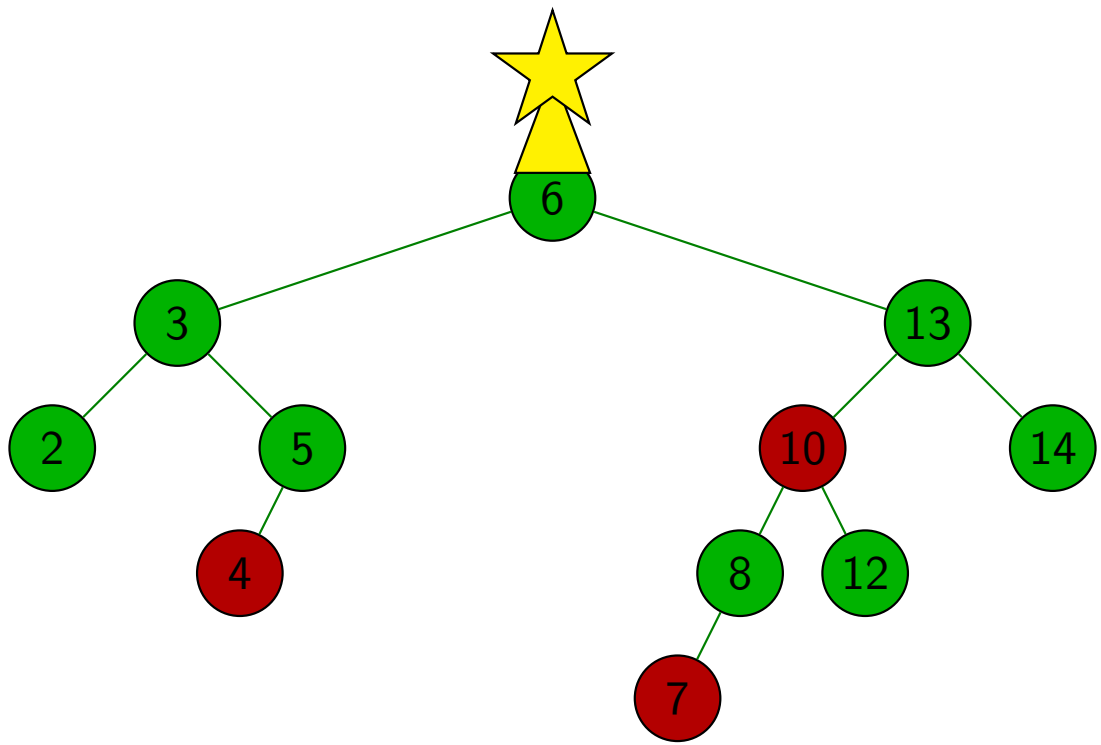
Pure Binary Finger Search Trees

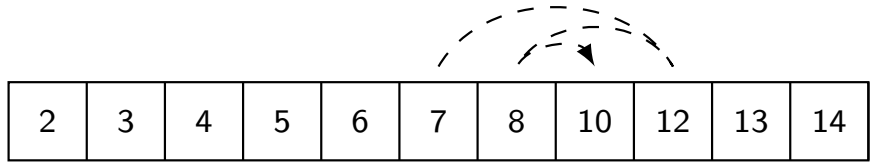


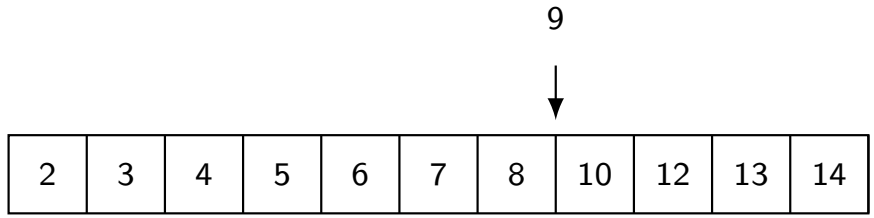
2	3	4	5	6	7	8	10	12	13	14

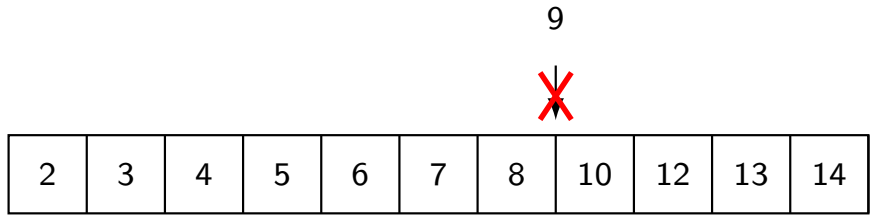
9?

