

Wrap-up!

CSE 401 Section 10
by Anand, Gavin, Yukai

Announcements

- HW 4 due today, 6/3/21
 - However everyone has 2 free late days! (Absolute last day is Saturday, 6/5/21)
- CSE M 501 final project due Saturday, 6/5/21, report due 1 day later
 - No late days on these
- Email course staff if you need any additional resources or support!

Memorial Day	31	23:00 Project: CSE 401 project reports due No late submissions accepted.	01	14:30-15:20 Lecture zoom link Back end overview; instruction selection & scheduling slides	02	Section hw4 last-minute questions; wrapup	03	14:30-15:20 Lecture zoom link Register allocation (no new slides); Wrapup	04
						23:00 hw4 due (dataflow & SSA) (SSA diagram (optional))		23:00 CSE M 501 project due Saturday 11 pm No late submissions accepted	
								23:00 CSE M 501 report due Sunday 11 pm No late submissions accepted	



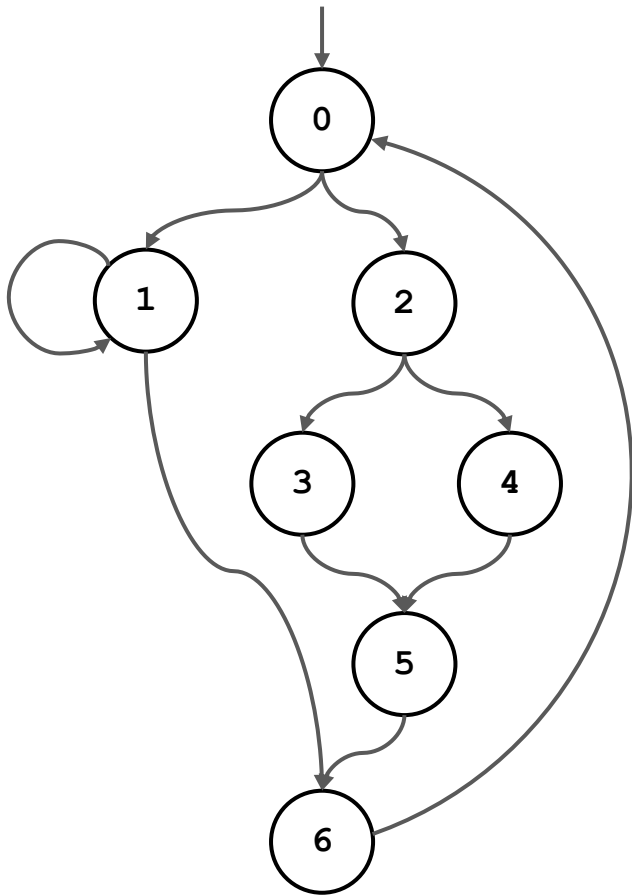
THANK YOU

For an amazing quarter! We hope you learned a lot and enjoyed being your TAs along the way 😊

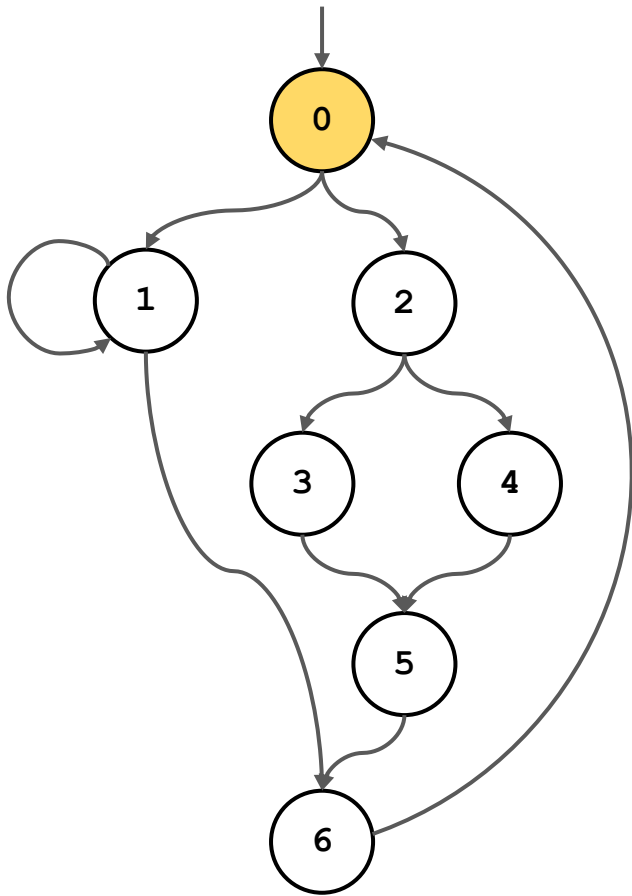
Agenda for Section

- TA chat/ goodbyes
- SSA Practice Problems
- HW Questions/General Office Hours
- Up to you!

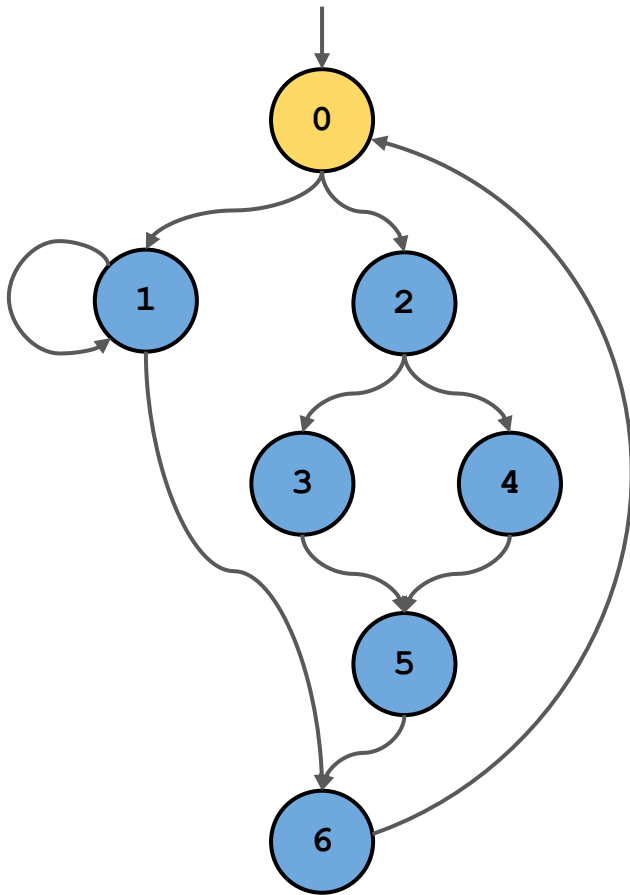
Problem 2(a)



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0		
1		
2		
3		
4		
5		
6		



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0		
1		
2		
3		
4		
5		
6		

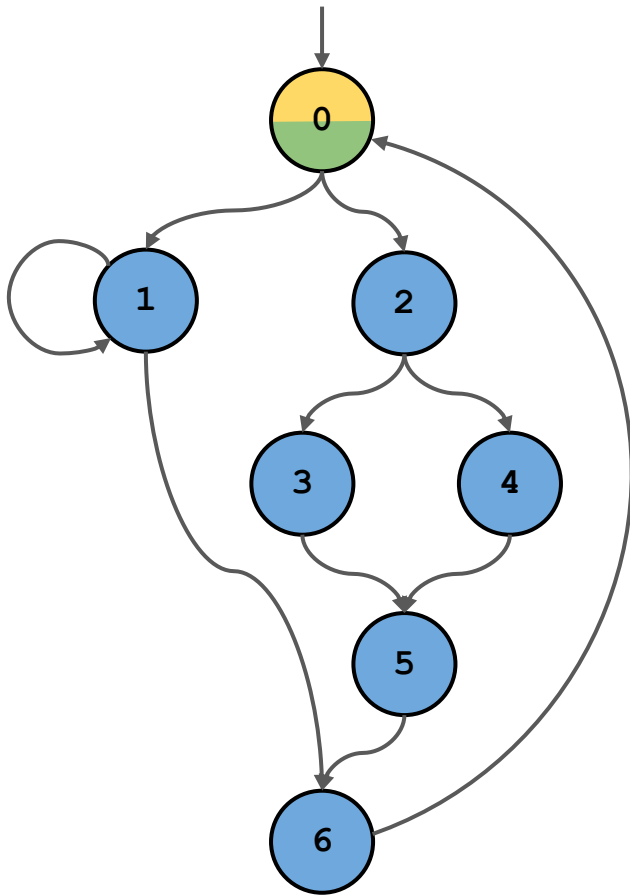


NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	
1		
2		
3		
4		
5		
6		

A node \mathbf{x} *dominates* a node \mathbf{y} iff every path from the entry point of the control flow graph to \mathbf{y} includes \mathbf{x} .

