Análisis de Flujo

CI4722 – Lenguajes de Programación III

Ernesto Hernández-Novich
<emhn@usb.ve>

Universidad "Simón Bolívar"

Copyright ©2012-2016

UNIVERSIDAD SIMÓN BOLÍVAR

Análisis de Ciclos

- Los algoritmos de análisis de flujo estudiados hasta ahora no hacen nada particularmente especial con los ciclos.
  - "Promueven" los conjuntos a través del grafo de flujo.
  - Avanzan un paso a la vez.
  - Si hay un ciclo, la clausura deja de agregar elementos eventualmente.
- Algunas mejoras requieren identificar los ciclos con precisión.
- Tiempo de convergencia de los algoritmos de análisis de flujo de datos proporcional a la complejidad de la estructura de ciclos.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores (*Dominators*)

Es una relación

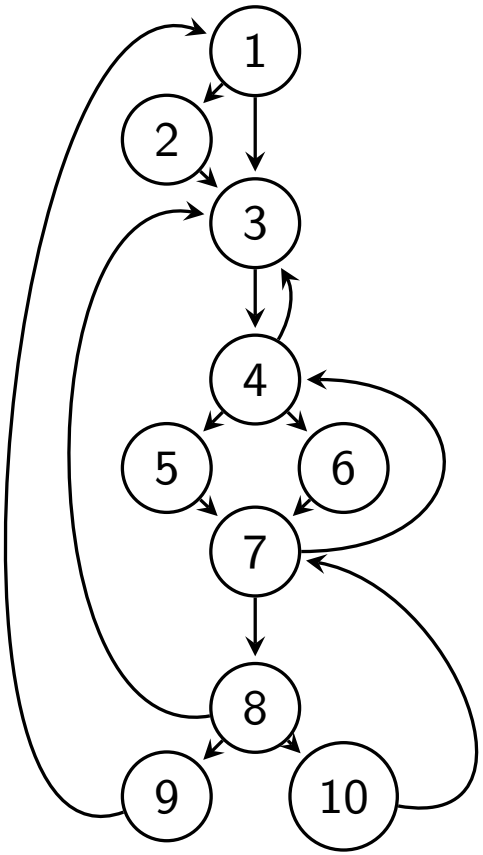
$$d \ dom \ n$$

- El nodo d **domina** al nodo n si todo camino desde el punto de entrada hasta n, pasa por d.
- Todo nodo se domina a sí mismo.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

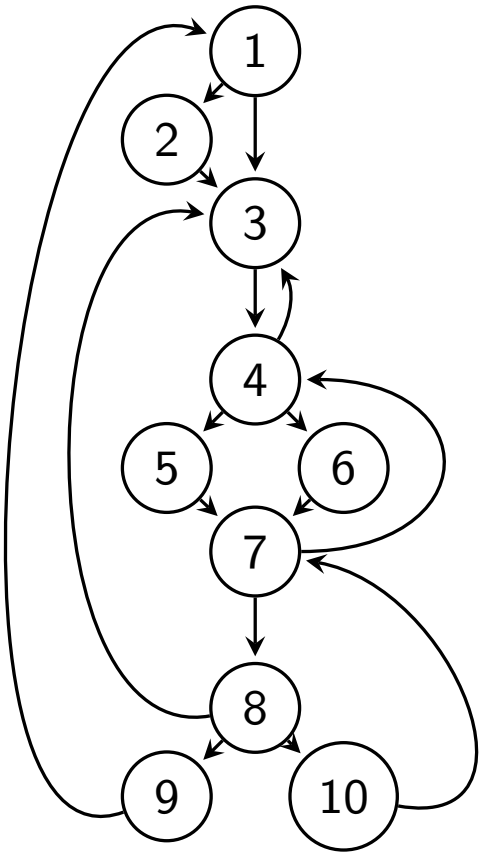
Basta mirar las posibilidades de flujo



UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

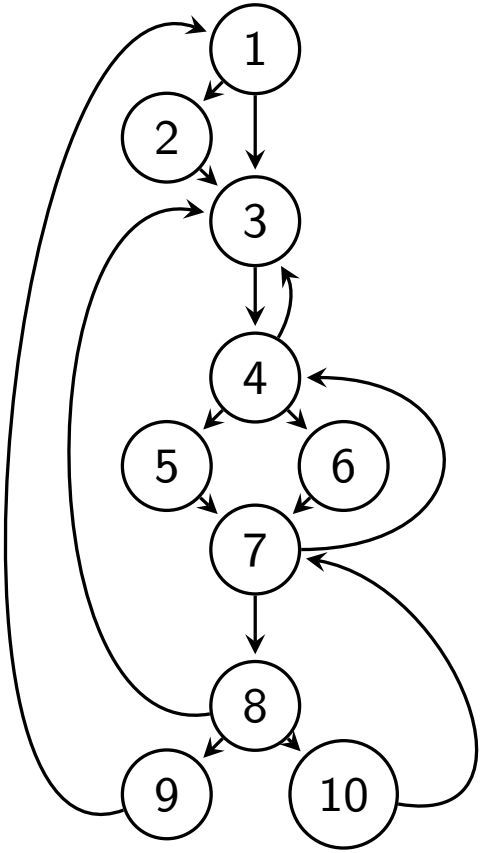


- 1 domina a todos los nodos – es el nodo de entrada.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

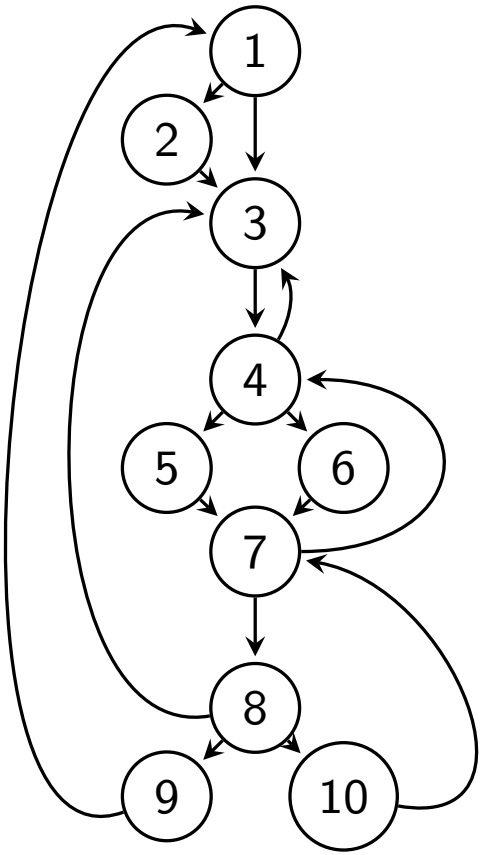


- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

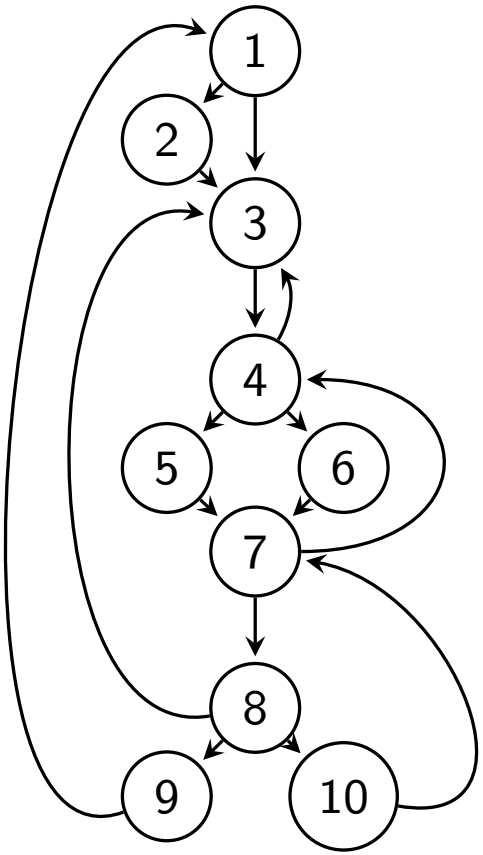


- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.
- 3 domina a todos, excepto 1 y 2.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

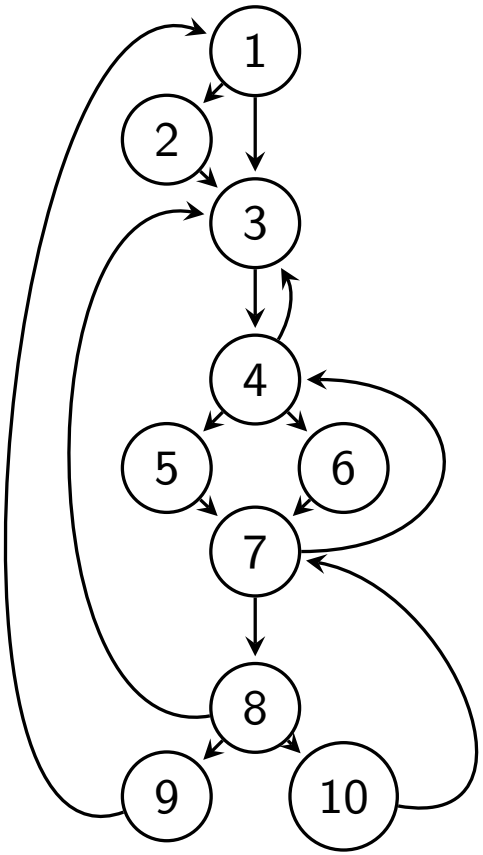


- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.
- 3 domina a todos, excepto 1 y 2.
- 4 domina a todos, excepto 1, 2 y 3.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

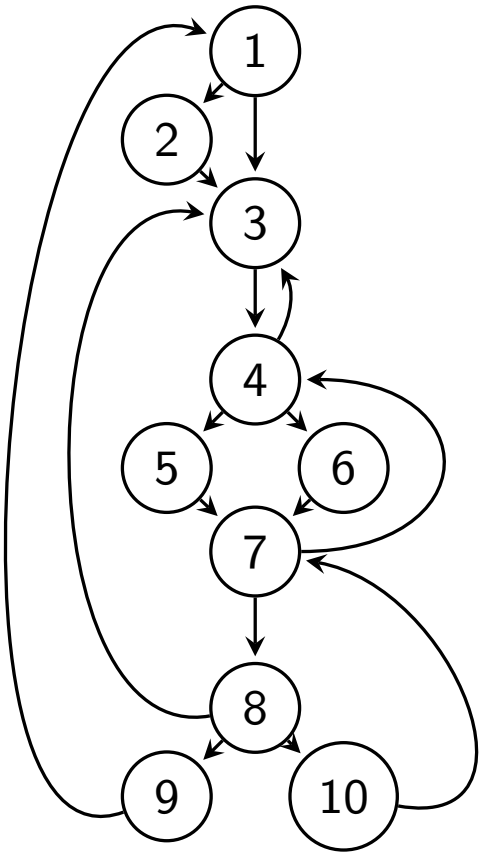


- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.
- 3 domina a todos, excepto 1 y 2.
- 4 domina a todos, excepto 1, 2 y 3.
- 5 y 6 sólo se dominan a si mismos.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