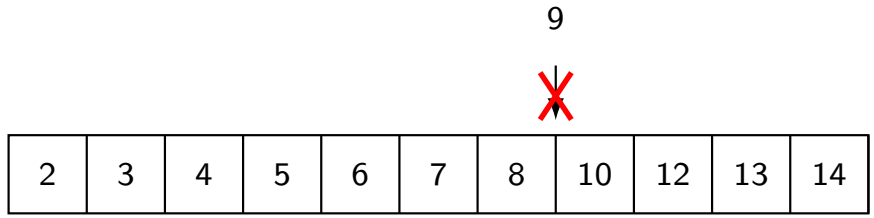
2	3	4	5	6	7	8	10	12	13	14

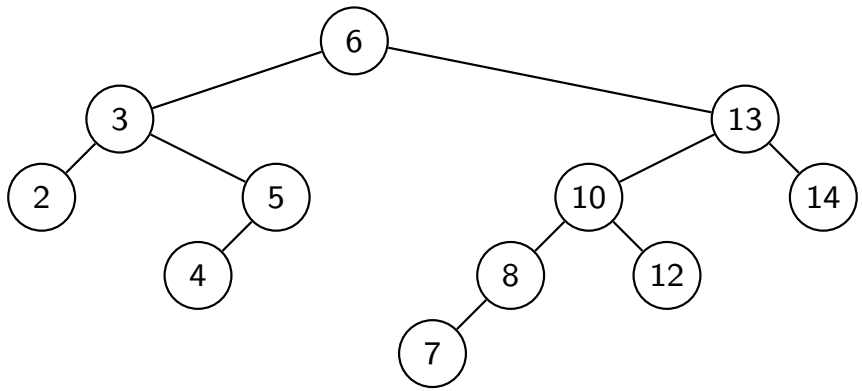
9?