A node \mathbf{x} *strictly dominates* a node \mathbf{y} iff \mathbf{x} dominates \mathbf{y} and $\mathbf{x} \neq \mathbf{y}$

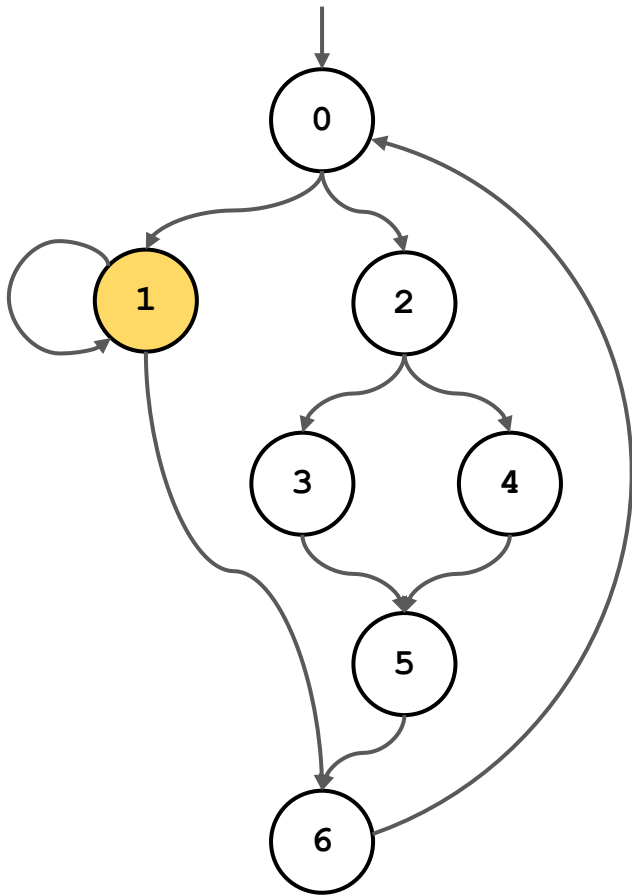
Need to go through 0 to get through 1, 2, 3, 4, 5, 6 and 0 cannot strictly dominate itself



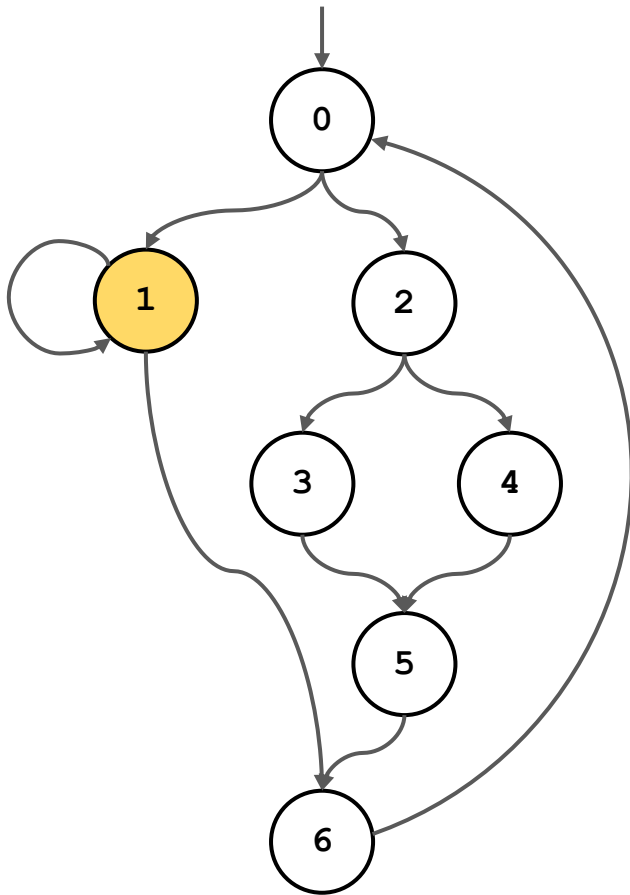
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1		
2		
3		
4		
5		
6		

A node \mathbf{Y} is in the *dominance frontier* of node \mathbf{X} iff \mathbf{X} dominates an immediate predecessor of \mathbf{Y} but \mathbf{X} does not strictly dominate \mathbf{Y} .
 A node $\mathbf{0}$ is in the *dominance frontier* of node $\mathbf{0}$ iff $\mathbf{0}$ dominates an immediate predecessor ($\mathbf{6}$) of $\mathbf{0}$ but $\mathbf{0}$ does not strictly dominate $\mathbf{0}$

0 dominates 6, 6 is an immediate predecessor of 0, 0 does not strictly dominate 0



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1		
2		
3		
4		
5		
6		

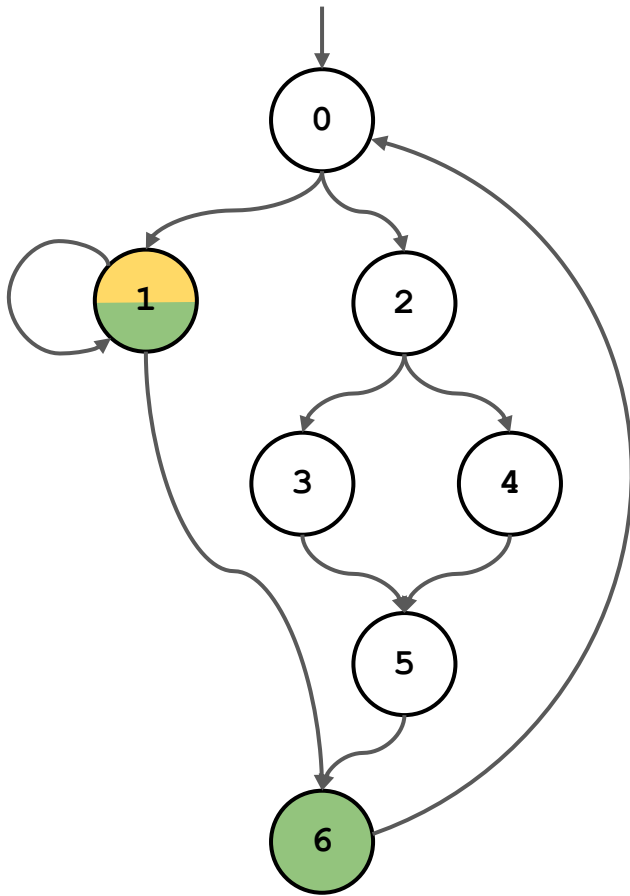


NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	\emptyset	
2		
3		
4		
5		
6		

A node **X** *dominates* a node **Y** iff every path from the entry point of the control flow graph to **Y** includes **X**.

A node **X** *strictly dominates* a node **Y** iff **X** dominates **Y** and **X** \neq **Y**

1 does not dominate 6 because there is a path from 5 that doesn't include 1. 1 does not strictly dominate itself

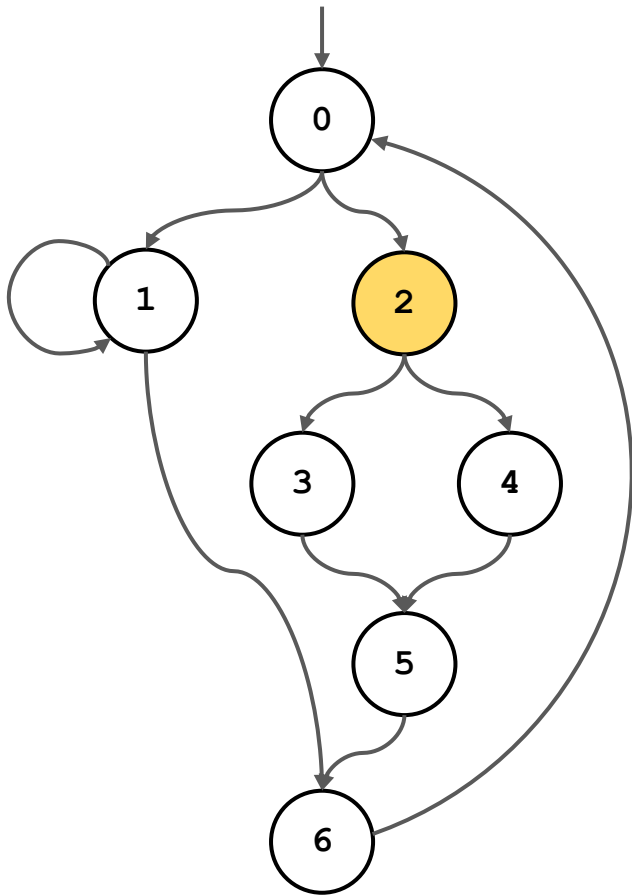


NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	\emptyset	1, 6
2		
3		
4		
5		
6		

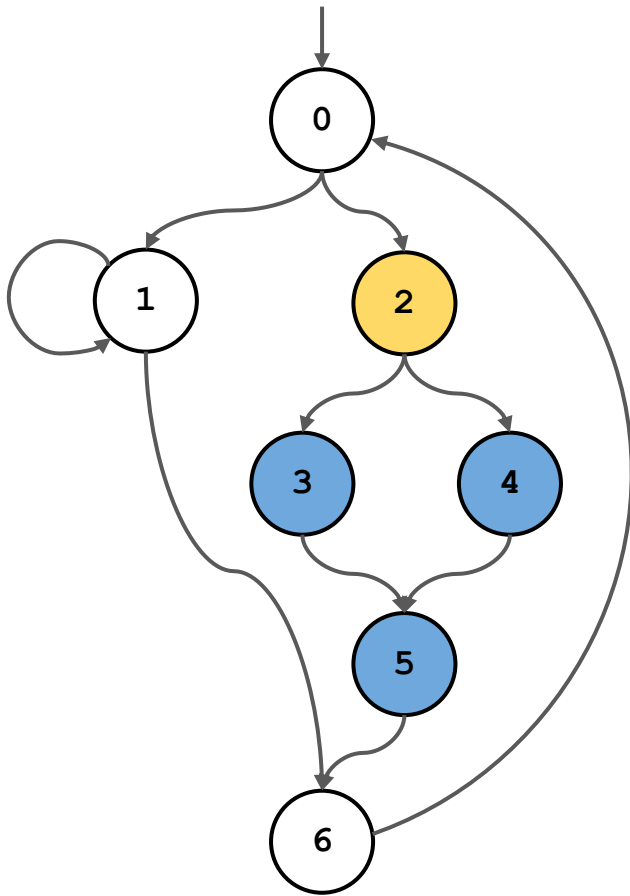
A node \mathbf{Y} is in the *dominance frontier* of node \mathbf{X} iff \mathbf{X} dominates an immediate predecessor of \mathbf{Y} but \mathbf{X} does not strictly dominate \mathbf{Y} .