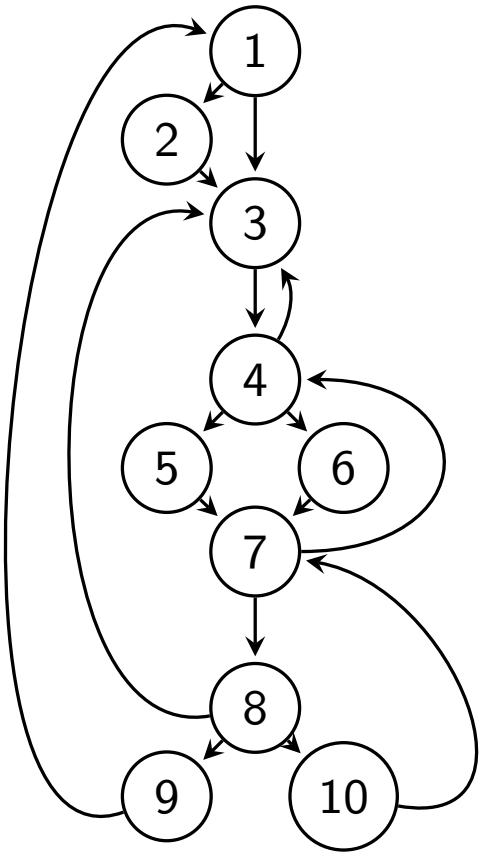


- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.
- 3 domina a todos, excepto 1 y 2.
- 4 domina a todos, excepto 1, 2 y 3.
- 5 y 6 sólo se dominan a si mismos.
- 7 domina a 7, 8, 9 y 10.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo

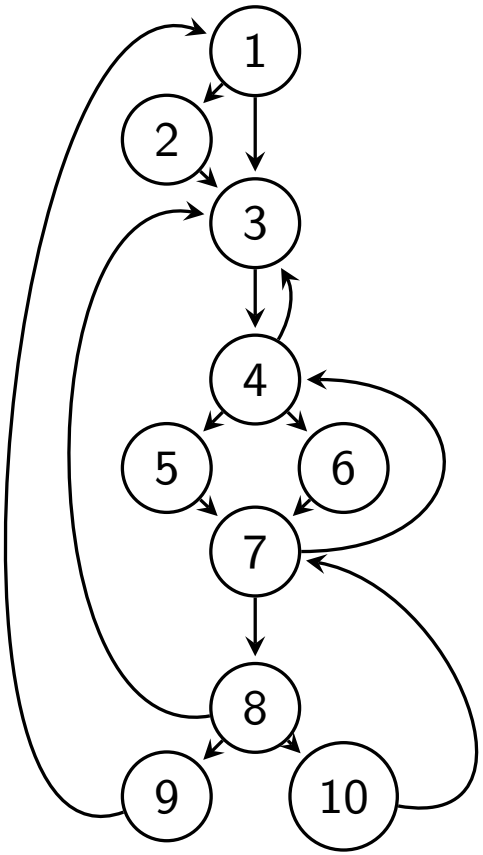


- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.
- 3 domina a todos, excepto 1 y 2.
- 4 domina a todos, excepto 1, 2 y 3.
- 5 y 6 sólo se dominan a si mismos.
- 7 domina a 7, 8, 9 y 10.
- 8 domina a 9 y 10.

UNIVERSIDAD SIMÓN BOLÍVAR

Dominadores

Basta mirar las posibilidades de flujo



- 1 domina a todos los nodos – es el nodo de entrada.
- 2 sólo se domina a si mismo.
- 3 domina a todos, excepto 1 y 2.
- 4 domina a todos, excepto 1, 2 y 3.
- 5 y 6 sólo se dominan a si mismos.
- 7 domina a 7, 8, 9 y 10.
- 8 domina a 9 y 10.
- 9 y 10 sólo se dominan a si mismos.

UNIVERSIDAD SIMÓN BOLÍVAR

Propiedades de la Relación

- Dado *cualquier* camino sin ciclos desde la entrada hasta el nodo n
  - Todos los dominadores de n aparecen en el camino.
  - Aparecen en el mismo orden en cualquier camino desde la entrada.

UNIVERSIDAD SIMÓN BOLÍVAR

Propiedades de la Relación

- Dado *cualquier* camino sin ciclos desde la entrada hasta el nodo n
  - Todos los dominadores de n aparecen en el camino.
  - Aparecen en el mismo orden en cualquier camino desde la entrada.
- Transitiva – $(a\ dom\ b \wedge b\ dom\ c) \Rightarrow a\ dom\ c$

UNIVERSIDAD SIMÓN BOLÍVAR

Propiedades de la Relación

- Dado *cualquier* camino sin ciclos desde la entrada hasta el nodo n
  - Todos los dominadores de n aparecen en el camino.
  - Aparecen en el mismo orden en cualquier camino desde la entrada.
- Transitiva – $(a\ dom\ b \wedge b\ dom\ c) \Rightarrow a\ dom\ c$
- Antisimétrica – $(a\ dom\ b \wedge b\ dom\ a) \Rightarrow a = b$

UNIVERSIDAD SIMÓN BOLÍVAR

Propiedades de la Relación

- Dado *cualquier* camino sin ciclos desde la entrada hasta el nodo n
  - Todos los dominadores de n aparecen en el camino.
  - Aparecen en el mismo orden en cualquier camino desde la entrada.
- Transitiva – $(a\ dom\ b \wedge b\ dom\ c) \Rightarrow a\ dom\ c$
- Antisimétrica – $(a\ dom\ b \wedge b\ dom\ a) \Rightarrow a = b$
- $(a\ dom\ n \wedge b\ dom\ n) \Rightarrow (a\ dom\ b \vee b\ dom\ a)$

UNIVERSIDAD SIMÓN BOLÍVAR

Propiedades de la Relación

- Dado *cualquier* camino sin ciclos desde la entrada hasta el nodo n
  - Todos los dominadores de n aparecen en el camino.
  - Aparecen en el mismo orden en cualquier camino desde la entrada.
- Transitiva – $(a\ dom\ b \wedge b\ dom\ c) \Rightarrow a\ dom\ c$
- Antisimétrica – $(a\ dom\ b \wedge b\ dom\ a) \Rightarrow a = b$
- $(a\ dom\ n \wedge b\ dom\ n) \Rightarrow (a\ dom\ b \vee b\ dom\ a)$
- **Dominador Inmediato Unico** (*Immediate Unique Dominator*)
  - Aquel dominador de n que aparece justo antes de n en cualquier camino acíclico desde la entrada.
  - El nodo de entrada no tiene Dominador Inmediato Unico – el resto de los nodos siempre tiene Dominador Inmediato Unico.

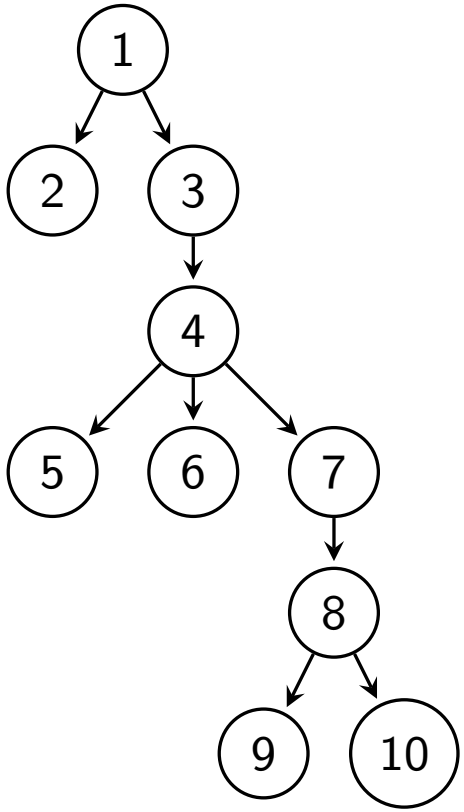
Esto da lugar a un árbol...

UNIVERSIDAD SIMÓN BOLÍVAR

Arbol de dominadores (*Dominator Tree*)

Basta expresar la relación

- El nodo de entrada es la raíz.
- Los descendientes de cada nodo son aquellos nodos a los cuales domina.



UNIVERSIDAD SIMÓN BOLÍVAR

¿Cómo calcular los dominadores?

- Si $p_1, p_2, \ldots p_k$ son los predecesores de n, y $d \neq n$, entonces d *dom* n si y sólo si d *dom* p_i

UNIVERSIDAD SIMÓN BOLÍVAR

¿Cómo calcular los dominadores?

- Si $p_1, p_2, \ldots p_k$ son los predecesores de n, y $d \neq n$, entonces d *dom* n si y sólo si d *dom* p_i
- ¡Es un problema de análisis de flujo de datos!
  - Dominio – partes de N (N – conjunto de nodos del grafo).
  - Dirección – *forward flow*.
  - Base – $OUT[ENTRY] = \{ENTRY\}$
  - Función de Transferencia

$$f_B(x) = x \cup \{B\}$$

  - Inicialización – $OUT[B] = N$
  - Ecuaciones

$$OUT[B] = f_B(IN[B])$$

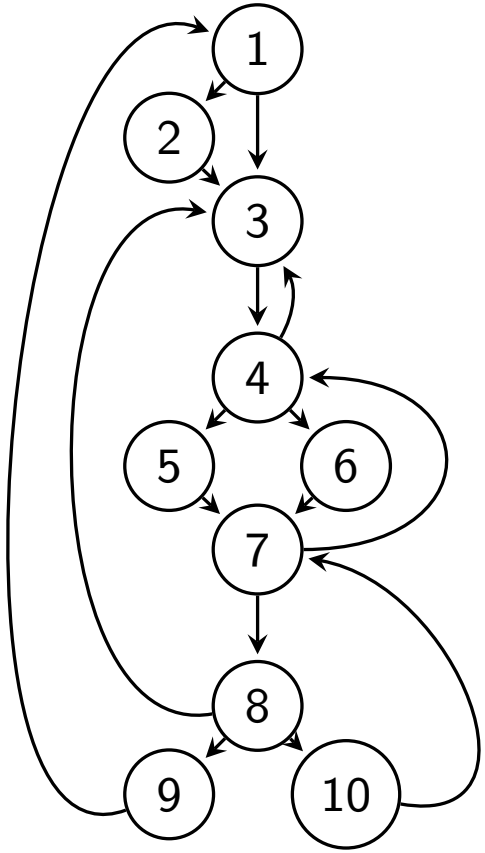
$$IN[B] = \bigcap_{P \text{ predecesor de } B} OUT[P]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Inicialización

Bloque	*IN*	*OUT*
1		1
2		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
3		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
4		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
5		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
6		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
7		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10

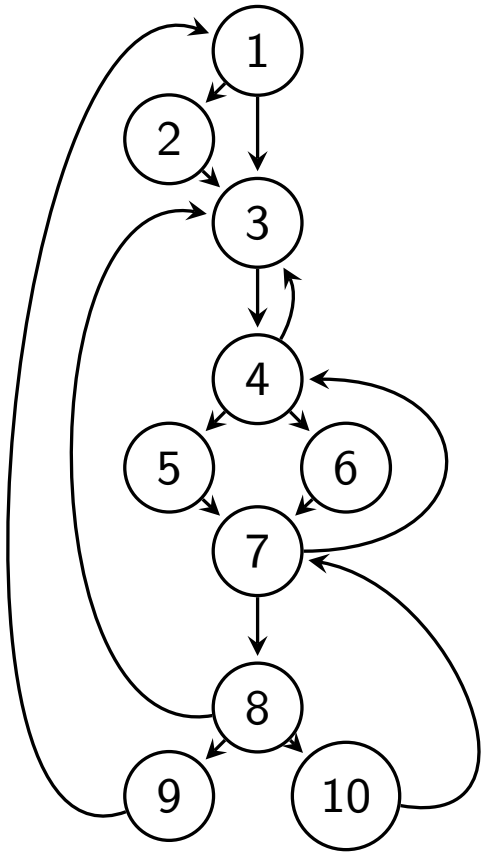


UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 2

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
4		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
5		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
6		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
7		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