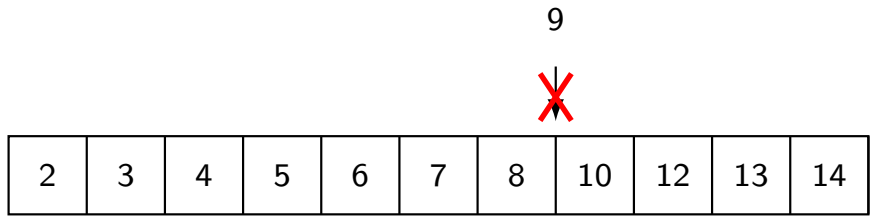


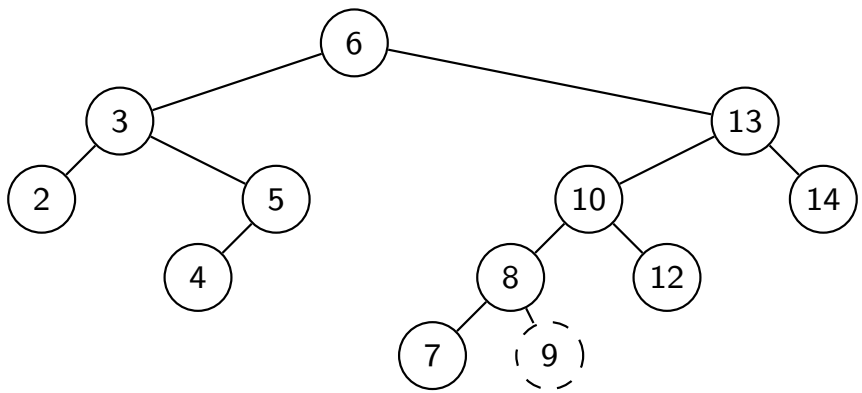

9
2
3
4
5
6
7
8
10
12
13
14


9
2
3
4
5
6
7
8
10
12
13
14


9
2
3
4
5
6
7
8
10
12
13
14

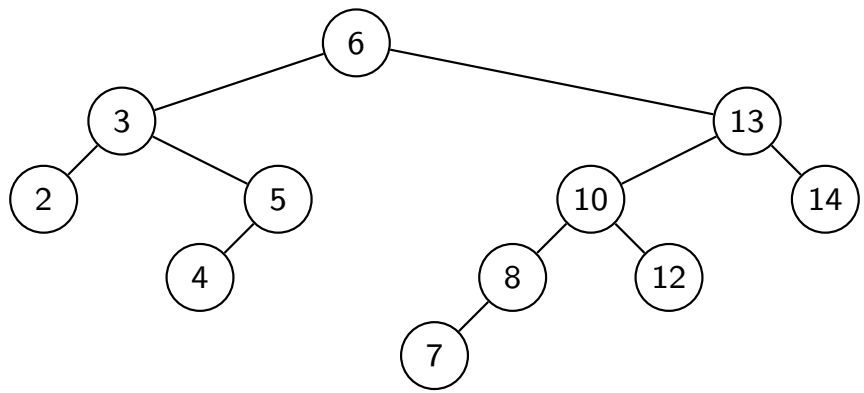

6
3
13
2
5
10
14
4
8
12
7


9
2
3
4
5
6
7
8
10
12
13
14

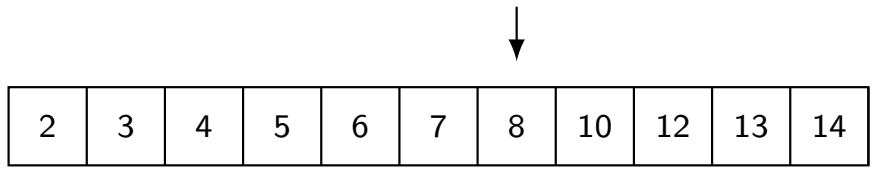

6
3
13
2
5
10
14
4
8
12
7
9

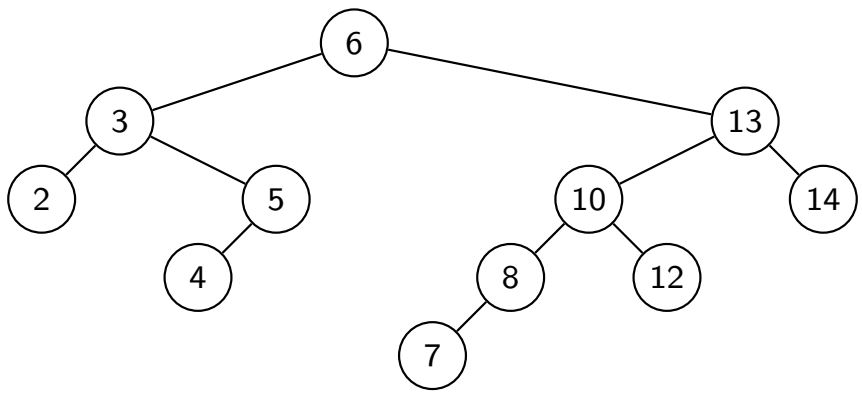
5?

2	3	4	5	6	7	8	10	12	13	14

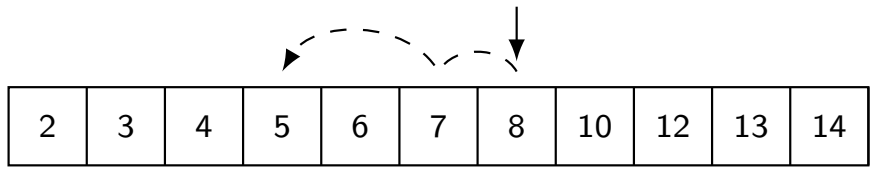


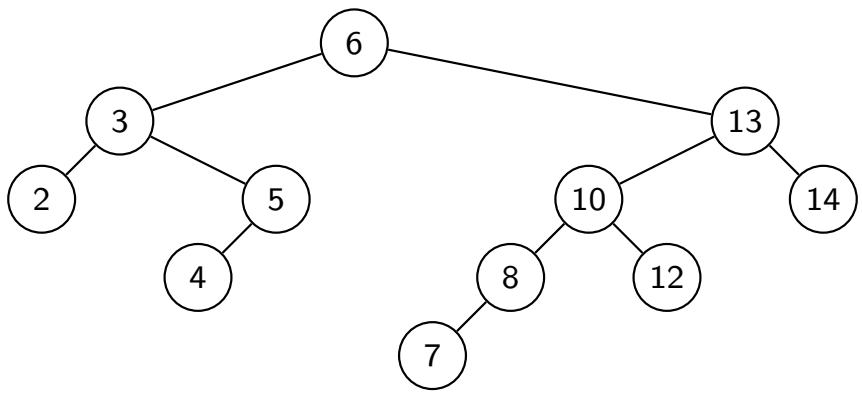
5?