$X = 1, Y = 6$, 1 dominates 1, 1 is an immediate predecessor of 6, 1 does not strictly dominate 6

$X = 1, Y = 1$, 1 dominates 1, 1 is an immediate predecessor of 1, 1 does not strictly dominate 1



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	∅	1, 6
2		
3		
4		
5		
6		

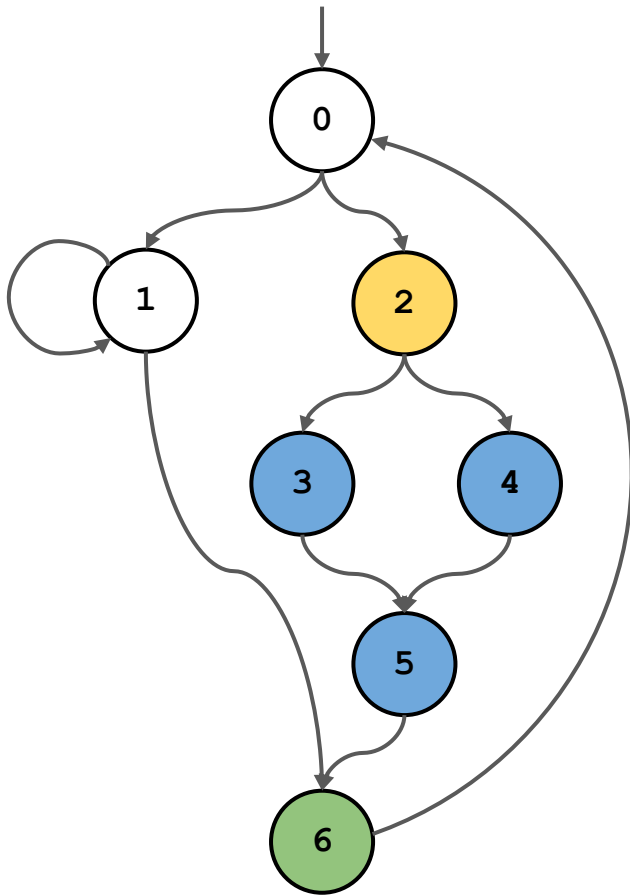


NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	\emptyset	1, 6
2	3, 4, 5	
3		
4		
5		
6		

A node \mathbf{x} *dominates* a node \mathbf{y} iff every path from the entry point of the control flow graph to \mathbf{y} includes \mathbf{x} .

A node \mathbf{x} *strictly dominates* a node \mathbf{y} iff \mathbf{x} dominates \mathbf{y} and $\mathbf{x} \neq \mathbf{y}$

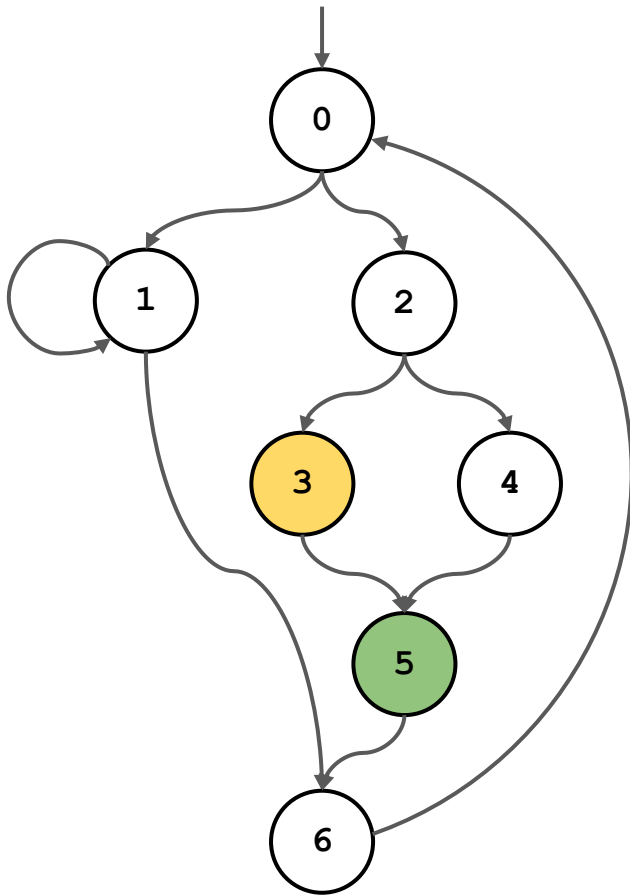
Need to go through 2 to get through 3, 4, 5 and 2 cannot strictly dominate itself



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	\emptyset	1, 6
2	3, 4, 5	6
3		
4		
5		
6		

A node Y is in the *dominance frontier* of node X iff X dominates an immediate predecessor of Y but X does not strictly dominate Y .

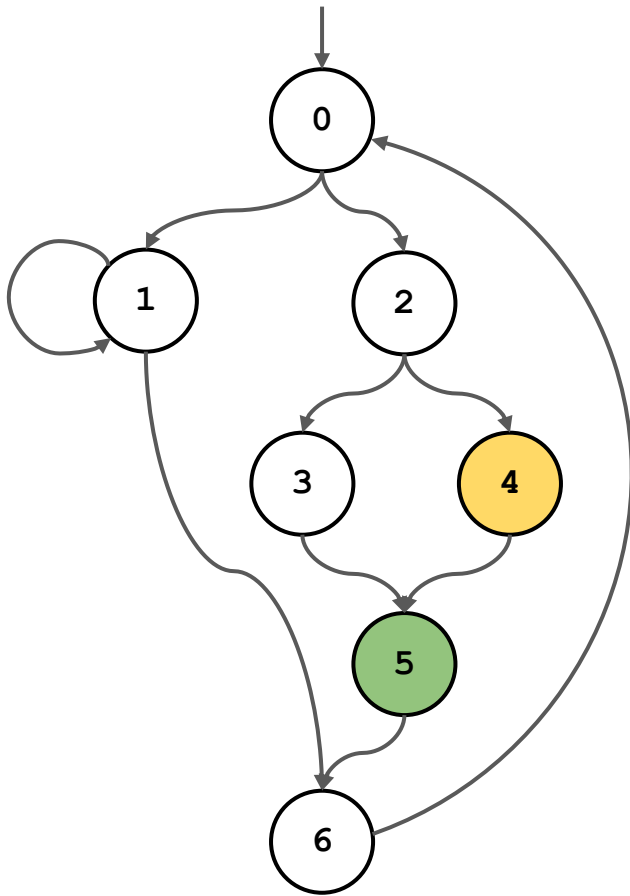
$X = 2, Y = 6$, 2 dominates 5, 5 is an immediate predecessor of 6, 2 does not strictly dominate 6



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	∅	1, 6
2	3, 4, 5	6
3	∅	5
4		
5		
6		

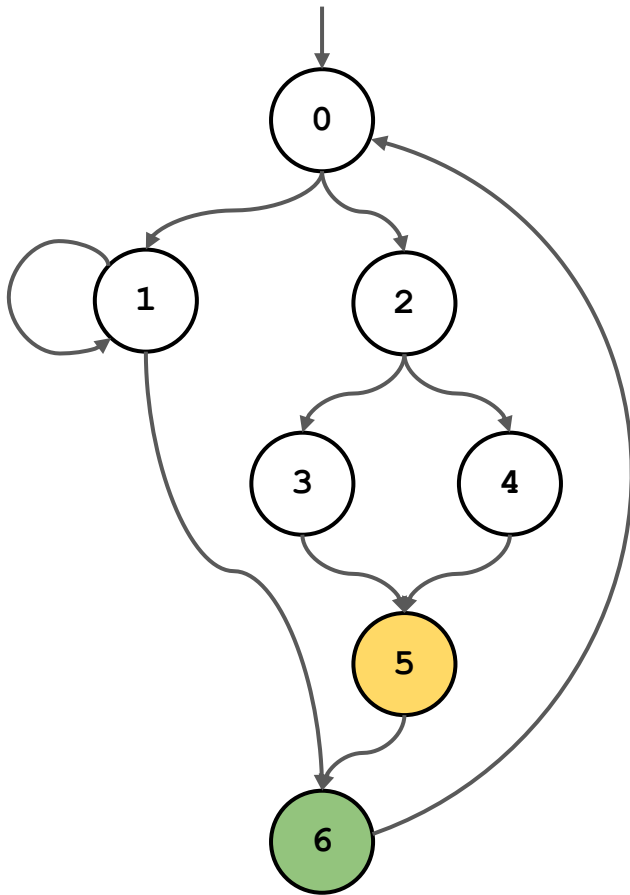
3 does not strictly dominate 5 (path through 4) and therefore does not strictly dominate anything else

3 dominates 3, 3 is an immediate predecessor of 5, 3 does not strictly dominate 5



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	∅	1, 6
2	3, 4, 5	6
3	∅	5
4	∅	5
5		
6		

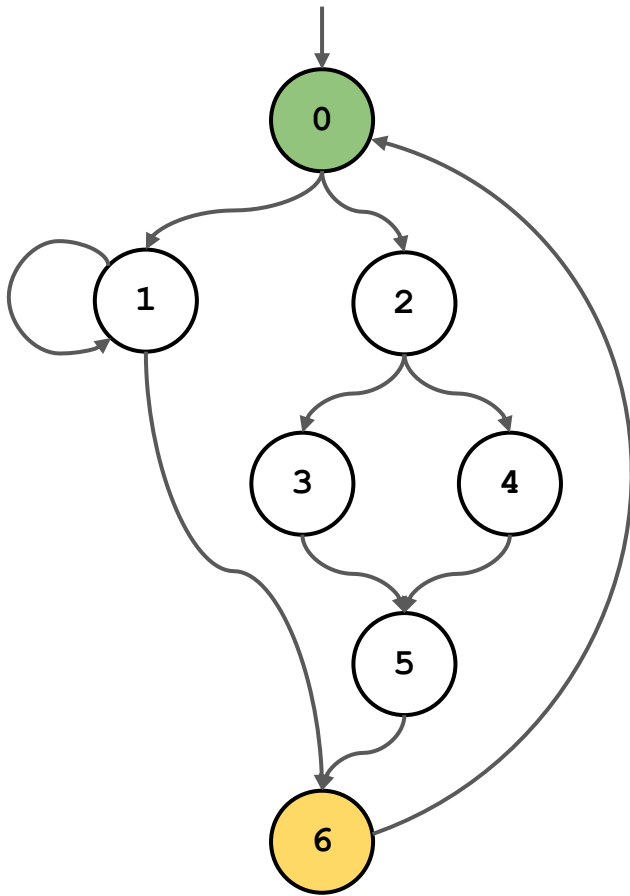
Same as previous slide but with 4 instead of 3



