CSc 553

Principles of Compilation

31 : Dominators and Natural Loops

Department of Computer Science
University of Arizona

collberg@gmail.com

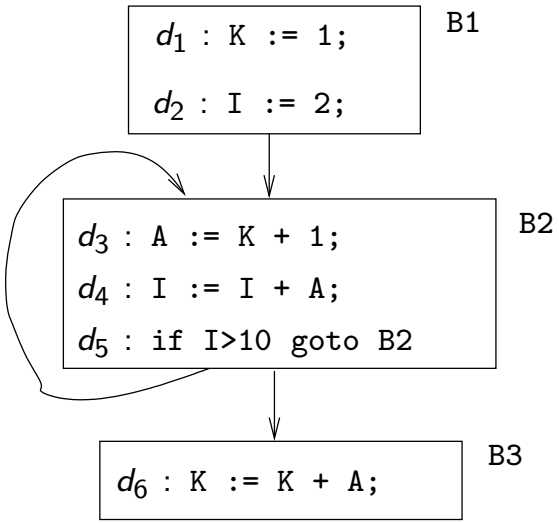
Copyright © 2011 Christian Collberg

Introduction

Loop Invariants

- Let C be a computation in a loop body. C is **invariant** if it computes the same value during all iterations. C can sometimes be moved out of the loop.

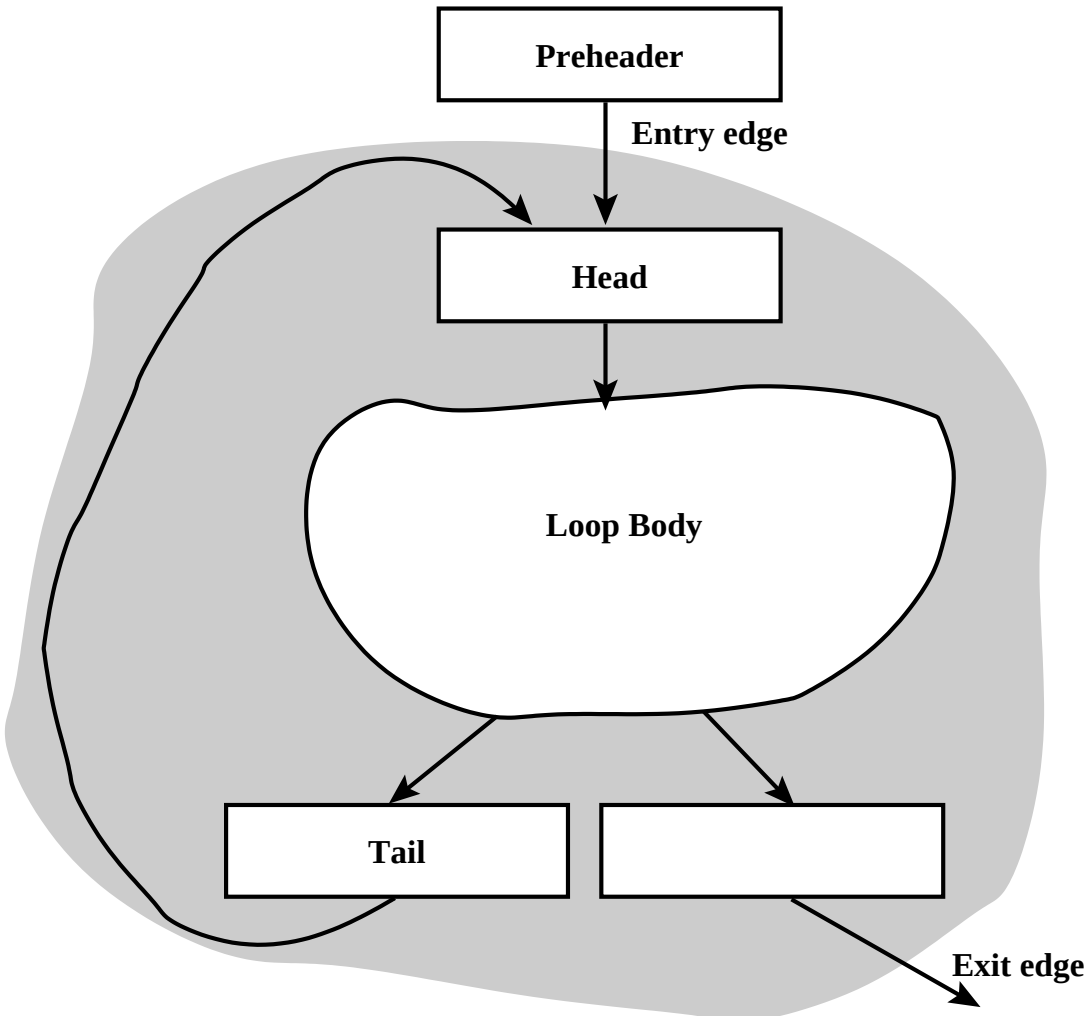
```
K := 1; I := 2;
REPEAT
   A := K + 1; I := I + A;
UNTIL I <= 10;
K := K + A;
```