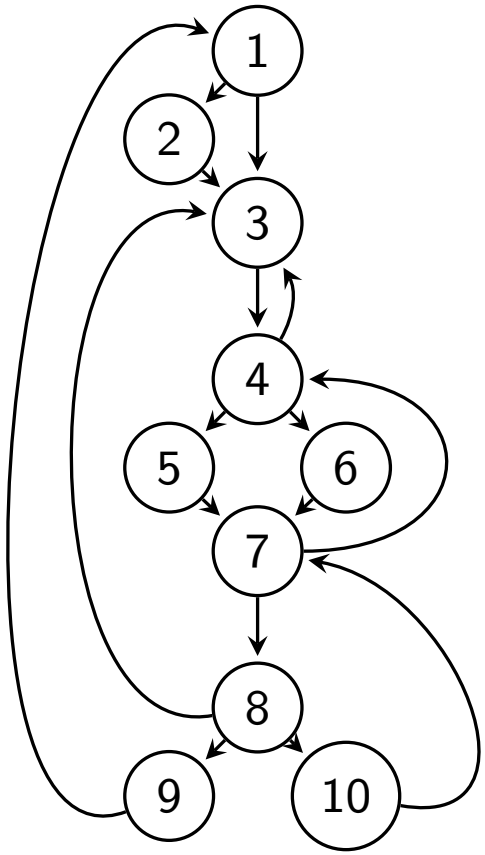
$$OUT[2] = \{2\} \cup OUT[1]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 3

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
5		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
6		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
7		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



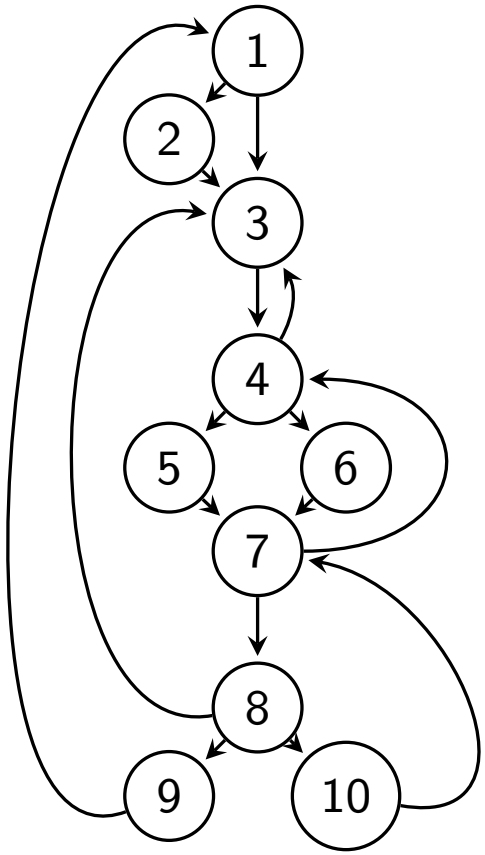
$$OUT[3] = \{3\} \cup (OUT[1] \cap OUT[2] \cap OUT[4] \cap OUT[8])$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 4

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
6		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
7		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



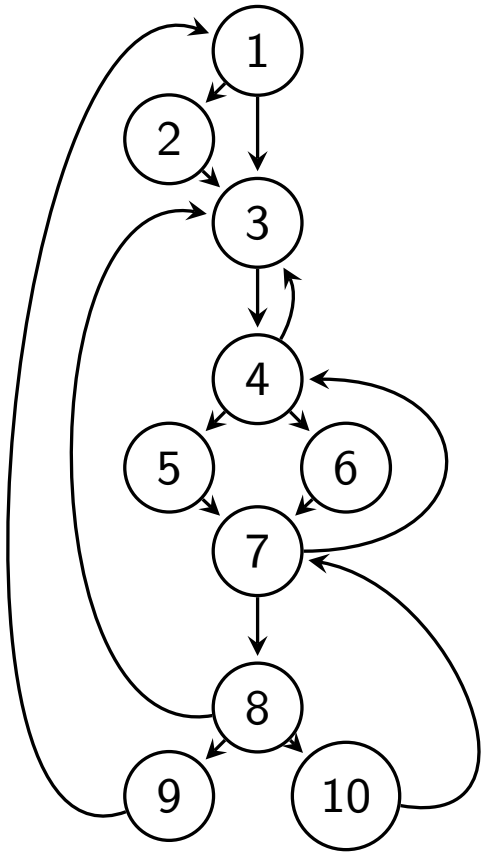
$$OUT[4] = \{4\} \cup (OUT[3] \cap OUT[7])$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 5

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
7		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



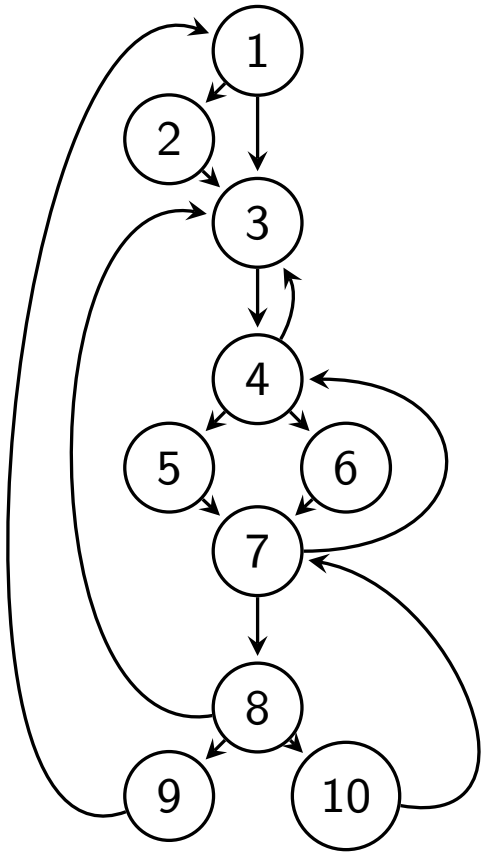
$$OUT[5] = \{5\} \cup OUT[4]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 6

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6	1, 3, 4	1, 3, 4, 6
7		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



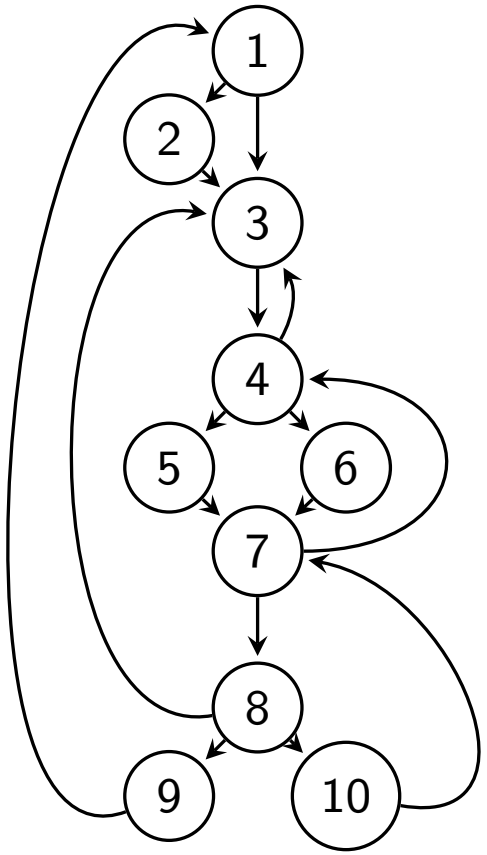
$$OUT[6] = \{6\} \cup OUT[4]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 7

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6	1, 3, 4	1, 3, 4, 6
7	1, 3, 4	1, 3, 4, 7
8		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



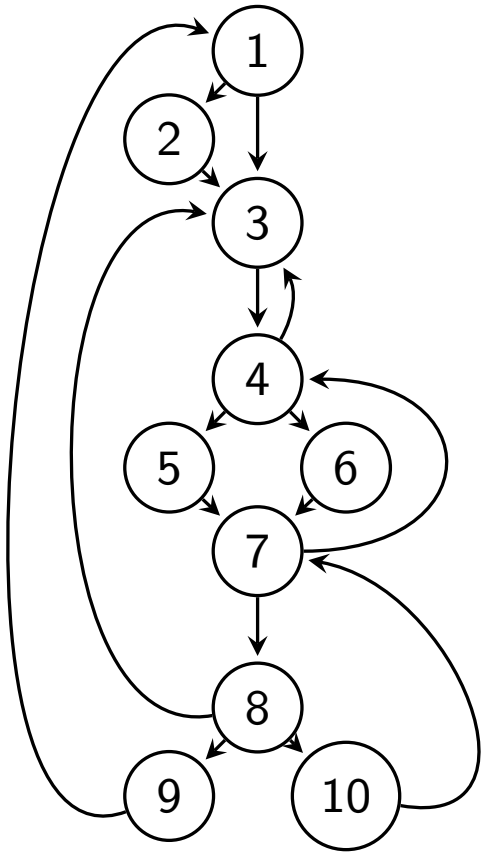
$$OUT[7] = \{7\} \cup (OUT[5] \cap OUT[6] \cap OUT[10])$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 8

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6	1, 3, 4	1, 3, 4, 6
7	1, 3, 4	1, 3, 4, 7
8	1, 3, 4, 7	1, 3, 4, 7, 8
9		1, 2, 3, 4, 5, 6, 7, 8, 9, 10
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



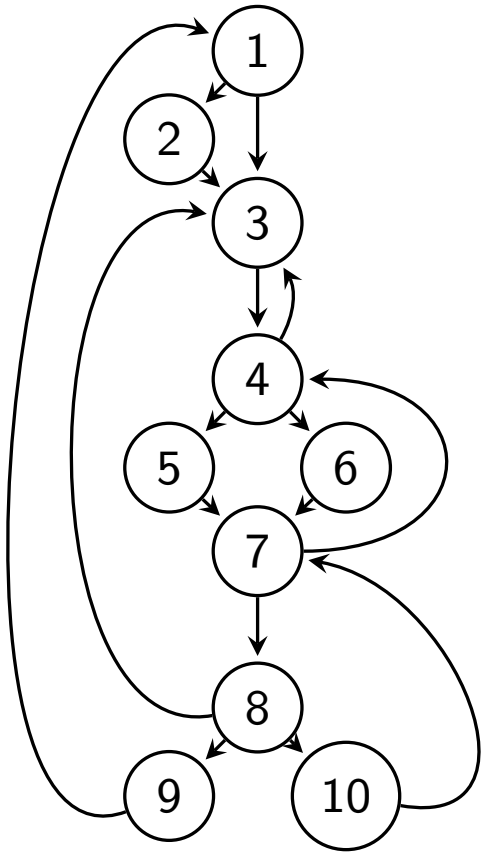
$$OUT[8] = \{8\} \cup OUT[7]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 9

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6	1, 3, 4	1, 3, 4, 6
7	1, 3, 4	1, 3, 4, 7
8	1, 3, 4, 7	1, 3, 4, 7, 8
9	1, 3, 4, 7, 8	1, 3, 4, 7, 8, 9
10		1, 2, 3, 4, 5, 6, 7, 8, 9, 10



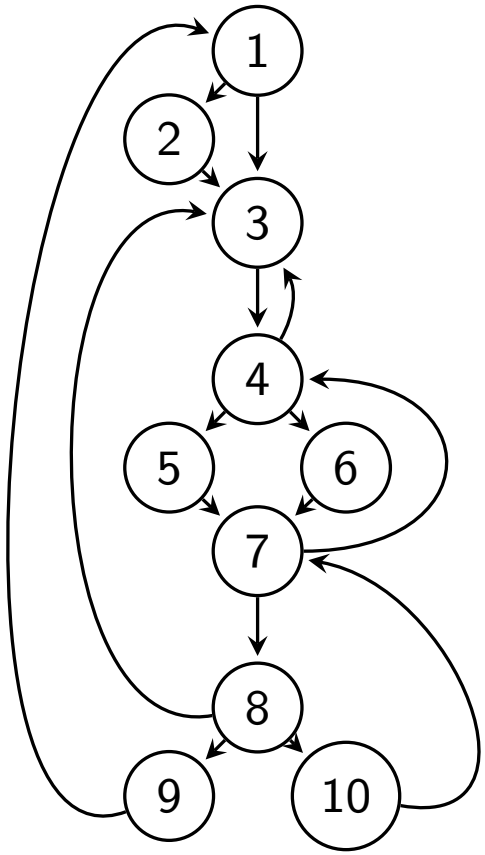
$$OUT[9] = \{9\} \cup OUT[8]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

