Rename(entry-node)



Recurse using Depth First Search

Unwind stack when done with this node

February 16, 2015

SSA and DFA

7