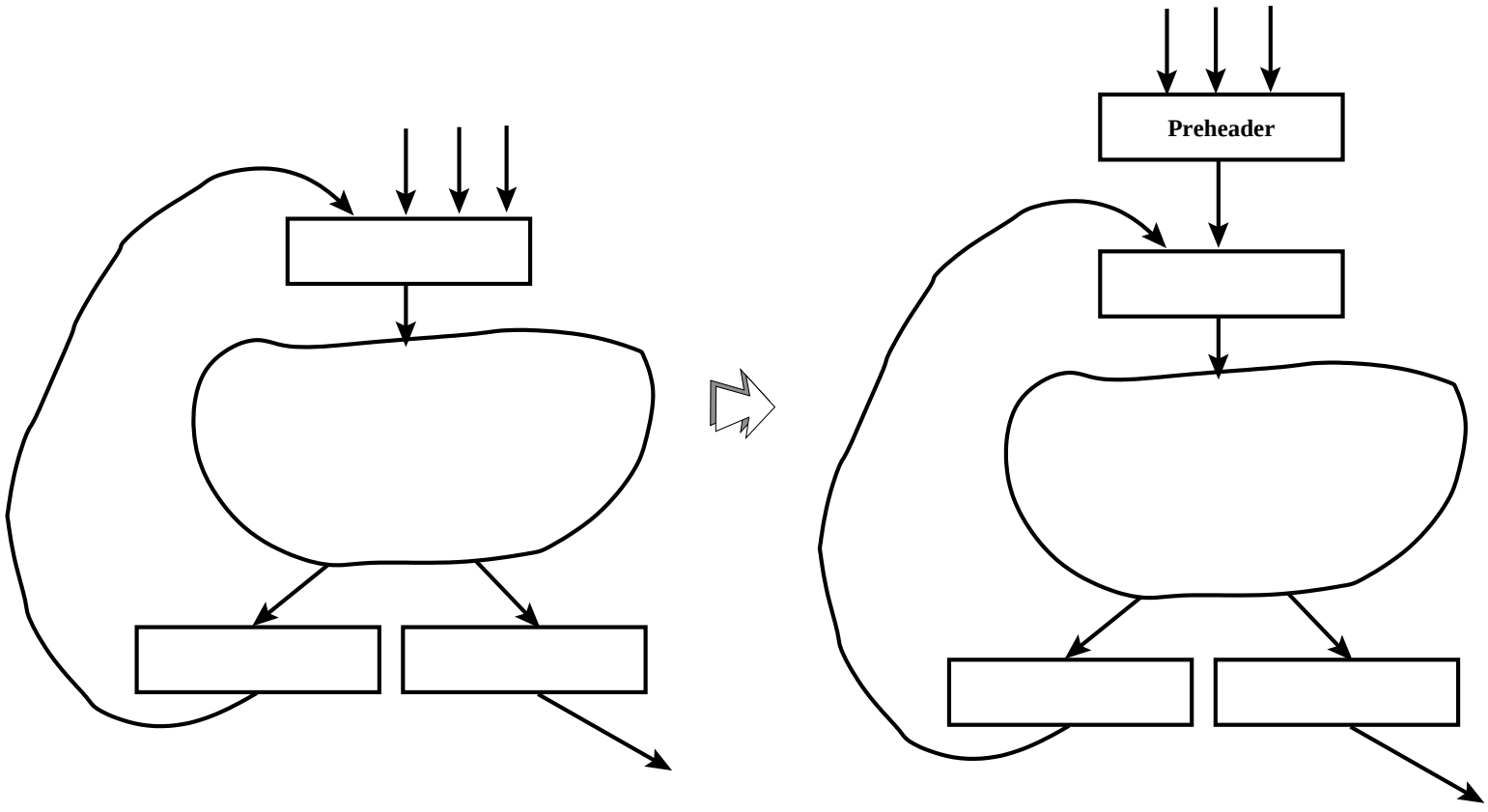
- How do we know what is a loop???

Loops

Loop Terminology



Preheaders



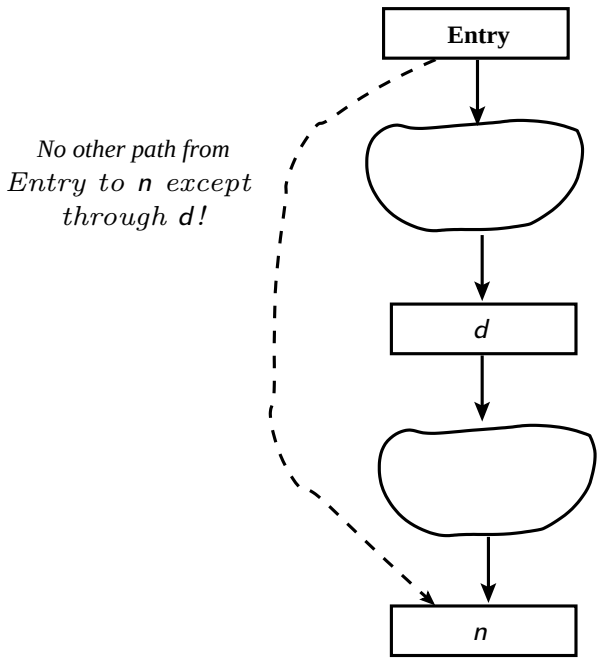
- A preheader is useful, for example if we want to move out loop-invariant computations.
- Not all loops have preheaders — but we can always add one.