Procesar el Nodo 10

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6	1, 3, 4	1, 3, 4, 6
7	1, 3, 4	1, 3, 4, 7
8	1, 3, 4, 7	1, 3, 4, 7, 8
9	1, 3, 4, 7, 8	1, 3, 4, 7, 8, 9
10	1, 3, 4, 7, 8	1, 3, 4, 7, 8, 10



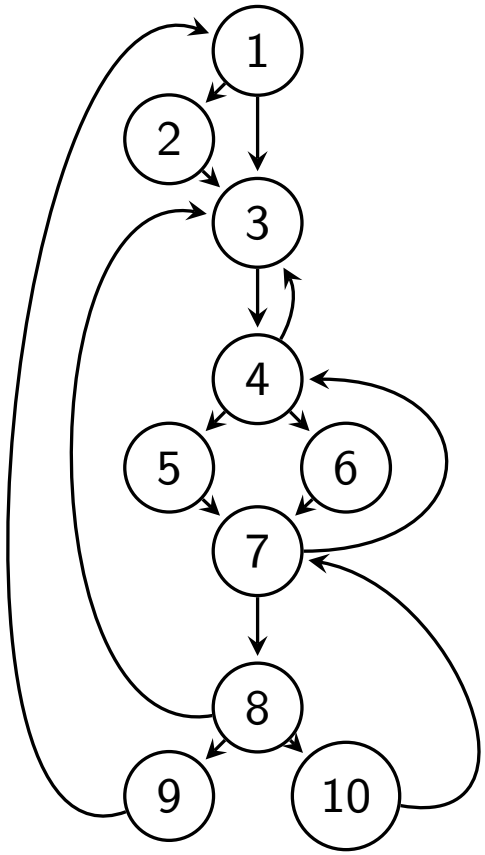
$$OUT[10] = \{10\} \cup OUT[8]$$

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de dominadores

No cambian en la segunda iteración

Bloque	*IN*	*OUT*
1		1
2	1	1, 2
3	1	1, 3
4	1, 3	1, 3, 4
5	1, 3, 4	1, 3, 4, 5
6	1, 3, 4	1, 3, 4, 6
7	1, 3, 4	1, 3, 4, 7
8	1, 3, 4, 7	1, 3, 4, 7, 8
9	1, 3, 4, 7, 8	1, 3, 4, 7, 8, 9
10	1, 3, 4, 7, 8	1, 3, 4, 7, 8, 10



$OUT[i]$ contiene los dominadores de i

UNIVERSIDAD SIMÓN BOLÍVAR

Recorridos

- *Depth-First Ordering*
  - Reverso de un recorrido *post-order*.
  - Se visita un nodo, y luego se *recorren* los hijos de *derecha* a *izquierda*.
- El algoritmo permite construir un **Arbol de Expansión** (*DFST*)

UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo del DFST

Inicialización

input: un grafo de flujo $G = (V, N)$
output: $DFST(G)$ con un orden de los nodos de G

$T \leftarrow \emptyset$
$c \leftarrow |N|$
for all $i \in 1 \ldots c$ **do**
  $dfn[i] \leftarrow 0$
end for
for all $n \in N$ **do**
  $n.visited \leftarrow$ **false**
end for
search(n_0)

- T – contendrá el conjunto de aristas del $DFST(G)$
- $dfn[i]$ – contendrá el orden para el $i-$ésimo nodo.

UNIVERSIDAD SIMÓN BOLÍVAR

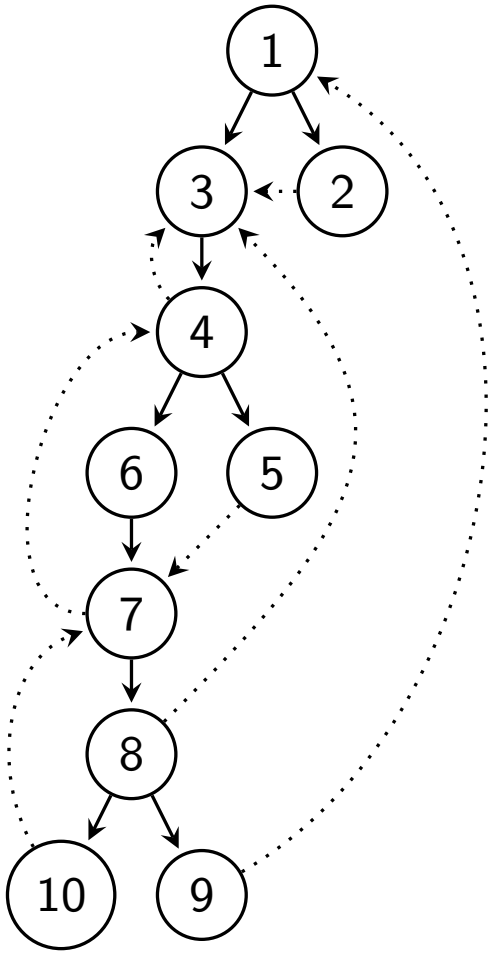
Cálculo del DFST

Recorrido y construcción

sub search(n)
  $n.visited \leftarrow$ **true**
  for all s sucesor de n **do**
    if not $s.visited$ **then**
      $T \leftarrow T \cup (n \rightarrow s)$
      search(s)
    end if
  end for
  $dfn[n] \leftarrow c$
  $c \leftarrow c - 1$

UNIVERSIDAD SIMÓN BOLÍVAR

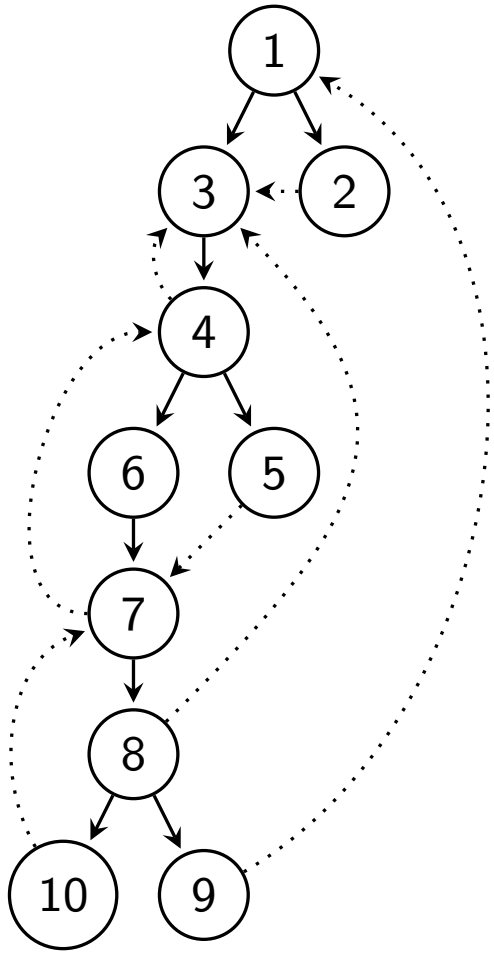
Cálculo de un DFST



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

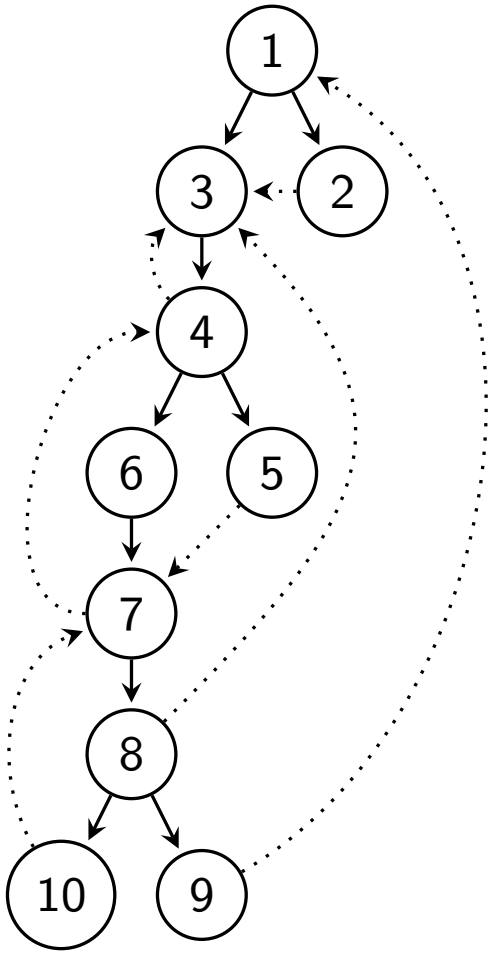
- search(1)
  - Escoger 3
  - Agregar $1 \rightarrow 3$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

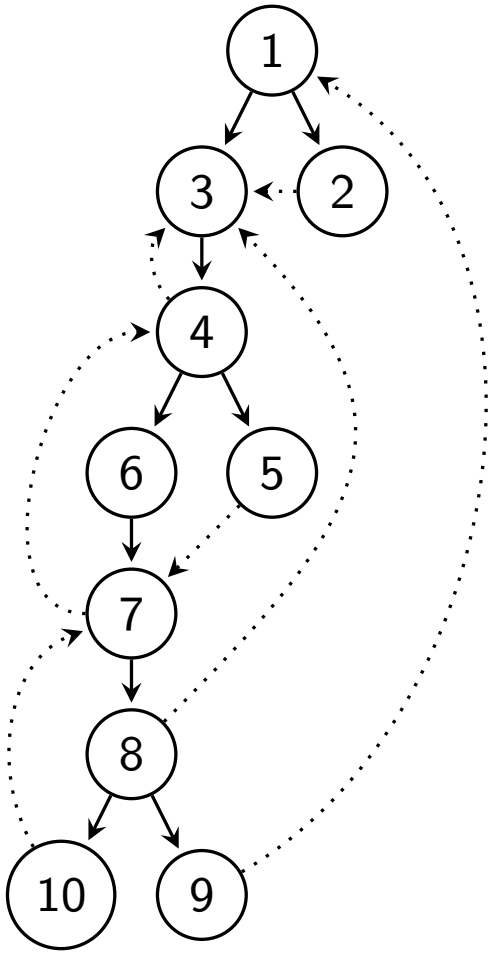
- search(1)
  - Escoger 3
  - Agregar $1 \rightarrow 3$
- search(3) – agregar $3 \rightarrow 4$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