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	∅	1, 6
2	3, 4, 5	6
3	∅	5
4	∅	5
5	∅	6
6		

5 does not strictly dominate 6 (path through 1) and therefore does not strictly dominate anything else

5 dominates 5, 5 is an immediate predecessor of 6, 5 does not strictly dominate 6



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	\emptyset	1, 6
2	3, 4, 5	6
3	\emptyset	5
4	\emptyset	5
5	\emptyset	6
6	\emptyset	0

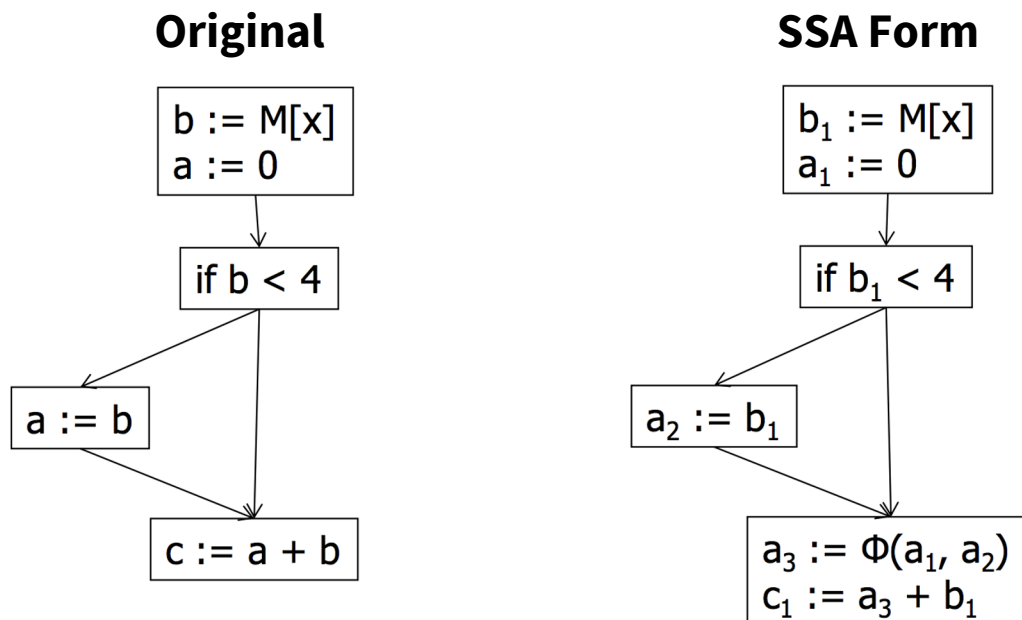
6 does not strictly dominate 0 (path through 0) and therefore does not strictly dominate anything else

6 dominates 6, 6 is an immediate predecessor of 0, 6 does not strictly dominate 0

Problem 2(b)

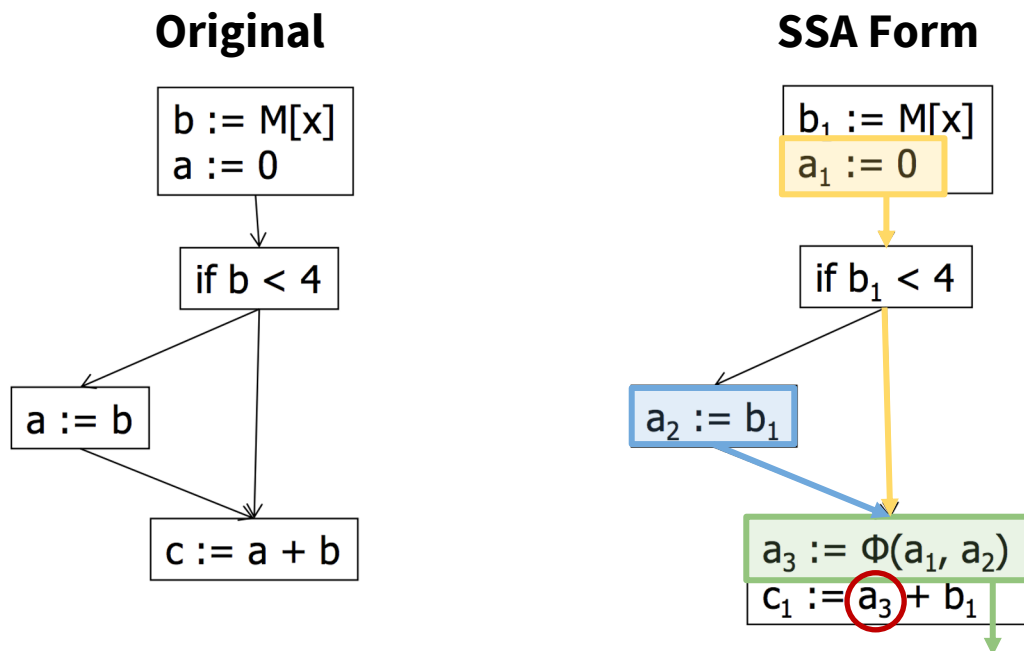
Phi-Functions

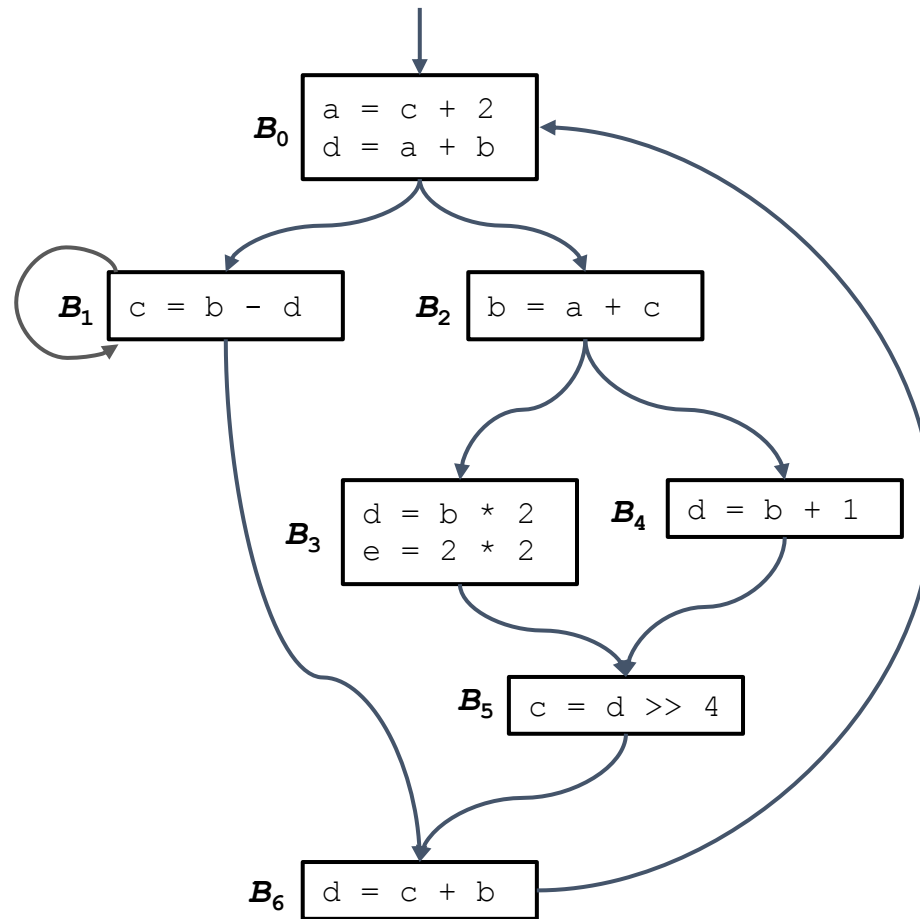
- A way to represent multiple possible values for a certain definition
 - Not a “real” instruction – just a form of bookkeeping needed for SSA