Dominators

Dominators

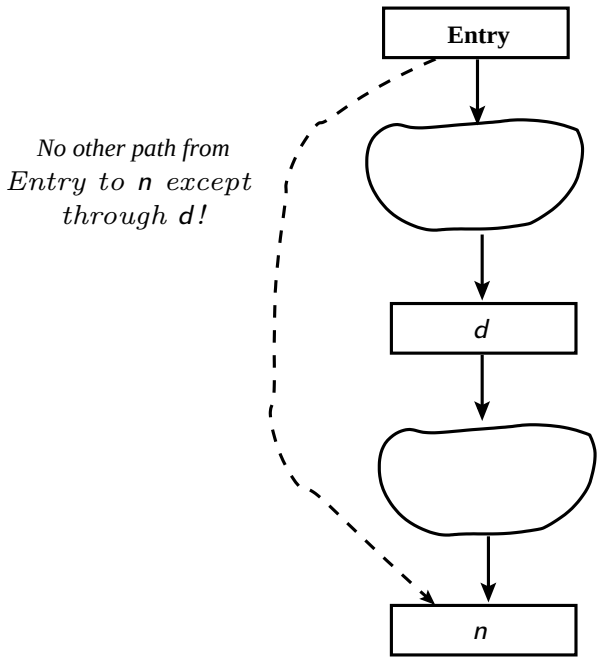
- To detect what the loops are in a program we first have to perform a *dominator analysis.*
- Definition:

  A node *d* dominates a node *n* if every path from the entry node to *n* must go through *d*.



Dominators

- Notation: d dom n — d strictly dominates n.
- Intuition: Given a node n, which blocks are guaranteed to have executed prior to executing n.



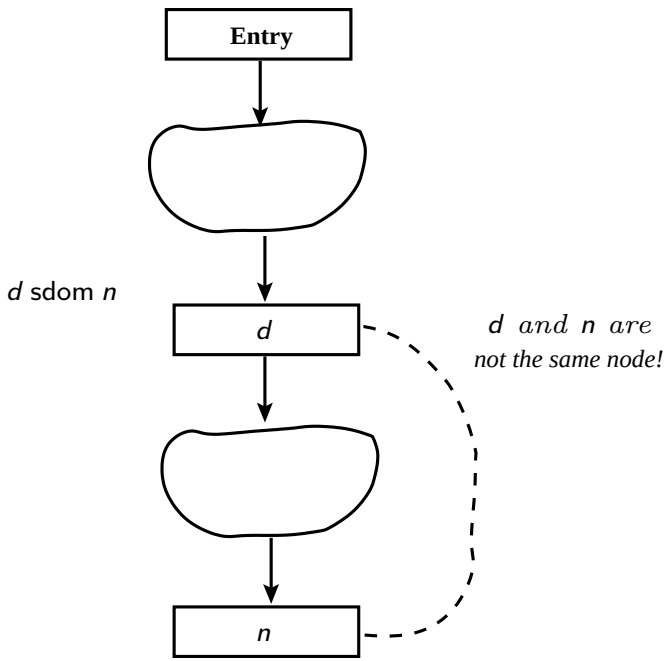
- Every node dominates itself: d dom d.

Strict Dominator

- Definition:

  A node d *strictly dominates* a node n if d dominates n and $d \neq n$.

- Notation: d sdom n — d strictly dominates n.

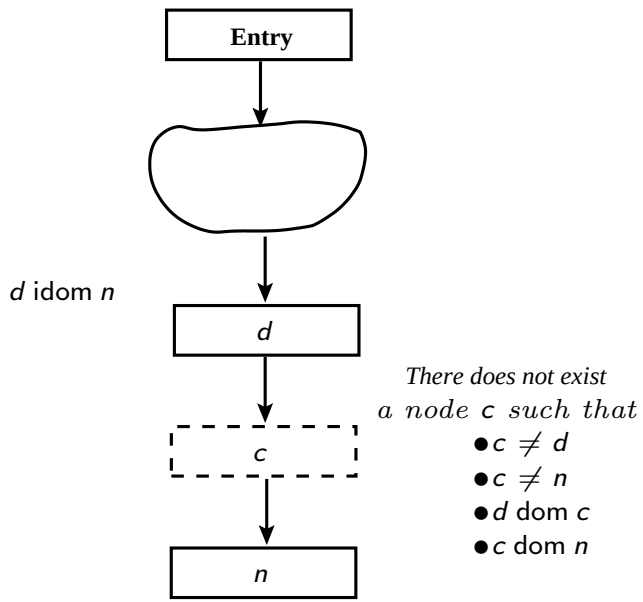


Immediate Dominator

- Definition:

  The immediate dominator d of a node n is the unique node that strictly dominates n but does not strictly dominate any other node that strictly dominates n.

- Entry nodes don't have an immediate dominator.
- Notation: d idom n — d immediately dominates n.