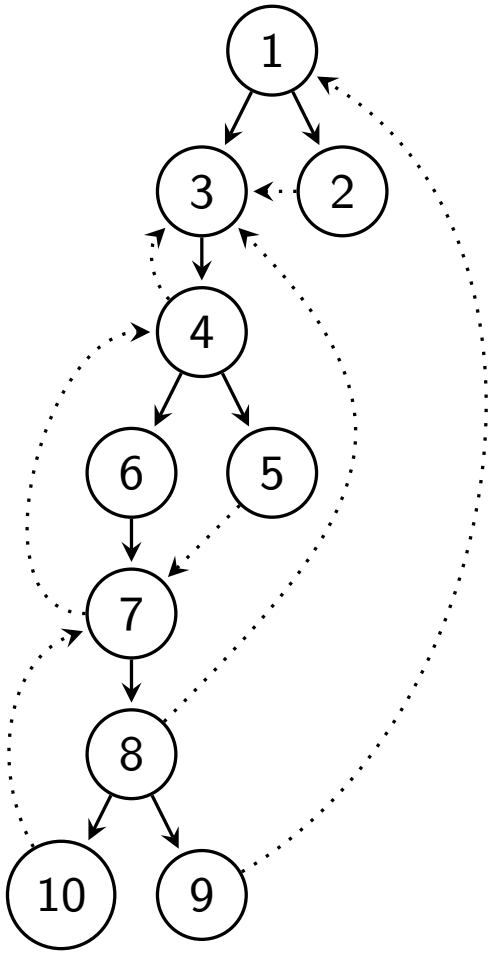
- search(1)
  - Escoger 3
  - Agregar $1 \rightarrow 3$
- search(3) – agregar $3 \rightarrow 4$
- search(4)
  - Escoger 6
  - Agregar $4 \rightarrow 6$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

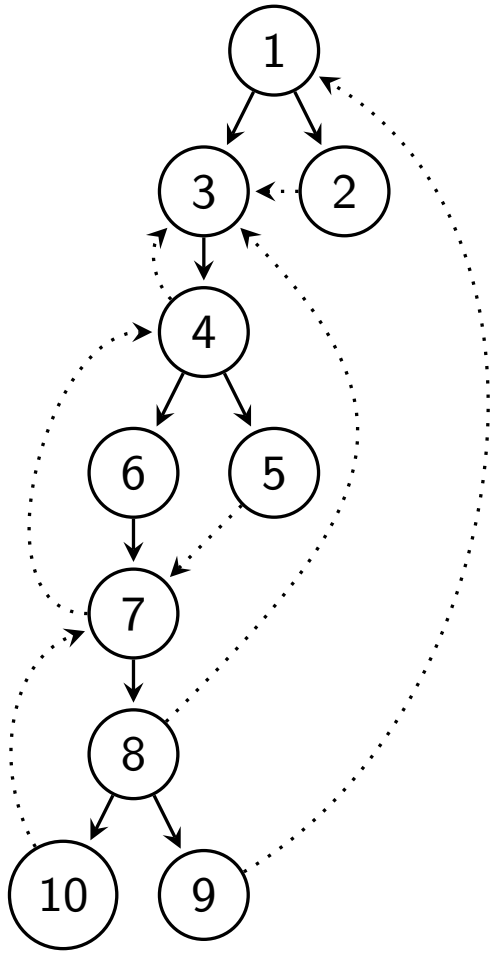
- search(1)
  - Escoger 3
  - Agregar $1 \rightarrow 3$
- search(3) – agregar $3 \rightarrow 4$
- search(4)
  - Escoger 6
  - Agregar $4 \rightarrow 6$
- search(6) – agregar $6 \rightarrow 7$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

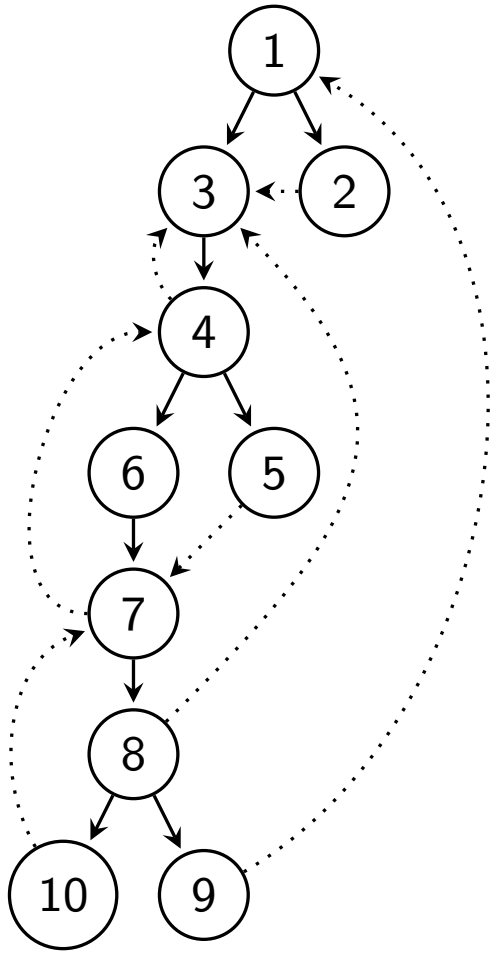
- search(1)
  - Escoger 3
  - Agregar $1 \rightarrow 3$
- search(3) – agregar $3 \rightarrow 4$
- search(4)
  - Escoger 6
  - Agregar $4 \rightarrow 6$
- search(6) – agregar $6 \rightarrow 7$
- search(7)
  - 4 fue visitado, sólo queda 8.
  - Agregar $7 \rightarrow 8$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

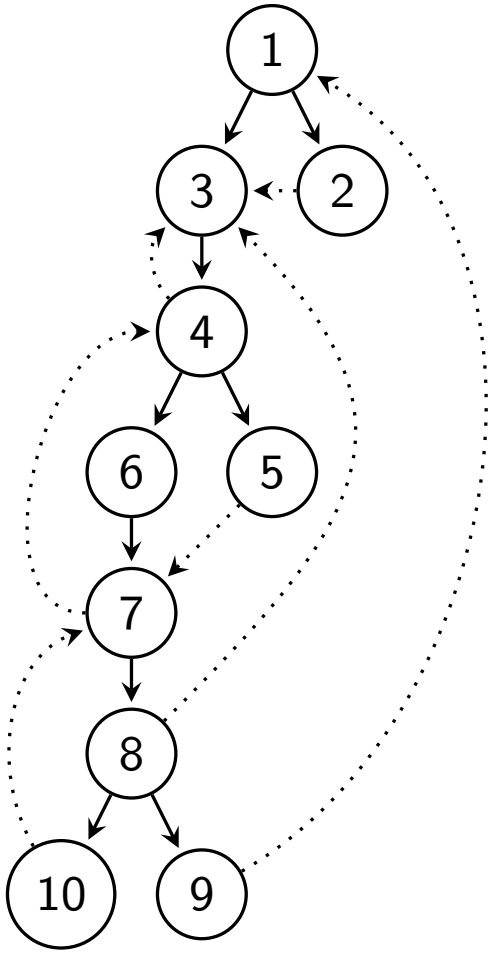
- search(1)
  - Escoger 3
  - Agregar $1 \rightarrow 3$
- search(3) – agregar $3 \rightarrow 4$
- search(4)
  - Escoger 6
  - Agregar $4 \rightarrow 6$
- search(6) – agregar $6 \rightarrow 7$
- search(7)
  - 4 fue visitado, sólo queda 8.
  - Agregar $7 \rightarrow 8$
- search(8)
  - Escoger 10
  - Agregar $8 \rightarrow 10$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

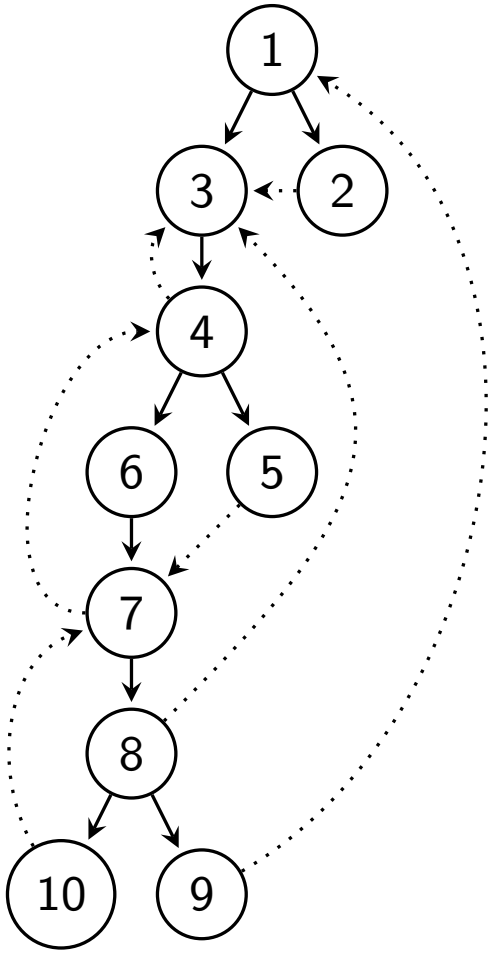
- search(10)
  - 7 fue visitado, no quedan sucesores.
  - $dfn[10] \leftarrow 10, c \leftarrow 9$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

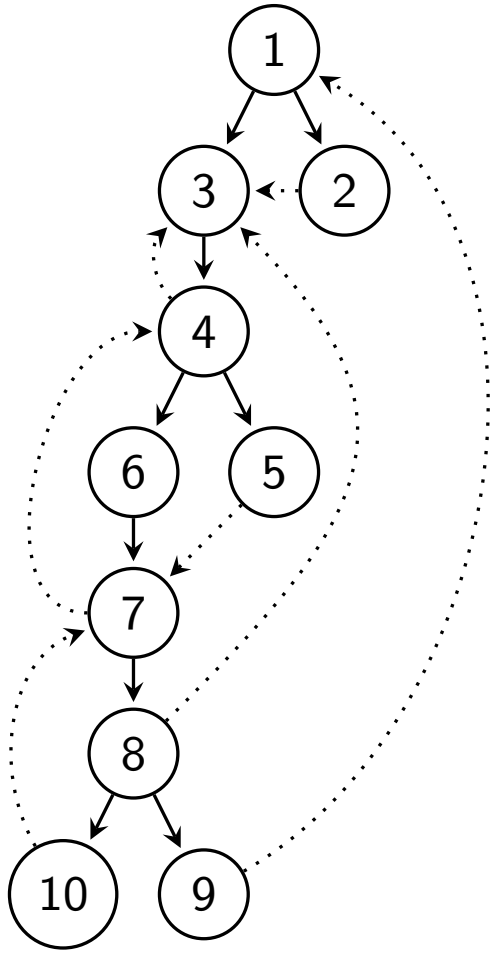
- search(10)
  - 7 fue visitado, no quedan sucesores.
  - $dfn[10] \leftarrow 10, c \leftarrow 9$
- Regresa a search(8)
  - 10 fue visitado, sólo queda 9.
  - Agregar $8 \rightarrow 9$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

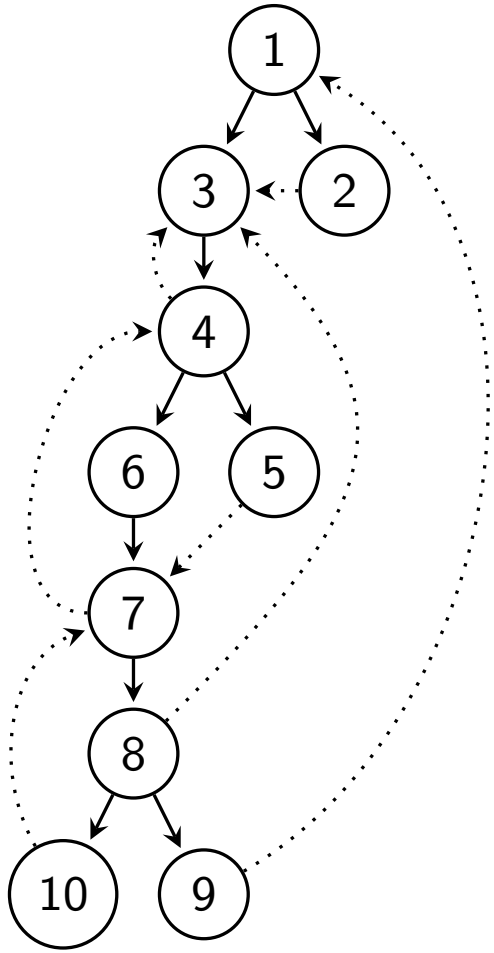
- search(10)
  - 7 fue visitado, no quedan sucesores.
  - $dfn[10] \leftarrow 10, c \leftarrow 9$
- Regresa a search(8)
  - 10 fue visitado, sólo queda 9.
  - Agregar $8 \rightarrow 9$
- search(9)
  - 1 fue visitado, no quedan sucesores.
  - $dfn[9] \leftarrow 9, c \leftarrow 8$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