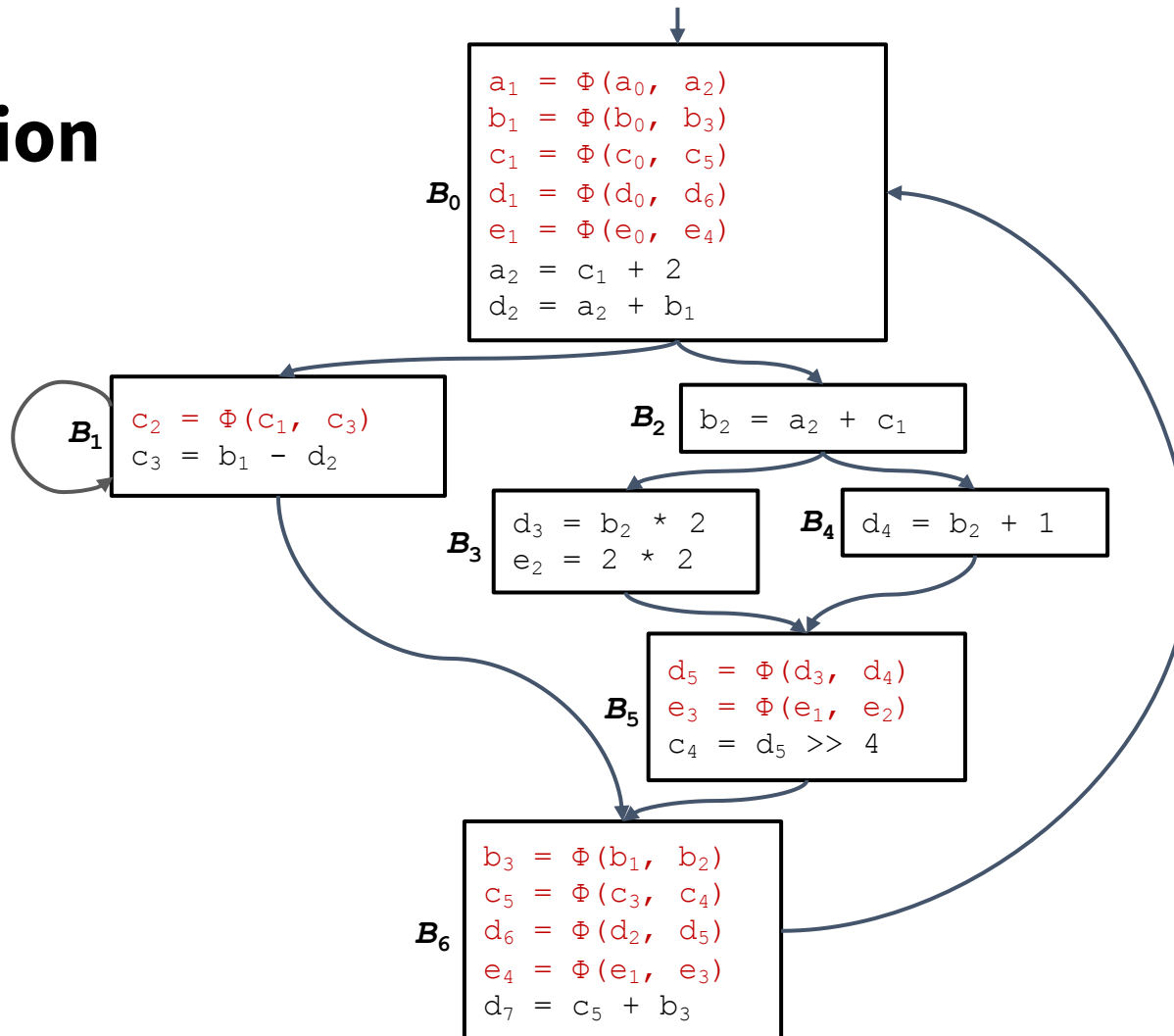
Where to place Phi-Functions?

- Wherever a variable has multiple possible definitions entering a block
 - Inefficient (and unnecessary!) to consider all possible phi-functions at the start of each block





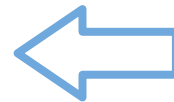
Solution



Converting to SSA

1

Compute the dominance frontier of each node



Already done (in problem 2a)

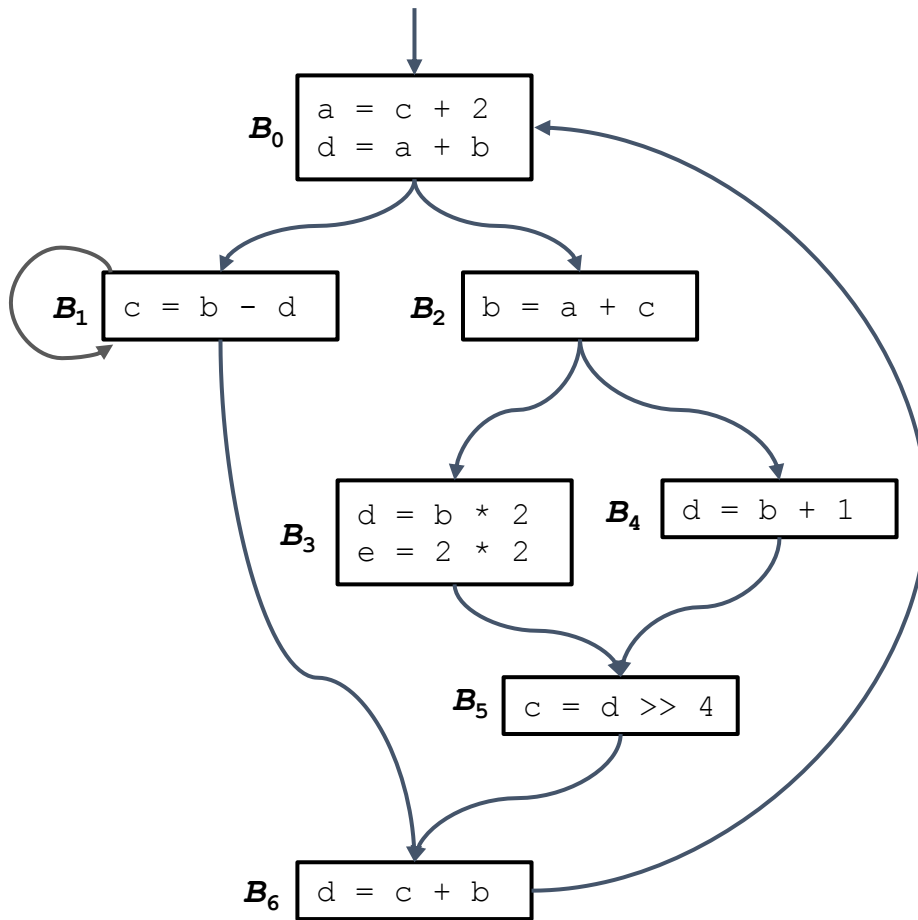
2

Determine which variables need merging in each node

3

Assign numbers to definitions and add phi functions

Step 1: Dominance Frontiers



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5, 6	0
1	\emptyset	1, 6
2	3, 4, 5	6
3	\emptyset	5
4	\emptyset	5
5	\emptyset	6
6	\emptyset	0

Converting to SSA

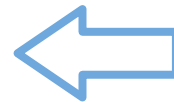
1

Compute the dominance frontier of each node



2

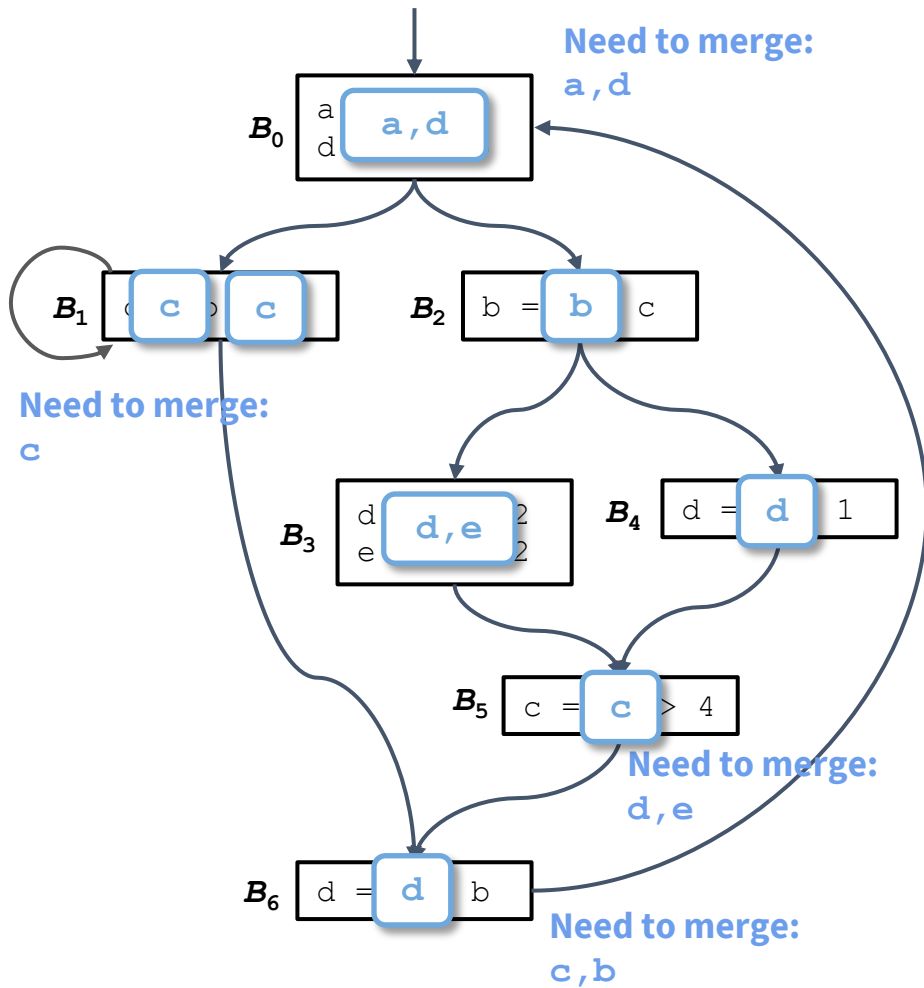
Determine which variables need merging in each node