Post dominator

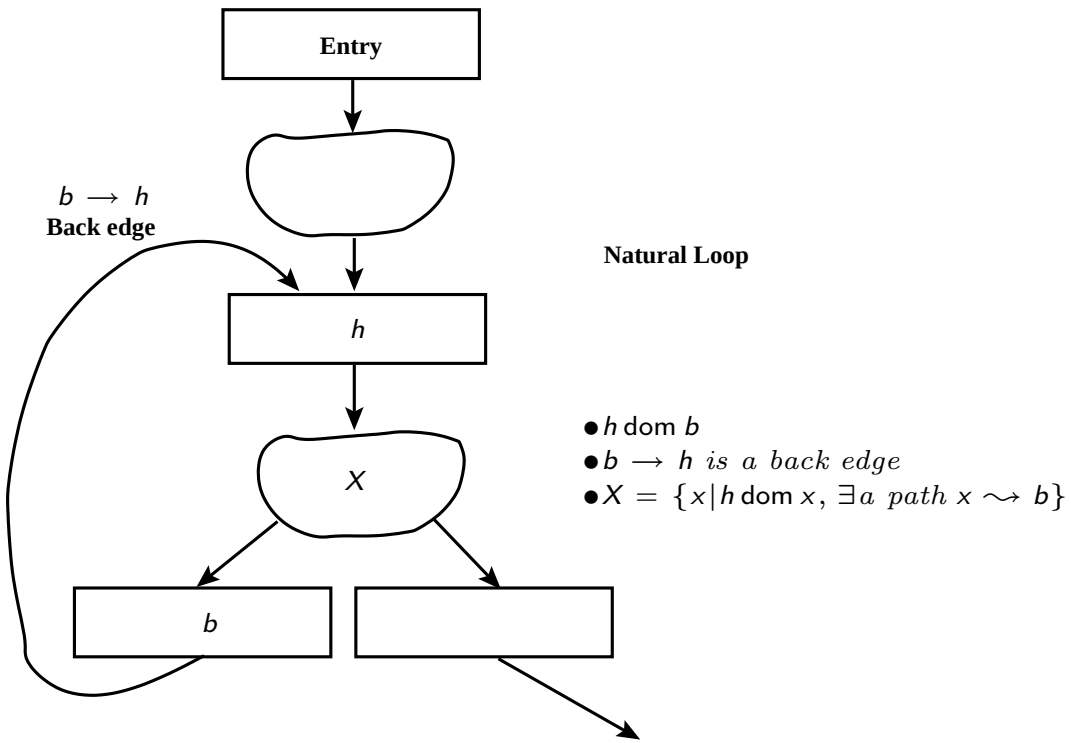
A node d post dominates a node n if every path from n to the exit node must go through d.

- Notation: d pdom n — d post dominates n.
- Intuition: Given a node n, which blocks are guaranteed to execute *after* executing n.

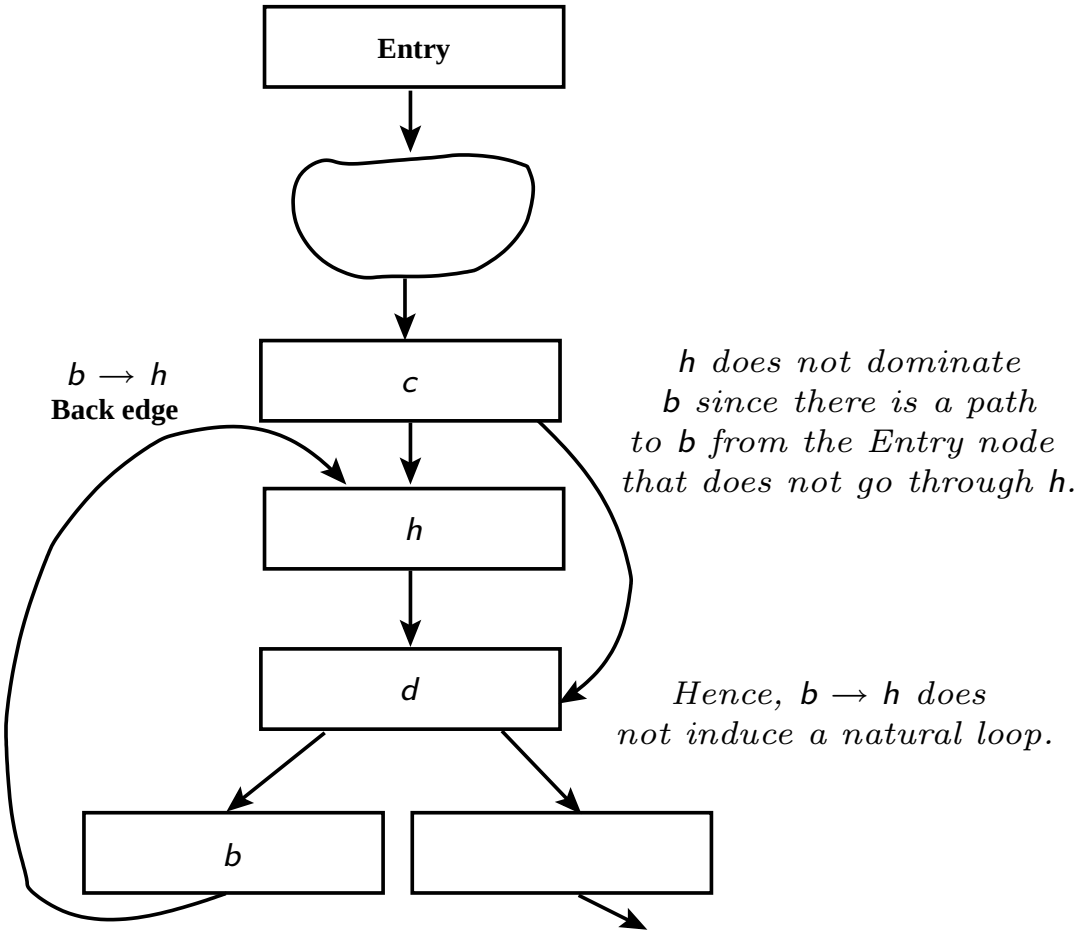
Natural Loop

- Definition:

  A back edge $b \longrightarrow h$, where h dom b, induces a *natural loop* consisting of all nodes x, where h dom x and there there is a path from x to b not containing b.