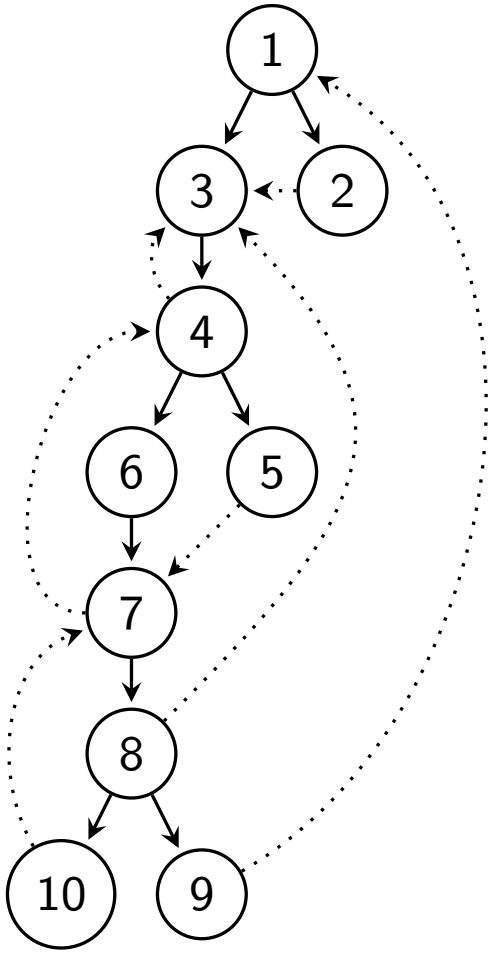
- search(10)
  - 7 fue visitado, no quedan sucesores.
  - $dfn[10] \leftarrow 10, c \leftarrow 9$
- Regresa a search(8)
  - 10 fue visitado, sólo queda 9.
  - Agregar $8 \rightarrow 9$
- search(9)
  - 1 fue visitado, no quedan sucesores.
  - $dfn[9] \leftarrow 9, c \leftarrow 8$
- Regresa a search(8)
  - 10, 9 y 3 fueron visitados, no quedan sucesores.
  - $dfn[8] \leftarrow 8, c \leftarrow 7$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

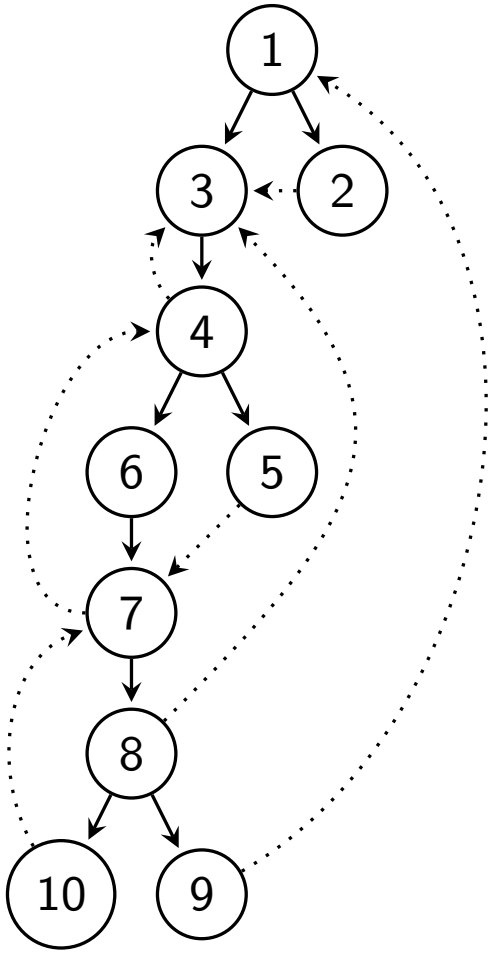
- Regresa a search(7)
  - No quedan sucesores.
  - $dfn[7] \leftarrow 7, c \leftarrow 6$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

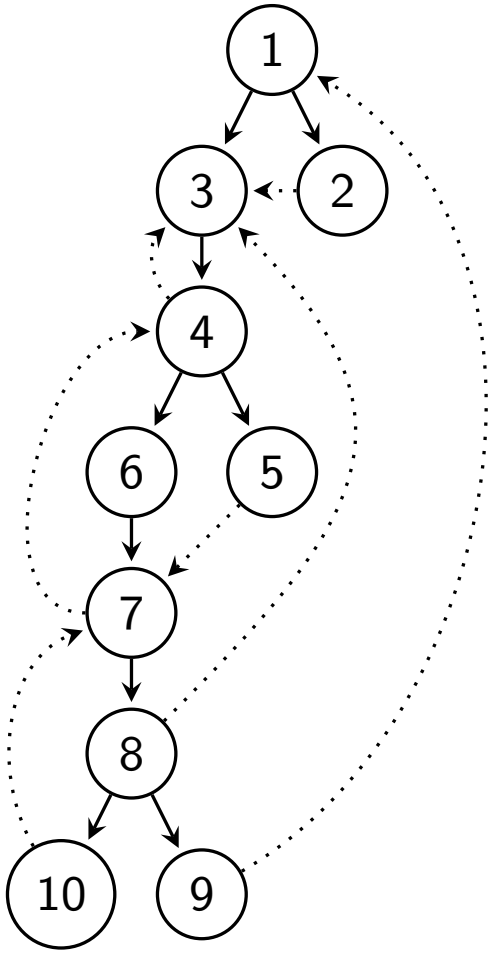
- Regresa a search(7)
  - No quedan sucesores.
  - $dfn[7] \leftarrow 7, c \leftarrow 6$
- Regresa a search(6)
  - No quedan sucesores.
  - $dfn[6] \leftarrow 6, c \leftarrow 5$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

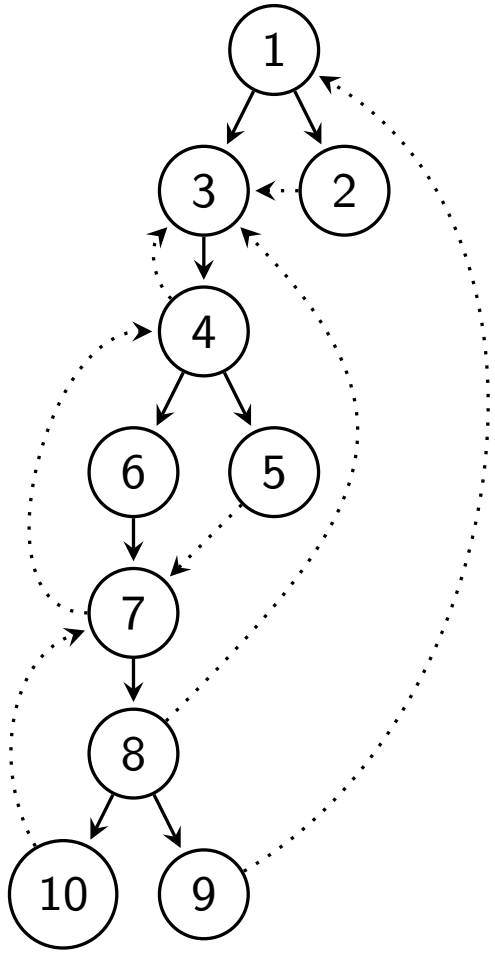
- Regresa a search(7)
  - No quedan sucesores.
  - $dfn[7] \leftarrow 7, c \leftarrow 6$
- Regresa a search(6)
  - No quedan sucesores.
  - $dfn[6] \leftarrow 6, c \leftarrow 5$
- Regresa a search(4)
  - 6 y 3 fueron visitados, queda 5.
  - Agregar $4 \rightarrow 5$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

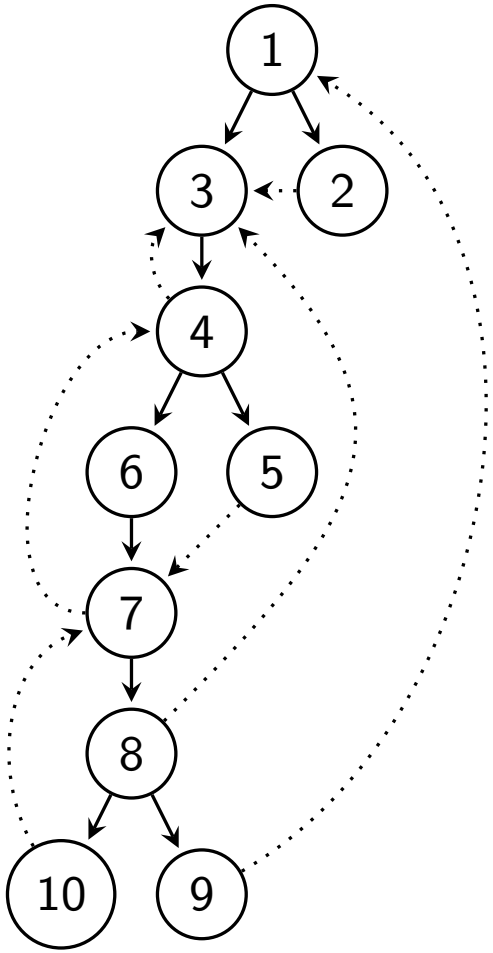
- Regresa a search(7)
  - No quedan sucesores.
  - $dfn[7] \leftarrow 7, c \leftarrow 6$
- Regresa a search(6)
  - No quedan sucesores.
  - $dfn[6] \leftarrow 6, c \leftarrow 5$
- Regresa a search(4)
  - 6 y 3 fueron visitados, queda 5.
  - Agregar $4 \rightarrow 5$
- search(5)
  - 7 fue visitados, no quedan sucesores.
  - $dfn[5] \leftarrow 5, c \leftarrow 4$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

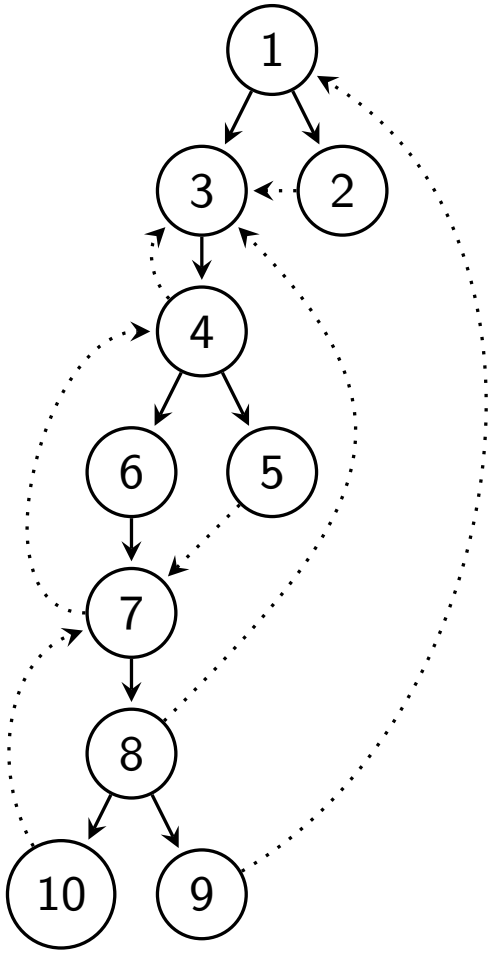
- Regresa a search(4)
  - 3, 5 y 6 fueron visitados, no quedan sucesores.
  - $dfn[4] \leftarrow 4, c \leftarrow 3$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