We will compute using the dominance frontiers

3

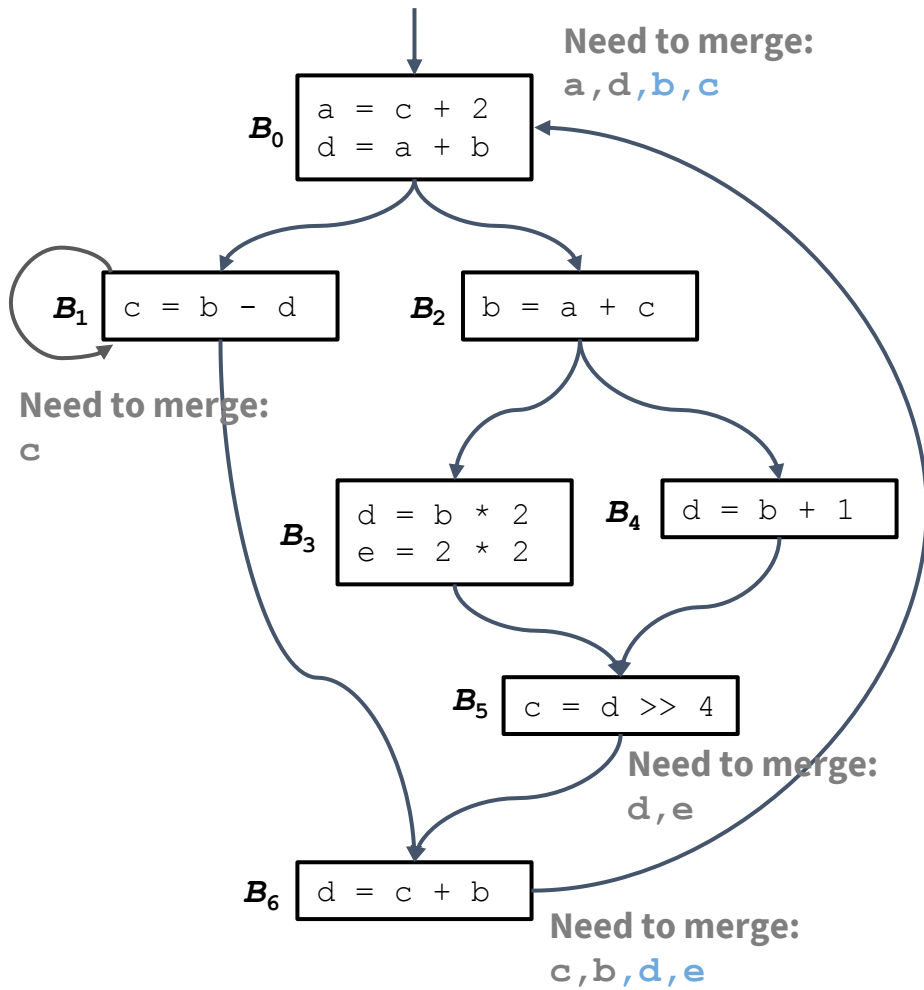
Assign numbers to definitions and add phi functions



Step 2: Determine Necessary Merges

ITERATION 1: Each node in the dominance frontier of node X will merge any definitions created in node X.

NODE		DOMINANCE FRONTIER
0	<code>a, d</code>	0
1	<code>c</code>	1, 6
2	<code>b</code>	6
3	<code>d, e</code>	5
4	<code>d</code>	5
5	<code>c</code>	6
6	<code>d</code>	0



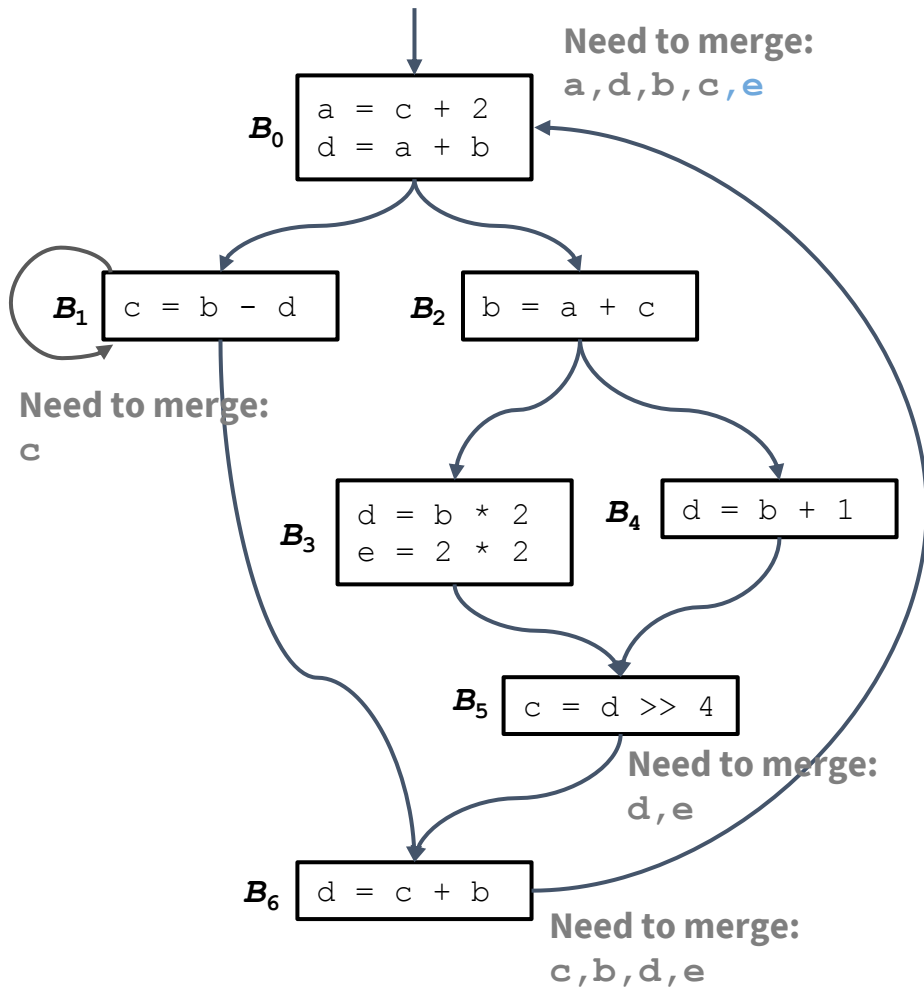
Step 2: Determine Necessary Merges

ITERATION 2: Each merge will create a new definition, which may need merging again.

NODE	DOMINANCE FRONTIER
0	0
1	1, 6
2	6
3	5
4	5
5	6
6	0

Annotations for Node 5: d, e

Annotations for Node 6: b, c



Step 2: Determine Necessary Merges

ITERATION 3: Each merge will create a new definition, which may need merging again.

NODE	DOMINANCE FRONTIER
0	0
1	1, 6
2	6
3	5
4	5
5	6
6	0

A blue arrow points from node 6 in the NODE column to node 0 in the DOMINANCE FRONTIER column, labeled d, e .

Converting to SSA

1

Compute the dominance frontier of each node



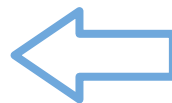
2

Determine which variables need merging in each node



3

Assign numbers to definitions and add phi functions



Place phi functions first, then increment subscripts

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

$$\mathbf{B}_0 \quad \begin{array}{l} a = c + 2 \\ d = a + b \end{array}$$

Need to merge:
a, b, c, d, e



$$\mathbf{B}_0 \quad \begin{array}{l} a_1 = \Phi(a_0, a_2) \\ b_1 = \Phi(b_0, b_3) \\ c_1 = \Phi(c_0, c_5) \\ d_1 = \Phi(d_0, d_7) \\ e_1 = \Phi(e_0, e_4) \\ a_2 = c_1 + 2 \\ d_2 = a_2 + b_1 \end{array}$$

Note: these subscripts determined after doing the rest of the CFG!

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

$$B_1 \quad c = b - d$$

Need to merge:

c



$$B_1 \quad \begin{array}{l} c_2 = \Phi(c_1, c_3) \\ c_3 = b_1 - d_2 \end{array}$$

Note: must merge its own (later) definition because of the back-edge!

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

$$B_2 \quad b = a + c$$



$$B_2 \quad b_2 = a_2 + c_1$$

Nothing to merge

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

$$\mathbf{B}_3 \begin{array}{l} d = b * 2 \\ e = 2 * 2 \end{array}$$



$$\mathbf{B}_3 \begin{array}{l} d_3 = b_2 * 2 \\ e_2 = 2 * 2 \end{array}$$

Nothing to merge

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

$$B_4 \quad d = b + 1$$



$$B_4 \quad d_4 = b_2 + 1$$

Nothing to merge

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

$$B_5 \quad c = d \gg 4$$



$$B_5 \quad \begin{array}{l} d_5 = \Phi(d_3, d_4) \\ e_3 = \Phi(e_1, e_2) \\ c_4 = d_5 \gg 4 \end{array}$$

Need to merge:
d, e

Step 3: Assign Definition Numbers

Merges go first, and each successive definition of a variable should increment its index by 1.