Example — Not a Natural Loop



Computing Dominators

Dataflow Equations

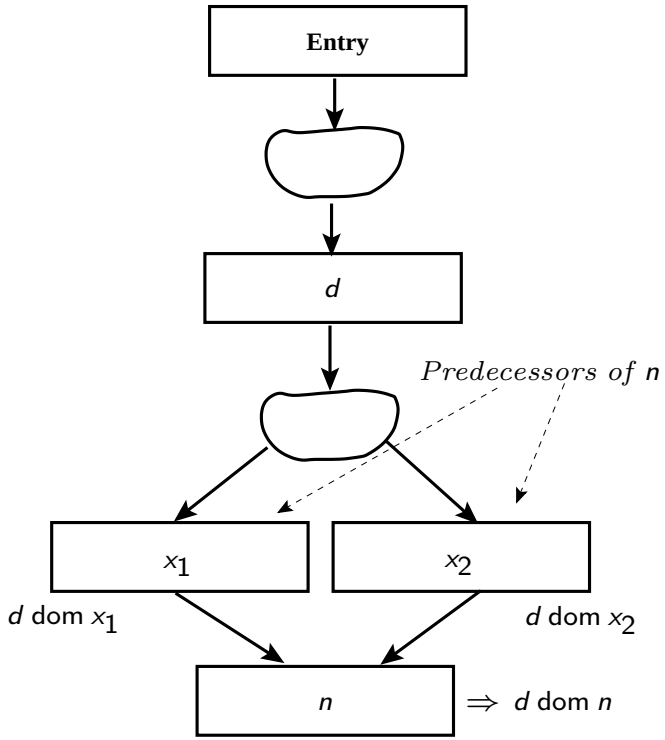
- The dominators of a node n are given by

$$\begin{aligned} \mathtt{dom}(\text{entry node}) &= \{\text{entry node}\} \\ \mathtt{dom}(n) &= \{n\} \cup \Big(\bigcap_{\mathtt{preds}\ p\ \mathtt{of}\ n} \mathtt{dom}(p) \Big) \end{aligned}$$

- The dominator of the entry node is the entry node itself.
- The set of dominators for a node n is the intersection of the set of dominators for all predecessors of n.
- n is also in the set of dominators for n.

Dataflow Equations — Intuition

$$\texttt{dom}(n) \;=\; \{n\} \cup \Big(\bigcap_{\texttt{preds } p \texttt{ of } n} \texttt{dom}(p) \Big)$$



- If d dominates all predecessors of n, then it also dominates n

Algorithm

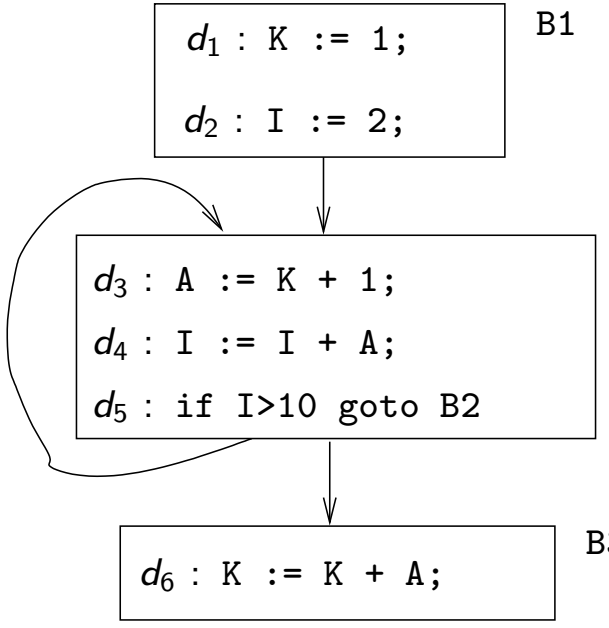
- N is the set of all nodes.
- n_0 is the entry node.

$$
\begin{array}{l}
\texttt{dom}(n_0) \texttt{ := } \{n_0\}\texttt{;} \\
\texttt{FOR EACH } n \in N - \{n_0\} \texttt{ DO} \\
\quad \texttt{dom}(n) \texttt{ := } N\texttt{;} \\
\texttt{WHILE CHANGES IN ANY dom}(n) \texttt{ DO} \\
\quad \texttt{FOR EACH } n \in N - \{n_0\} \texttt{ DO} \\
\qquad \texttt{dom}(n) \texttt{ := } \{n\} \cup (\bigcap_{\texttt{preds } p \texttt{ of } n} \texttt{dom}(p))
\end{array}
$$

Example 1

Example 1 — Initialization

```
K:=1;
I:=2;
REPEAT
   A:=K+1;
   I:=I+A;
UNTIL I<=10;
K:=K+A;
```