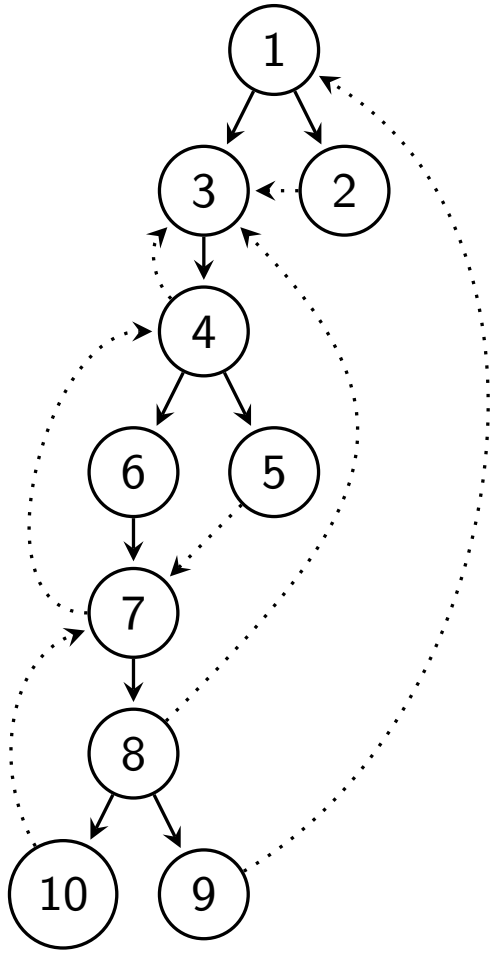
- Regresa a search(4)
  - 3, 5 y 6 fueron visitados, no quedan sucesores.
  - $dfn[4] \leftarrow 4, c \leftarrow 3$
- Regresa a search(3)
  - No quedan sucesores.
  - $dfn[3] \leftarrow 3, c \leftarrow 2$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

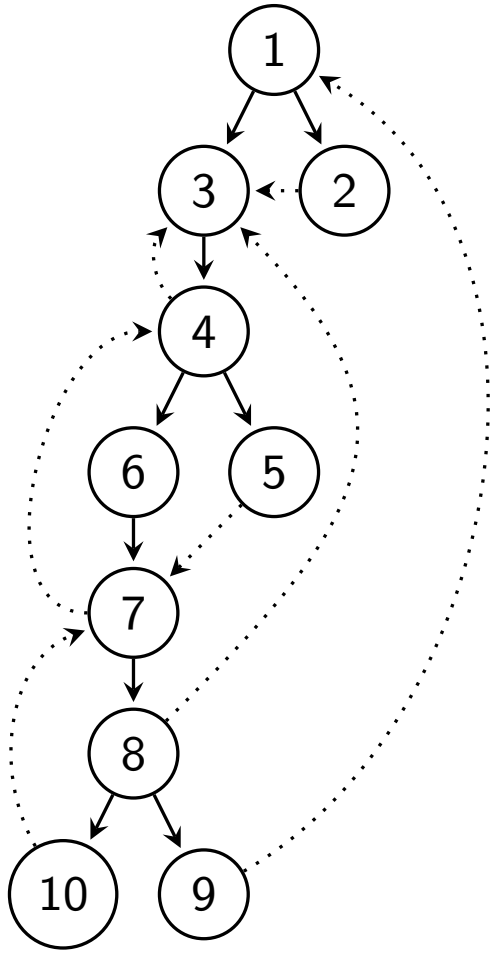
- Regresa a search(4)
  - 3, 5 y 6 fueron visitados, no quedan sucesores.
  - $dfn[4] \leftarrow 4, c \leftarrow 3$
- Regresa a search(3)
  - No quedan sucesores.
  - $dfn[3] \leftarrow 3, c \leftarrow 2$
- Regresa a search(1)
  - 3 fue visitado, queda 2.
  - Agregar $1 \rightarrow 2$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

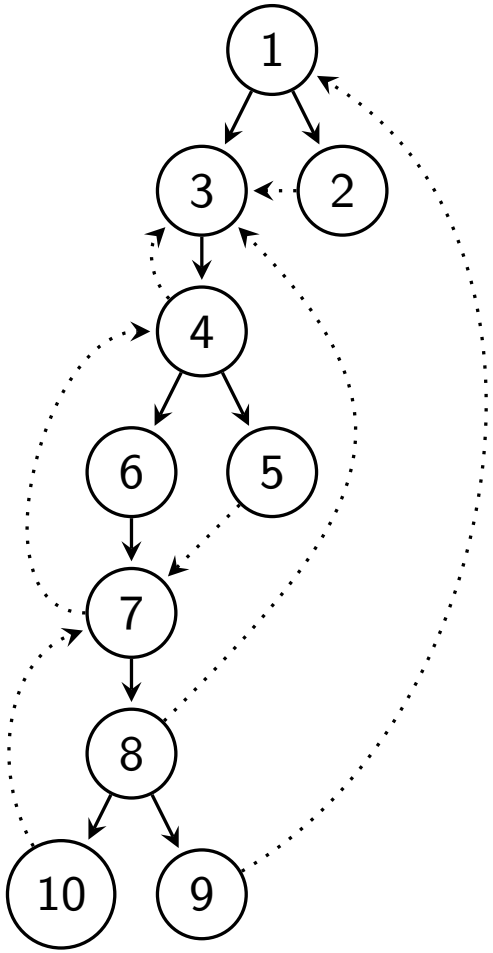
- Regresa a search(4)
  - 3, 5 y 6 fueron visitados, no quedan sucesores.
  - $dfn[4] \leftarrow 4, c \leftarrow 3$
- Regresa a search(3)
  - No quedan sucesores.
  - $dfn[3] \leftarrow 3, c \leftarrow 2$
- Regresa a search(1)
  - 3 fue visitado, queda 2.
  - Agregar $1 \rightarrow 2$
- search(2)
  - 3 ya fue visitado, no quedan sucesores.
  - $dfn[2] \leftarrow 2, c \leftarrow 1$



UNIVERSIDAD SIMÓN BOLÍVAR

Cálculo de un DFST

- Regresa a search(1)
  - 3 y 2 fueron visitados, no quedan sucesores.
  - $dfn[1] \leftarrow 1, c \leftarrow 0$
- Aristas sólidas conforman el DFST.



UNIVERSIDAD SIMÓN BOLÍVAR

Tipos de aristas

- **Aristas de Avance** (*Advancing Edges*)
  - De un nodo m hacia un descendiente propio de m.
  - Todas las aristas de un DSFT son de avance.

Tipos de aristas

- **Aristas de Avance** (*Advancing Edges*)
  - De un nodo m hacia un descendiente propio de m.
  - Todas las aristas de un DSFT son de avance.
- **Aristas de Retroceso** (*Retreating Edges*)
  - De un nodo m hacia un ancestro de m – posiblemente el propio m.

UNIVERSIDAD SIMÓN BOLÍVAR

Tipos de aristas

- **Aristas de Avance** (*Advancing Edges*)
  - De un nodo m hacia un descendiente propio de m.
  - Todas las aristas de un DSFT son de avance.
- **Aristas de Retroceso** (*Retreating Edges*)
  - De un nodo m hacia un ancestro de m – posiblemente el propio m.
- **Aristas de Cruce** (*Cross Edges*)
  - Aristas $m \rightarrow n$ tales que ninguno es ancestro del otro.
  - Al graficar el árbol de manera que los hijos de un nodo se presenten de *izquierda* a *derecha* según el orden en que fueron agregados al árbol, estas aristas irán de *derecha* a *izquierda*.
  - $m \rightarrow n$ es arista de cruce, si y sólo si $dfn[m] \geq dfn[n]$

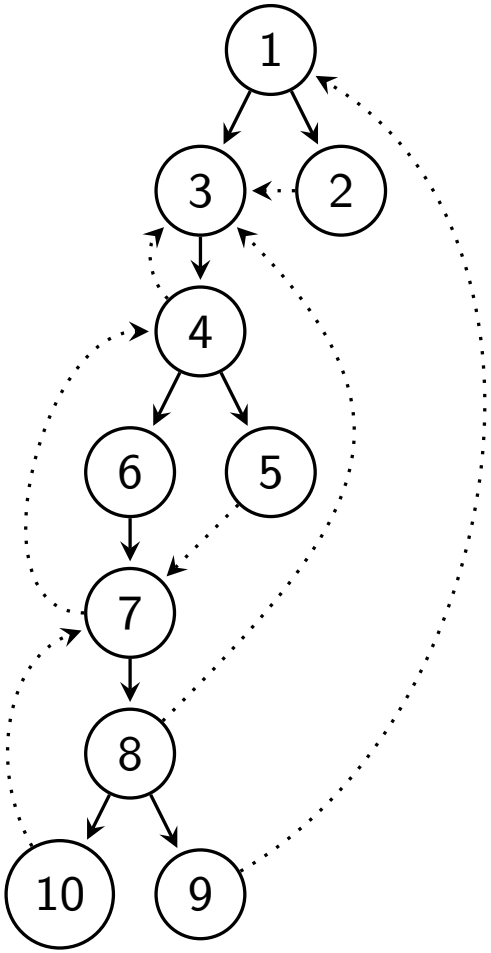
UNIVERSIDAD SIMÓN BOLÍVAR

Tipos de aristas

- **Aristas de Avance** (*Advancing Edges*)
  - De un nodo m hacia un descendiente propio de m.
  - Todas las aristas de un DSFT son de avance.
- **Aristas de Retroceso** (*Retreating Edges*)
  - De un nodo m hacia un ancestro de m – posiblemente el propio m.
- **Aristas de Cruce** (*Cross Edges*)
  - Aristas $m \rightarrow n$ tales que ninguno es ancestro del otro.
  - Al graficar el árbol de manera que los hijos de un nodo se presenten de *izquierda* a *derecha* según el orden en que fueron agregados al árbol, estas aristas irán de *derecha* a *izquierda*.
  - $m \rightarrow n$ es arista de cruce, si y sólo si $dfn[m] \geq dfn[n]$
- **Aristas de Retorno** (*Back Edges*)
  - $a \rightarrow b$ tal que b *dom* a.

UNIVERSIDAD SIMÓN BOLÍVAR

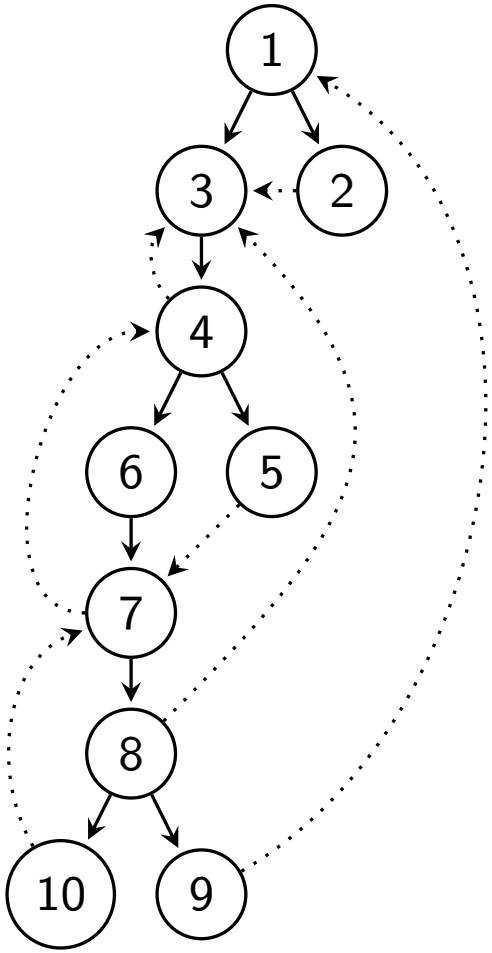
Tipos de Aristas



- Aristas de Avance – todas las del DFST.