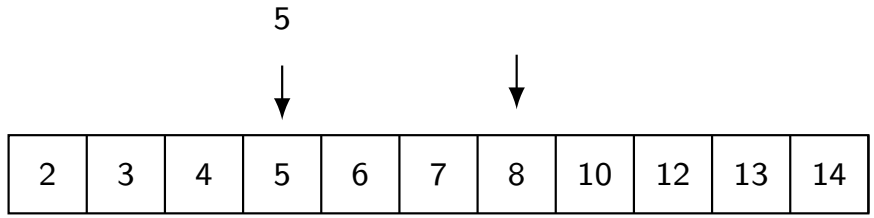


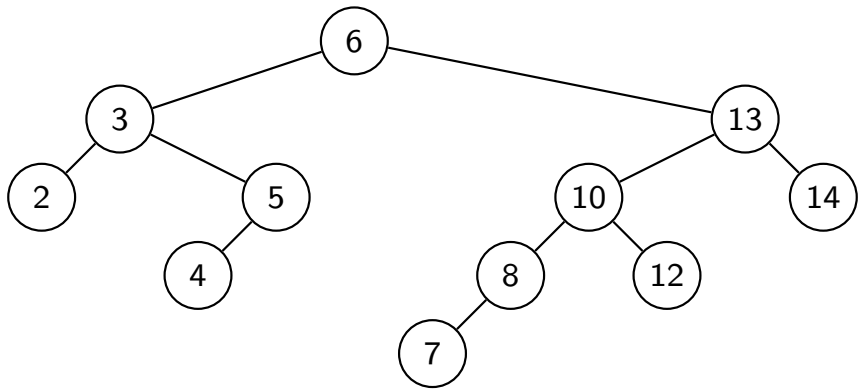


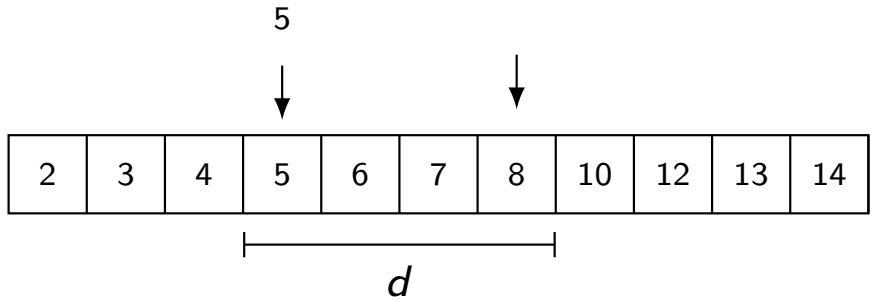
5?