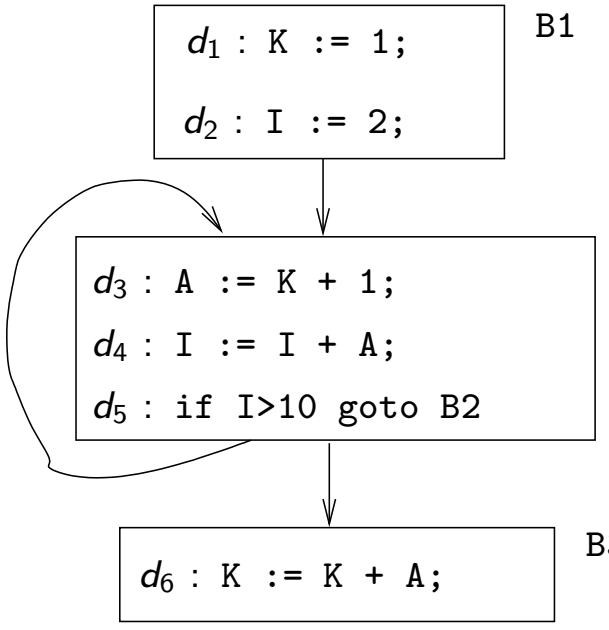


dom=$\{B_1\}$

dom=$\{B_1, B_2, B_3\}$

dom=$\{B_1, B_2, B_3\}$

Example 1 — First Iteration



$$\text{dom}=\{B_1\}$$

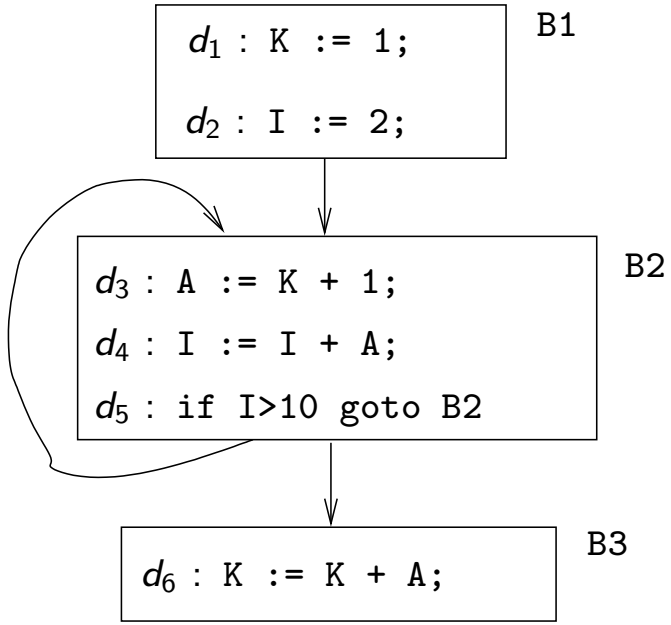
$$\begin{aligned}\text{dom}&=\{B_2\}\cup(\{B_1\}\cap\{B_1,B_2,B_3\})\\&=\{B_1,B_2\}\end{aligned}$$

$$\begin{aligned}\text{dom}&=\{B_3\}\cup(\{B_1,B_2\})\\&=\{B_1,B_2,B_3\}\end{aligned}$$

Example 1 — Final Result

- A back edge $b \longrightarrow h$, where h dom b, induces a *natural loop* consisting of all nodes x, where h dom x and there there is a path from x to b not containing b.