UNIVERSIDAD SIMÓN BOLÍVAR

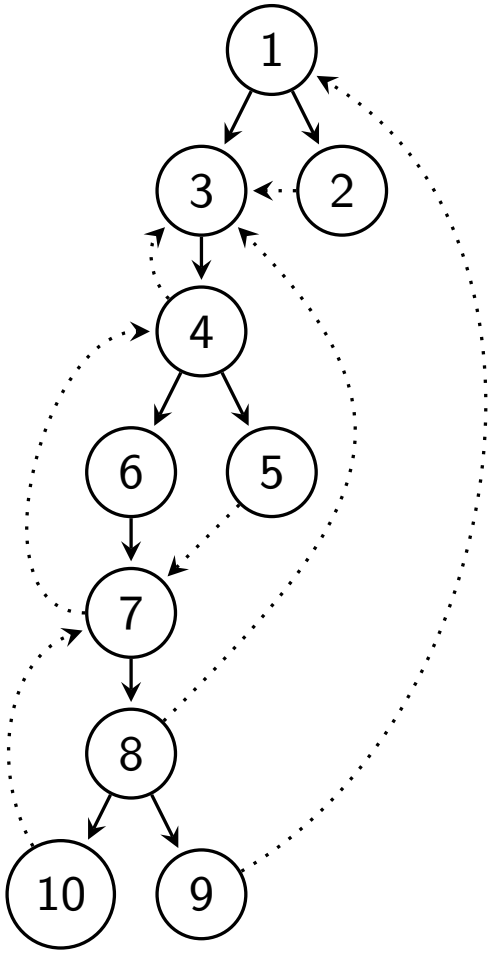
Tipos de Aristas



- Aristas de Avance – todas las del DFST.
- Aristas de Retroceso
  - $4 \rightarrow 3$
  - $7 \rightarrow 4$
  - $10 \rightarrow 7$
  - $8 \rightarrow 3$
  - $9 \rightarrow 1$.

UNIVERSIDAD SIMÓN BOLÍVAR

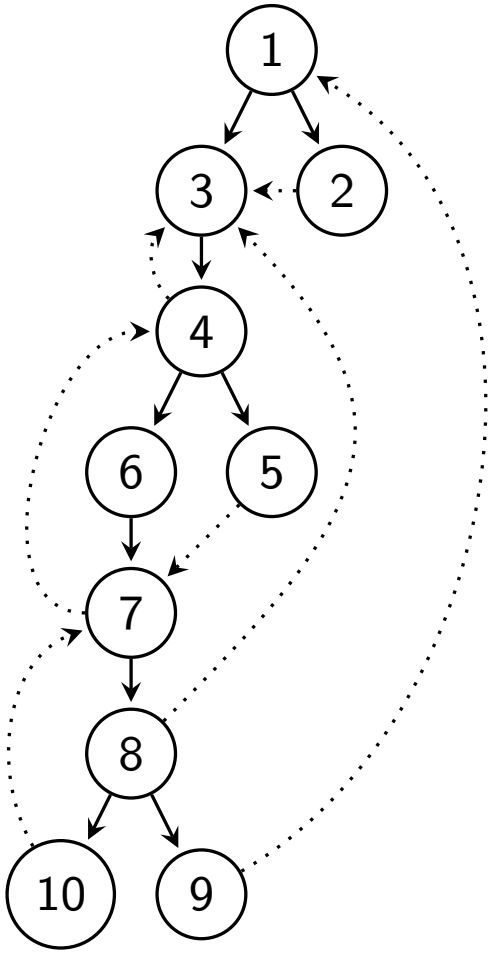
Tipos de Aristas



- Aristas de Avance – todas las del DFST.
- Aristas de Retroceso
  - $4 \rightarrow 3$
  - $7 \rightarrow 4$
  - $10 \rightarrow 7$
  - $8 \rightarrow 3$
  - $9 \rightarrow 1$.
- Aristas de Cruce
  - $2 \rightarrow 3$
  - $5 \rightarrow 7$.

UNIVERSIDAD SIMÓN BOLÍVAR

Tipos de Aristas



- Aristas de Avance – todas las del DFST.
- Aristas de Retroceso
  - $4 \rightarrow 3$
  - $7 \rightarrow 4$
  - $10 \rightarrow 7$
  - $8 \rightarrow 3$
  - $9 \rightarrow 1$.
- Aristas de Cruce
  - $2 \rightarrow 3$
  - $5 \rightarrow 7$.
- Aristas de Retorno – todas las de Retroceso.

UNIVERSIDAD SIMÓN BOLÍVAR

Retroceso vs. Retorno

- Para cualquier Grafo de Flujo
  - Toda arista de retorno también es de retroceso.
  - No toda arista de retroceso es de retorno.

Retroceso vs. Retorno

- Para cualquier Grafo de Flujo
  - Toda arista de retorno también es de retroceso.
  - No toda arista de retroceso es de retorno.
- El Grafo de Flujo es **reducible** si *todas* las aristas de retroceso también son de retorno, para *cualquier* DFST.
  - Ser reducible implica que todos los DFST tienen el mismo conjunto de aristas de *retroceso*, que son exactamente todas las aristas de *retorno*.
  - Basta eliminar las aristas de *retorno* en un DFST – ¡si quedan ciclos, no es reducible!

UNIVERSIDAD SIMÓN BOLÍVAR

Retroceso vs. Retorno

- Para cualquier Grafo de Flujo
  - Toda arista de retorno también es de retroceso.
  - No toda arista de retroceso es de retorno.
- El Grafo de Flujo es **reducible** si *todas* las aristas de retroceso también son de retorno, para *cualquier* DFST.
  - Ser reducible implica que todos los DFST tienen el mismo conjunto de aristas de *retroceso*, que son exactamente todas las aristas de *retorno*.
  - Basta eliminar las aristas de *retorno* en un DFST – ¡si quedan ciclos, no es reducible!
- En la práctica, todos los Grafos de Flujo son reducibles.
  - Las construcciones de programación estructurada, *garantizan* la reducibilidad de sus Grafos de Flujo.
  - Uso de `goto` por adultos responsables también suele ser reducible.

UNIVERSIDAD SIMÓN BOLÍVAR

Profundidad de un Grafo de Flujo

- Sea un DFST cualquiera asociado a un Grafo de Flujo – su **profundidad** es el número máximo de aristas de retroceso en cualquier camino sin ciclos.
- Intuición – nunca es más que el máximo anidamiento.
- Suponiendo que el Grafo de Flujo es reducible, entonces podemos usar el número de aristas de retorno y así la profundidad será independiente del DFST particular.
- En el camino sin ciclos pueden participar aristas de avance, de retroceso o de cruce en cualquier combinación – la profundidad solamente depende de las aristas de retroceso (o retorno).

Nuestro ejemplo tiene profundidad 3 por el camino
$10 \rightarrow 7 \rightarrow 4 \rightarrow 3$

UNIVERSIDAD SIMÓN BOLÍVAR

Detección de Ciclos

- No importa la construcción de alto nivel usada para expresarlo – sólo importa que tengan las propiedades, y así poder mejorarlos.
- Nos interesan los **ciclos naturales**, que cumplen:
  - Nodo único de entrada (*header*) domina al resto de los nodos del ciclo.
  - Debe haber una arista de retorno hacia el *header*.
- El **ciclo natural de una arista** de retorno $n \rightarrow d$ lo conforman d y los nodos que pueden alcanzar n *sin* pasar por d – d es el *header*.

UNIVERSIDAD SIMÓN BOLÍVAR

Algoritmo de Construcción de Ciclos Naturales

Partiendo de una arista de retroceso

input: un grafo de flujo $G = (V, N)$ y arista de retorno $n \rightarrow d$
output: conjunto de nodos para el Ciclo Natural

$loop \leftarrow \{n, d\}$
$d.visited \leftarrow$ **true**
Completar un DFS sobre el Grafo de Control *inverso* partiendo de n
Agregar los nodos visitados a *loop*

UNIVERSIDAD SIMÓN BOLÍVAR

Ciclo dentro de un ciclo dentro de un ciclo...

Sólo nos interesan los ciclos naturales

- Dos ciclos naturales con *header* diferente.
  - Son disjuntos o anidados.
  - ¡Fácil determinar si un ciclo no contiene ciclos!
- Dos ciclos naturales con el mismo *header*
  - Difícil decidir cuál es el interior.
  - Los combinaremos y trabajaremos como un ciclo único.

UNIVERSIDAD SIMÓN BOLÍVAR

Velocidad de Convergencia

- ¿Cómo mejorar el tiempo de un algoritmo de análisis de flujo?
  - Hay una cota superior dada por las condiciones matemáticas del problema general – detalles en el libro.
  - Es posible reducirla a mucho menos que eso – el “cinco”

UNIVERSIDAD SIMÓN BOLÍVAR

Velocidad de Convergencia

- ¿Cómo mejorar el tiempo de un algoritmo de análisis de flujo?
  - Hay una cota superior dada por las condiciones matemáticas del problema general – detalles en el libro.
  - Es posible reducirla a mucho menos que eso – el "cinco"
- ¿Cuán *rápido* llegan los eventos de flujo de datos a un nodo?
  - Lo más rápido es a través de un camino sin ciclos.
  - Y el más rápido es el camino mas corto sin ciclos.

UNIVERSIDAD SIMÓN BOLÍVAR

Velocidad de Convergencia

- ¿Cómo mejorar el tiempo de un algoritmo de análisis de flujo?
  - Hay una cota superior dada por las condiciones matemáticas del problema general – detalles en el libro.
  - Es posible reducirla a mucho menos que eso – el "cinco"
- ¿Cuán *rápido* llegan los eventos de flujo de datos a un nodo?
  - Lo más rápido es a través de un camino sin ciclos.
  - Y el más rápido es el camino mas corto sin ciclos.
- ¿Cómo influyen los ciclos?
  - Nada – variables vivas, expresiones disponibles.
  - Mucho – propagación de constantes.

UNIVERSIDAD SIMÓN BOLÍVAR

Velocidad de Convergencia

Cuando los ciclos no agregan nada...

- El algoritmo para calcular DFST establece que $a \rightarrow b$ una arista de retroceso, si y sólo si $dfn[b] < dfn[a]$
- Si el problema de análisis de flujo es *forward flow*, conviene visitar los nodos según el orden *dfn*
  - El número de pasos necesarios para propagar definiciones sobre caminos acíclicos es a lo sumo uno más el número de aristas de retroceso – ¡uno más la profundidad!
  - Knuth estableció que el promedio de profundidad es 2.75
- Si el problema de análisis de flujo es *backward flow*, conviene visitar los nodos según el orden inverso *dfn*.

UNIVERSIDAD SIMÓN BOLÍVAR

Velocidad de Convergencia

Cuando los ciclos agregan...

- Cálculo de dominadores
  - Si el grafo es reducible, el orden *dfn* permite encontrar los dominadores en *una* iteración.
  - Si el grafo no es reducible, el orden *dfn* permite encontrar los dominadores en *dos* iteraciones.

UNIVERSIDAD SIMÓN BOLÍVAR

Bibliografía

- [*Aho*]
  - Sección 9.6
  - Ejercicios 9.6.1 a 9.6.13

UNIVERSIDAD SIMÓN BOLÍVAR