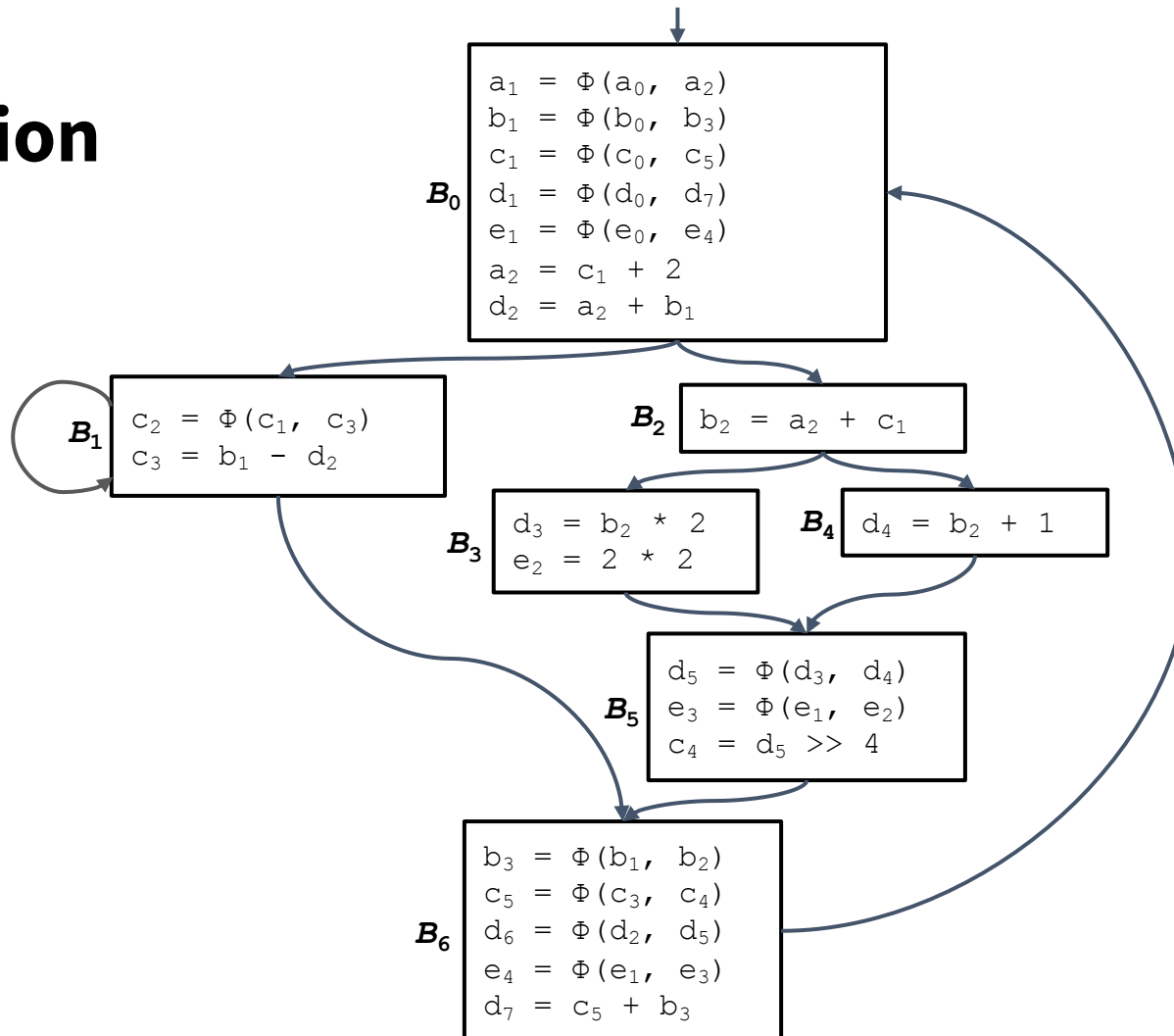
$$B_6 \quad d = c + b$$

Need to merge:
 b, c, d, e

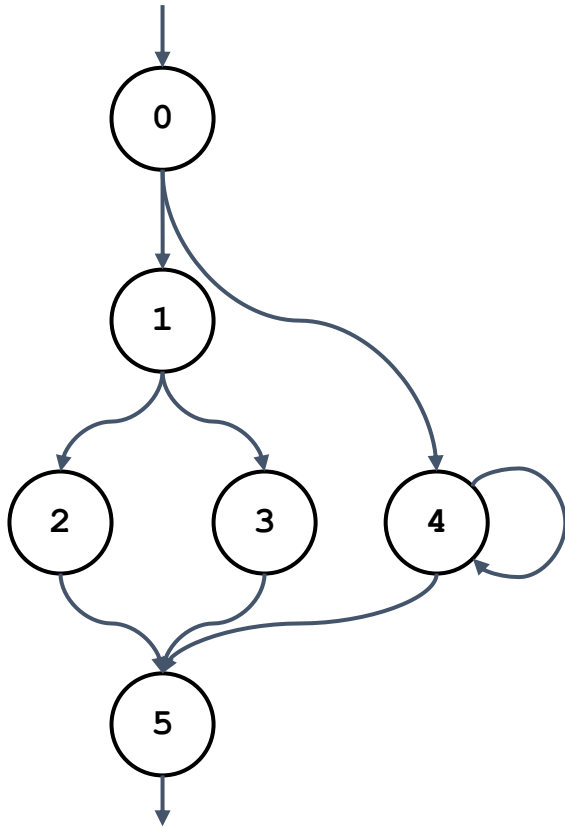


$$B_6 \quad \begin{array}{l} b_3 = \Phi(b_1, b_2) \\ c_5 = \Phi(c_3, c_4) \\ d_6 = \Phi(d_2, d_5) \\ e_4 = \Phi(e_1, e_3) \\ d_7 = c_5 + b_3 \end{array}$$

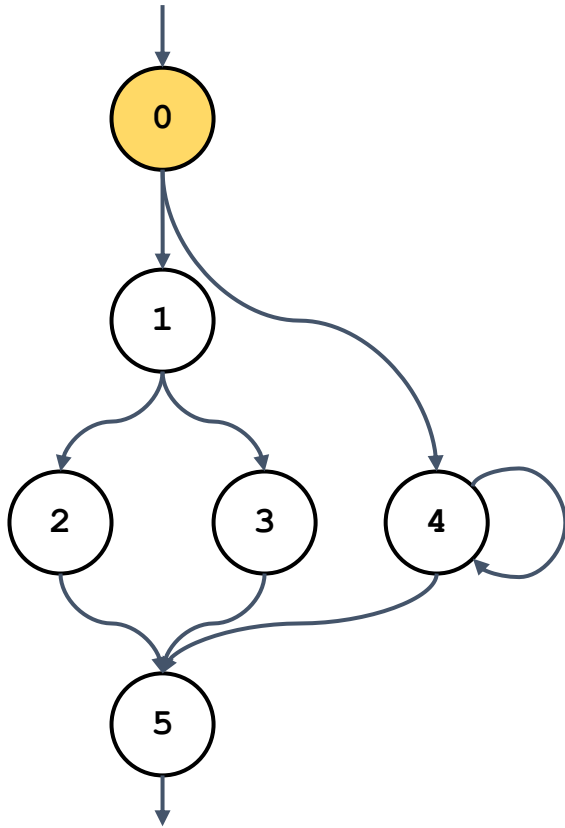
Solution



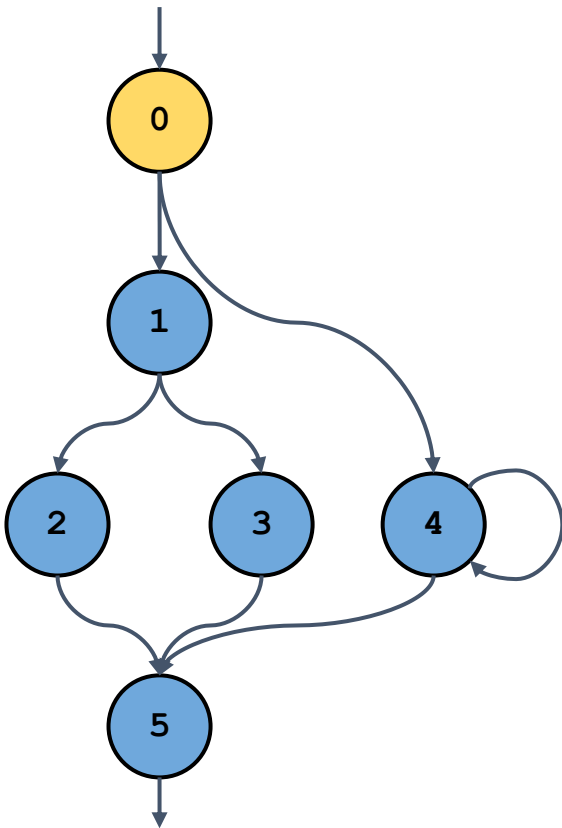
Problem 3



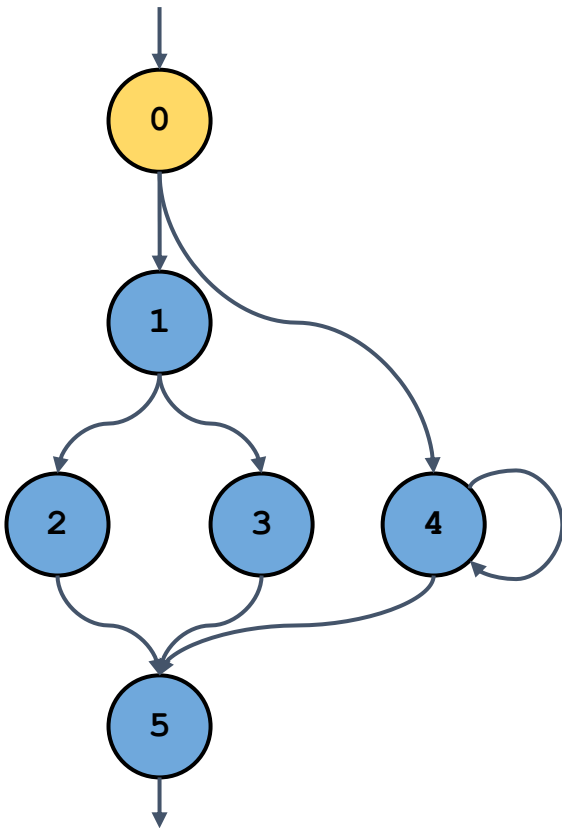
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0		
1		
2		
3		
4		
5		



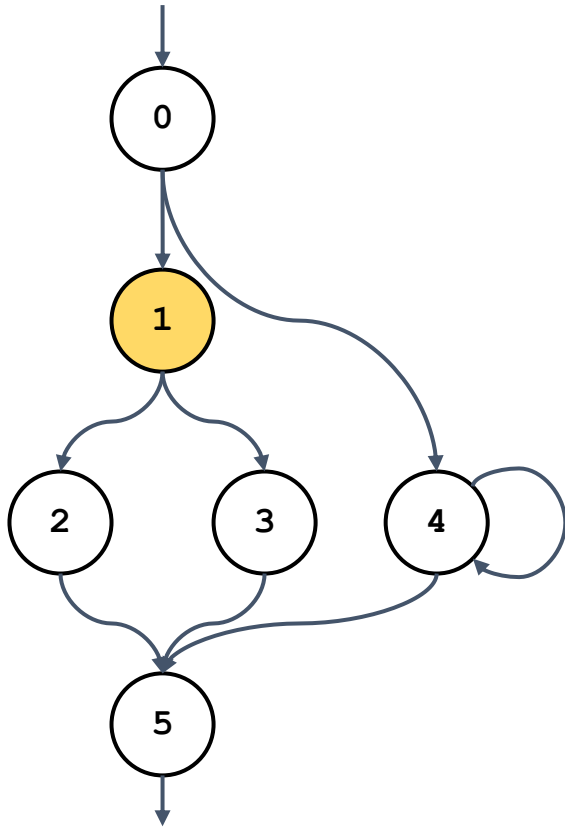
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0		
1		
2		
3		
4		
5		