B_2 dom B_2



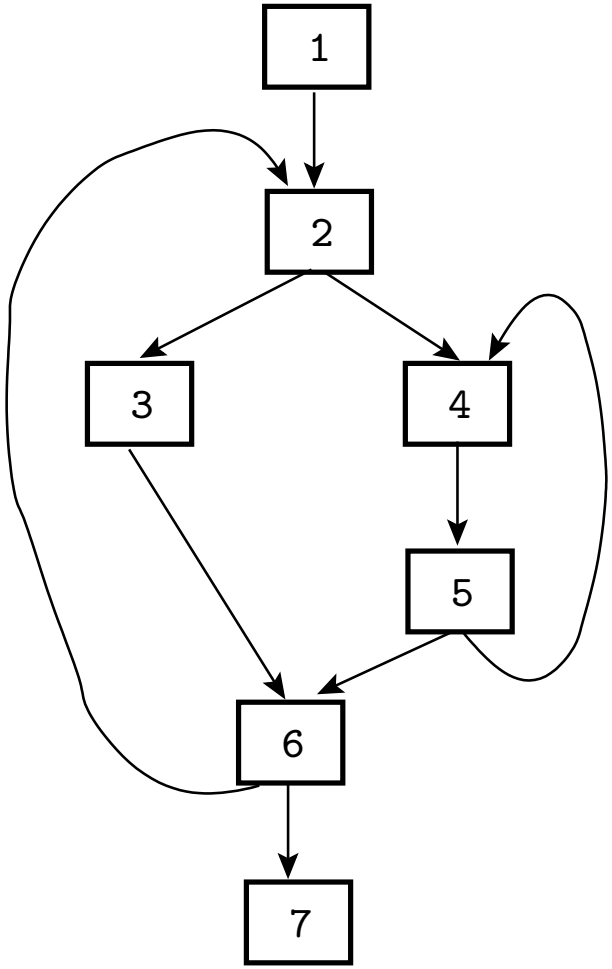
dom=$\{B_1\}$

dom=$\{B_1, B_2\}$

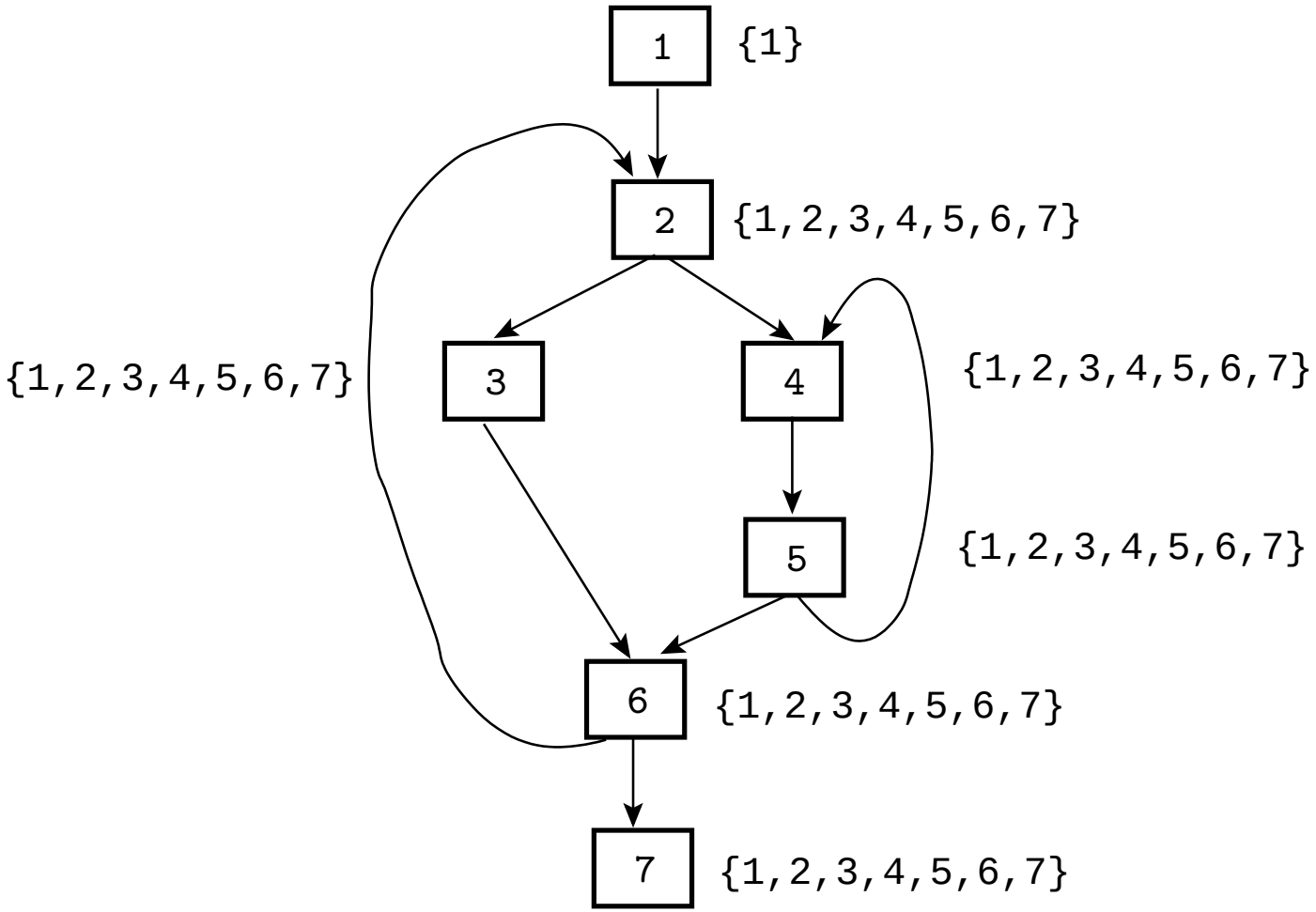
dom=$\{B_1, B_2, B_3\}$

Example 2

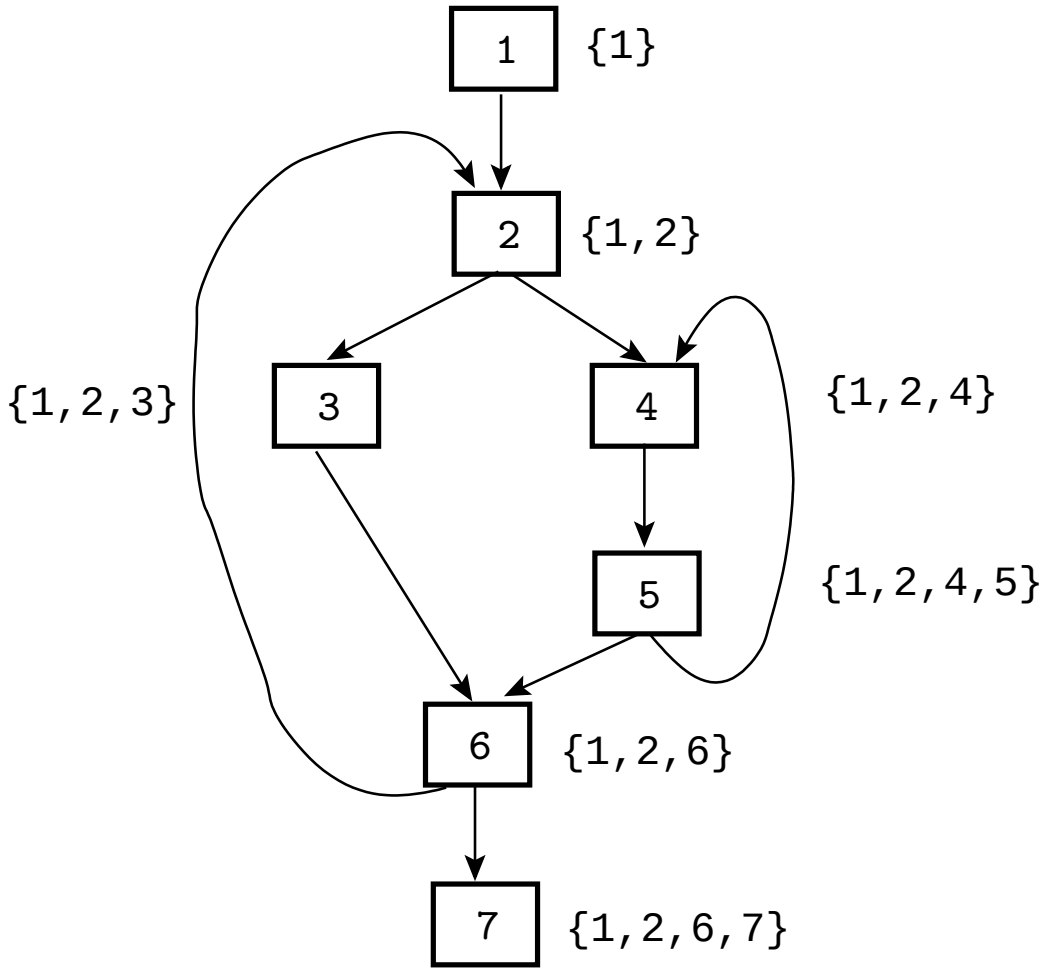
Example 2



Example 2 — Initialization

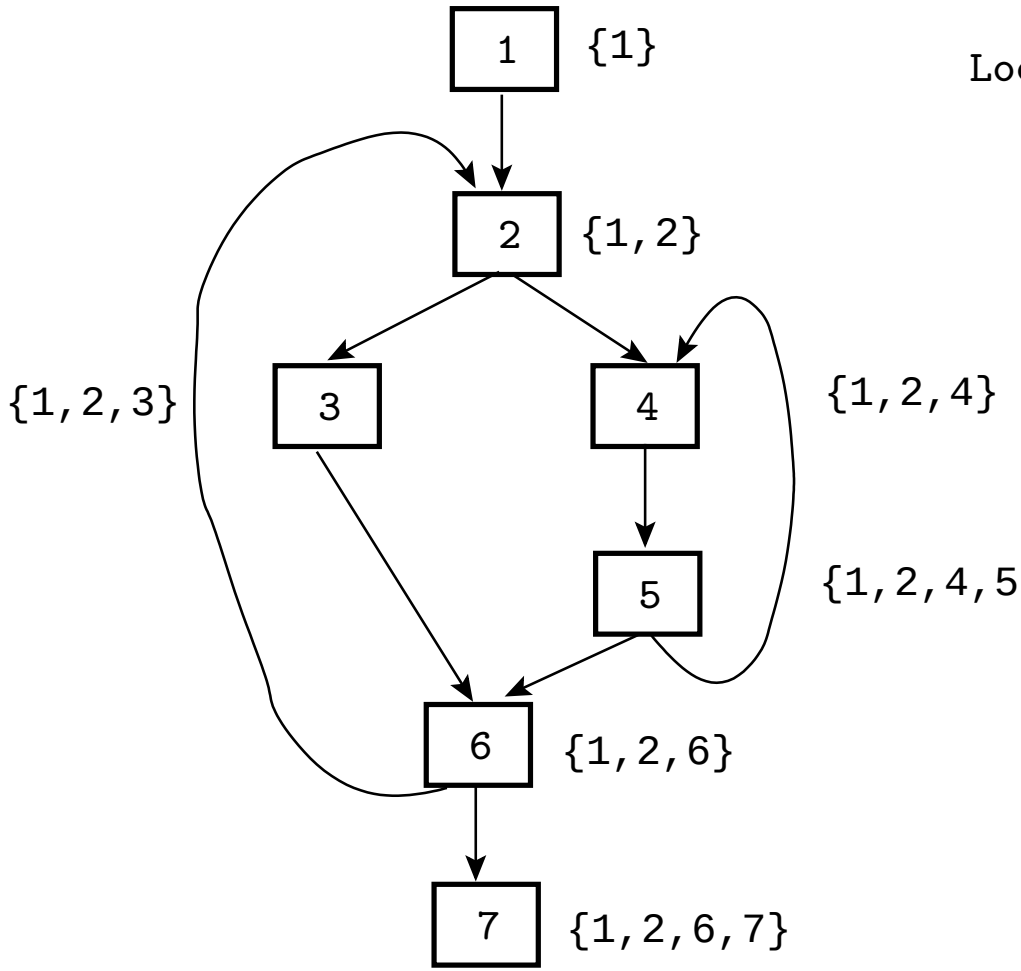


Example 2 — First iteration



Example 2 — Identifying loops

Back edge $b \to h$, $h \,\mathrm{dom}\, b$, induces a loop with all nodes x, where $h \,\mathrm{dom}\, x$ and there there is a path $x \rightsquigarrow b$ not containing b.



Loop 1:

Back edge: $6 \to 2$
$2 \,\mathrm{dom}\, 6$
Contains: {2,3,4,5,6}

Loop 2:
Back edge: $5 \to 4$
$4 \,\mathrm{dom}\, 5$
Contains: {4,5}

Summary

Summary

- Each node dominates itself.
- If x dominates y, and y dominates z, then x dominates z.
- If x dominates z and y dominates z, then either x dominates y or y dominates x.