CS 6340 – Spring 2013 – Assignment 3
Assigned: January 23, 2013
Due: January 30, 2013

Name Sam Britt, Shriram Swaminathan,
Name and Sivaramachandran Ganesan

At the beginning of class on the due date, submit your neatly presented solution with this page stapled to the front (40 points).

NOTE: All work on this problem set is to be done with your partner and without solutions from other past or current students. Any violations will be dealt with according to the Georgia Tech Academic Honor Code and according to the College of Computing process for resolving academic honor code violations. All work must be done using some document creation tool. In addition, graphs must be drawn with a graph-drawing tool—no hand-drawn graphs will be accepted. We'll discuss this requirement more in class.

Given the following program and the statement-based control-flow graph for that program (which you created in Problem Set 1):

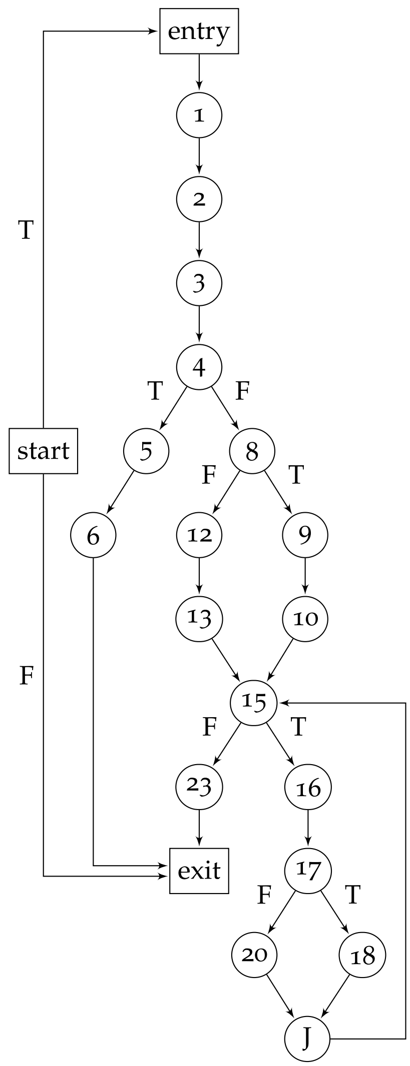
```
procedure sqrt(real x):real
  real x1,x2,x3,eps,errval;

  begin
    x3 = 1
    errval = 0.0
    eps = .001
    if (x <= 0.0)
       output("illegal operand");
       return errval;
    else
       if (x < 1)
          x1 = x;
          x2 = 1;
       else
          x1 = eps;
          x2 = x;
       endif
       while ( (x2-x1) >= 2.0*eps )
          x3 = (x1+x2)/2.0
          if ( (x3*x3-x)*(x1*x1-x) < 0)
             x2 = x3;
          else
             x1 = x3;
          endif;
       endwhile;
       return x3;
    endif;
  end.
```

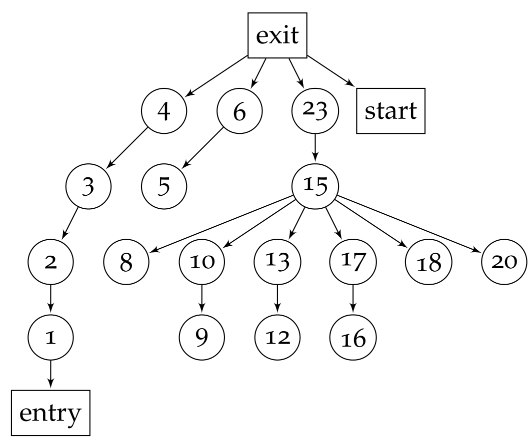
Sam Britt
Shriram Swaminathan
Sivaramachandran Ganesan

CS 6340
Assignment 4
Jan. 30, 2013

1. We start with the control dependence graph, augmented with a "start" node (below, left). From this, we create the postdominator tree.



Augmented CFG.



Corresponding postdominator tree.

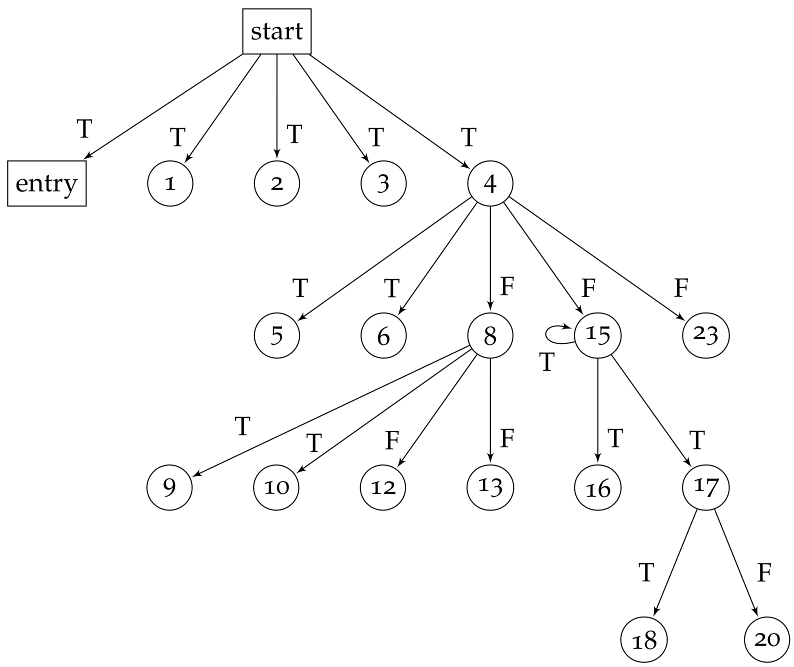
We then find the set S, where each element in S is an edge $(A, B) \in G$, where G is the above CFG and B does not postdominate A. We find that

$$S = \{(\text{start, entry}), (4, 5), (4, 8), (8, 9), (8, 12), (15, 16), (17, 18), (17, 20)\}.$$