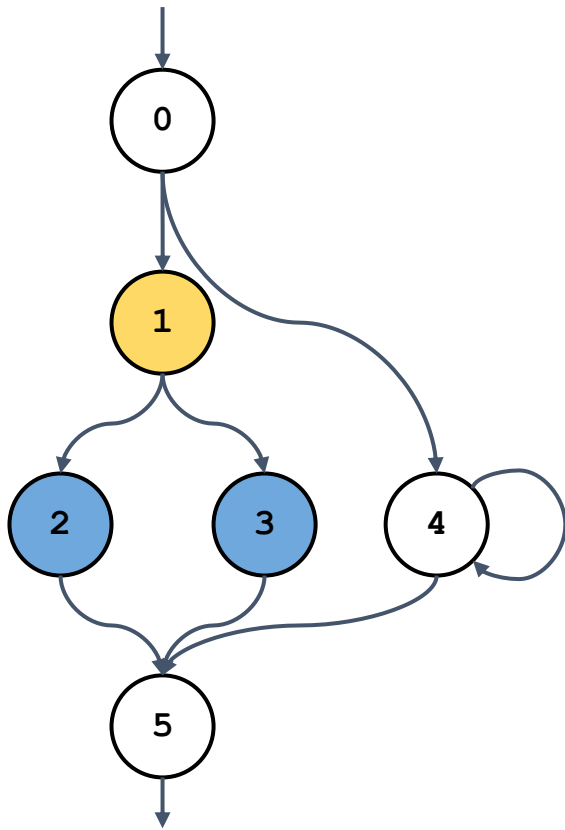
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	
1		
2		
3		
4		
5		



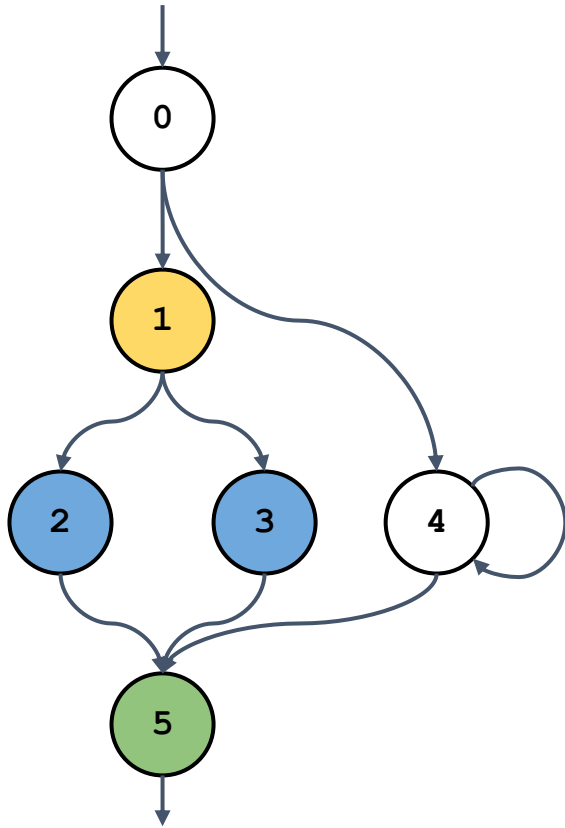
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	∅
1		
2		
3		
4		
5		



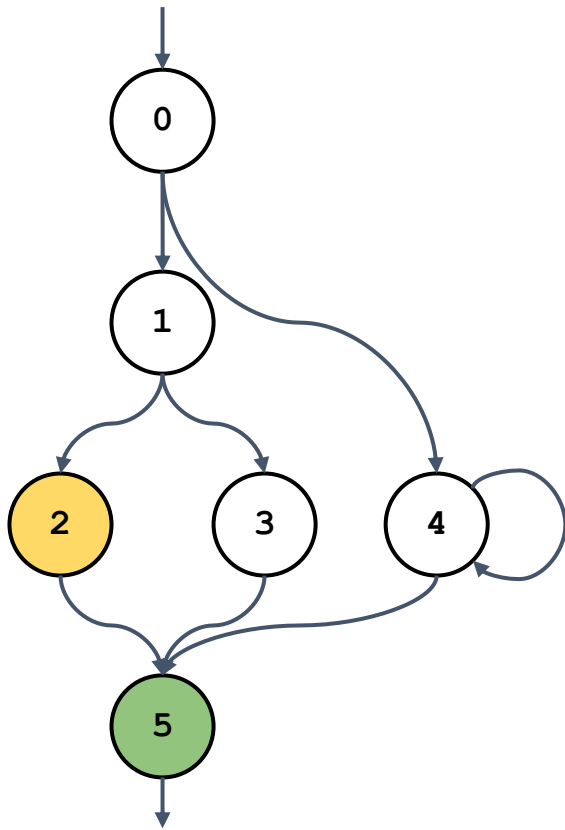
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1		
2		
3		
4		
5		



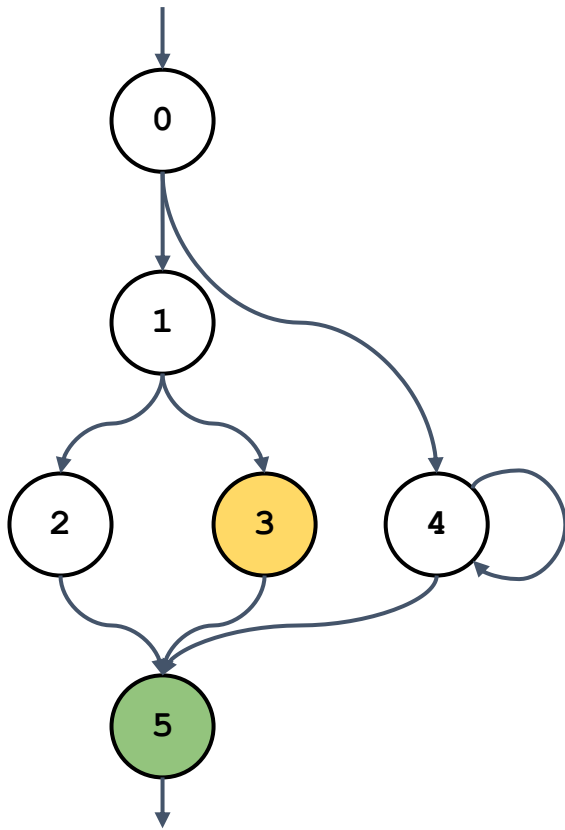
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1	2, 3	
2		
3		
4		
5		



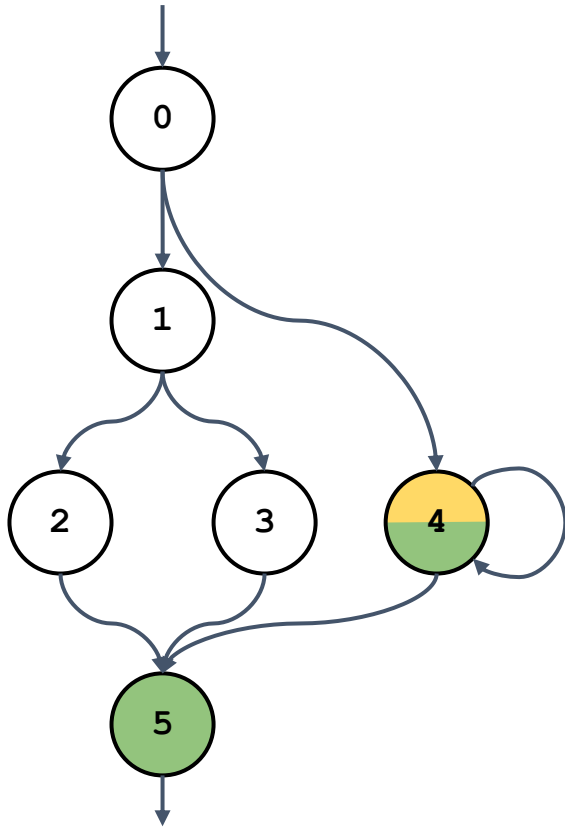
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1	2, 3	5
2		
3		
4		
5		



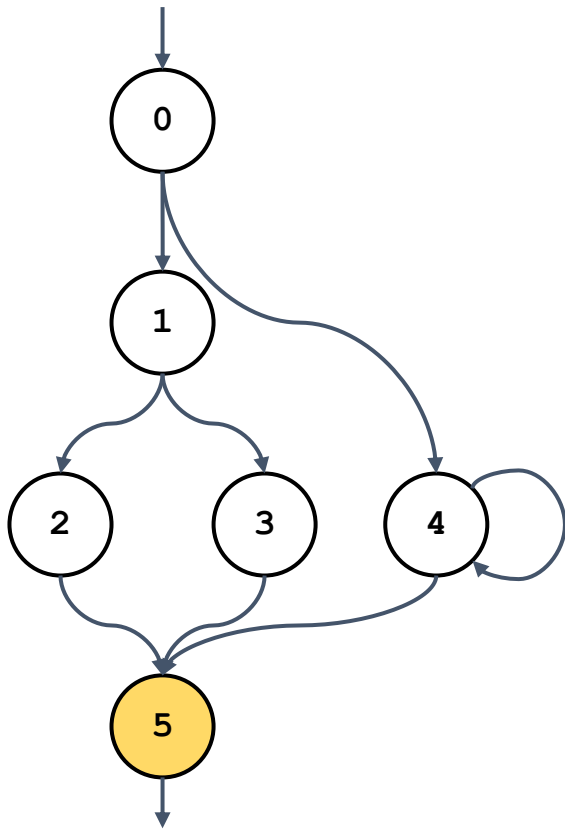
NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1	2, 3	5
2	\emptyset	5
3		
4		
5		



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1	2, 3	5
2	\emptyset	5
3	\emptyset	5
4		
5		



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1	2, 3	5
2	\emptyset	5
3	\emptyset	5
4	\emptyset	4, 5
5		



NODE	STRICTLY DOMINATES	DOMINANCE FRONTIER
0	1, 2, 3, 4, 5	\emptyset
1	2, 3	5
2	\emptyset	5
3	\emptyset	5
4	\emptyset	4, 5
5	\emptyset	\emptyset