For each edge $(A, B) \in S$, we find L to be the common ancestor of A and B. Finally, the nodes that are control-dependent on A are those the path from L to B on the postdominator tree, including B, and including L only if $L = A$. These results are summarized in the following table:

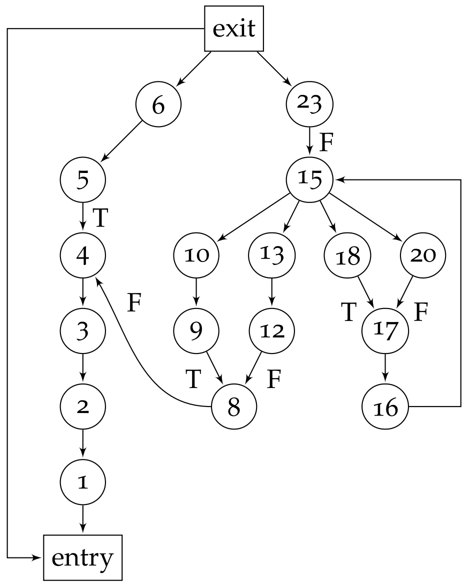
$(A, B) \in S$	Condition (T/F)	L	Nodes dependent on A
(start, entry)	T	exit	{entry, 1, 2, 3, 4}
(4, 5)	T	exit	{5, 6}
(4, 8)	F	exit	{8, 15, 23}
(8, 9)	T	15	{9, 10}
(8, 12)	F	15	{12, 13}
(15, 16)	T	15	{15, 16, 17}
(17, 18)	T	15	{18}
(17, 20)	F	15	{20}

The control dependence graph (below) is constructed directly from the above table.

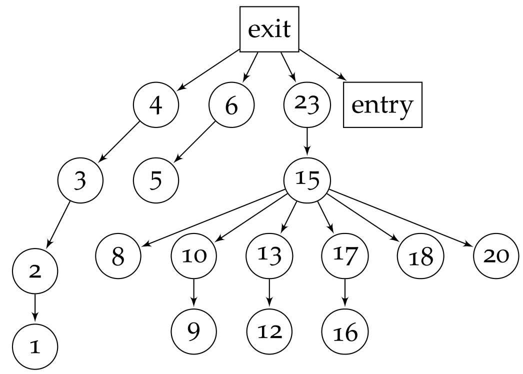


Control dependence graph.

2. We start by augmenting the CFG with an edge from entry to start, and then reverse it (below, left). From that we create the dominator tree (below, right).



Reverse CFG.



Corresponding dominator tree.

For each node n, we traverse down the reverse CFG (RCFG) until we find a node that n does not dominate; that is, a node that is a child of n in the RCFG but not in the dominator tree. This set of nodes is the dominance frontier of n. These are tabulated below.†

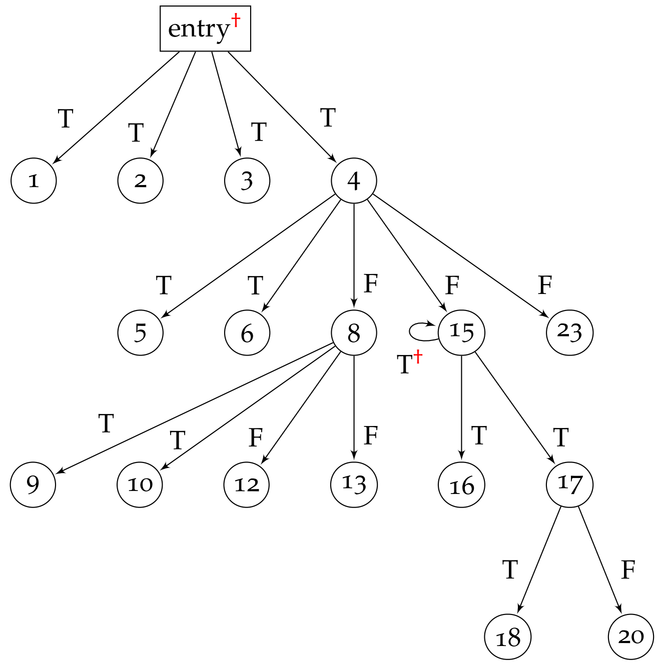
†mention condition if including

†does "condition" even make sense?

Block	Dominance frontier	Condition†
entry	∅	T
1	{entry}	T
2	{entry}	T
3	{entry}	T
4	{entry}	T
5	{4}	T
6	{4}	T
8	{4}	F
9	{8}	T
10	{8}	T
12	{8}	F
13	{8}	F
15†	{4}	F
16	{15}	T
17	{15}	T
18	{17}	T
20	{17}	F
23	{4}	F

†does 15 dominate 15?

By inverting the dominance frontier sets, we arrive at the control dependence graph below.



†note the difference from above

†include?

Control dependence graph.