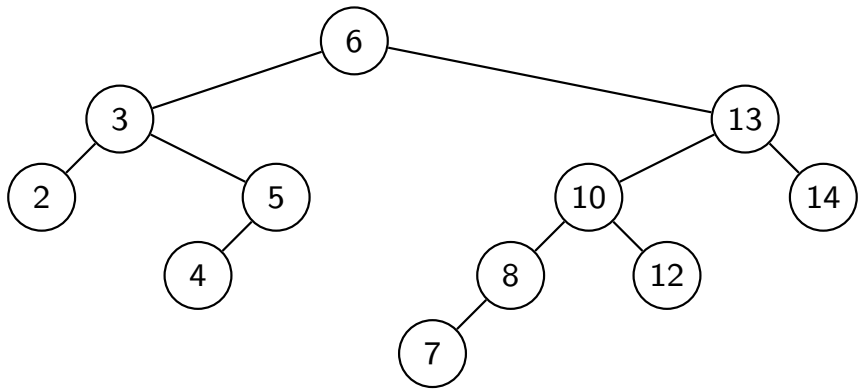


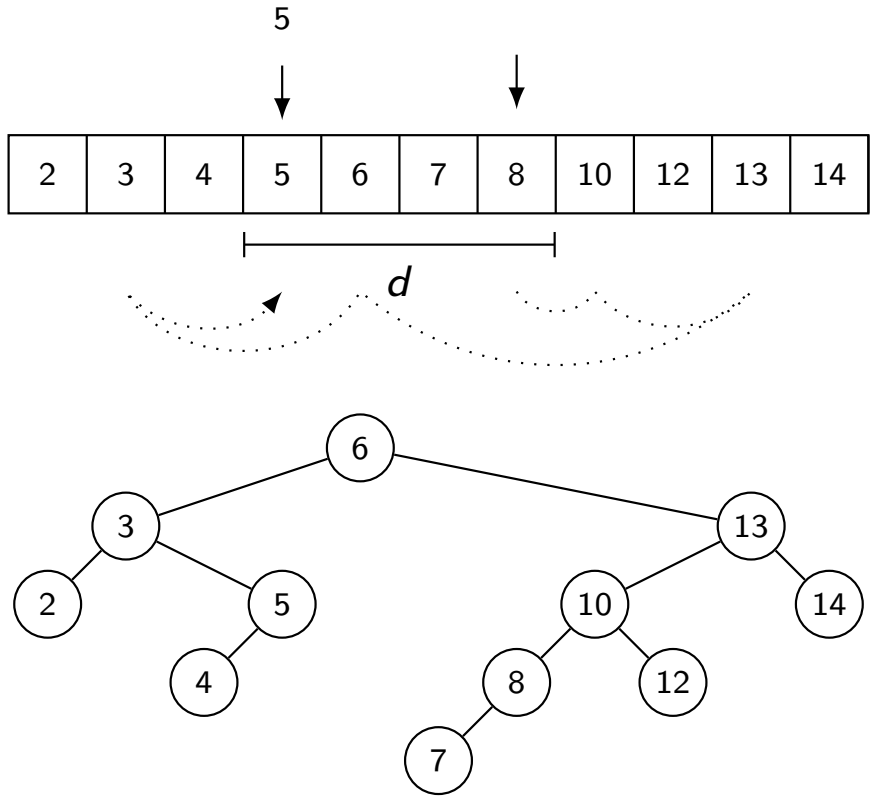



5
2
3
4
5
6
7
8
10
12
13
14


6
3
13
2
5
10
14
4
8
12
7


5
2
3
4
5
6
7
8
10
12
13
14
d

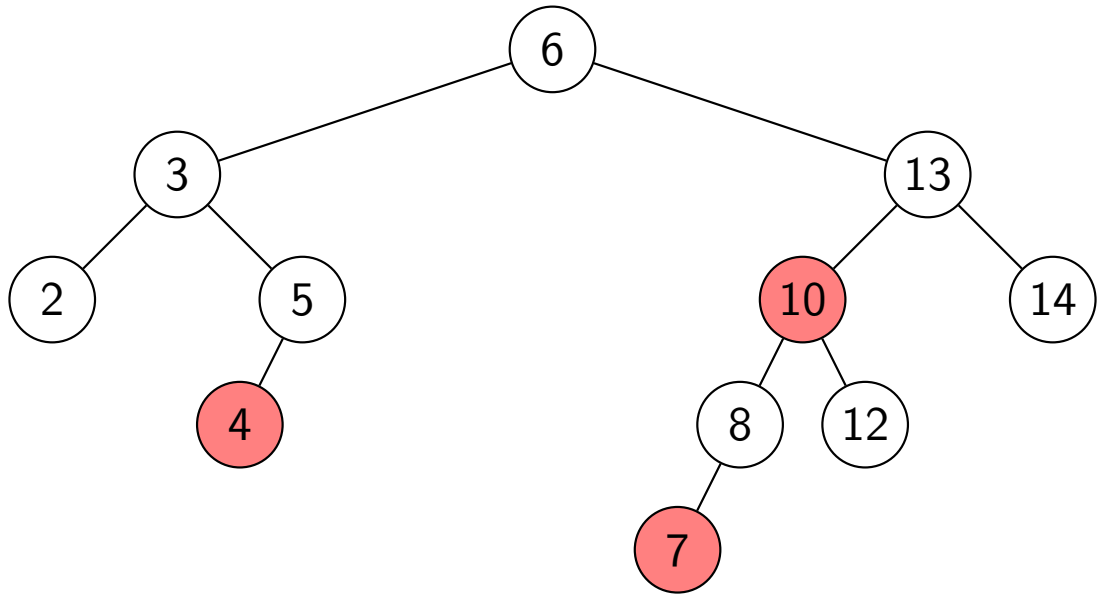

6
3
13
2
5
10
14
4
8
12
7

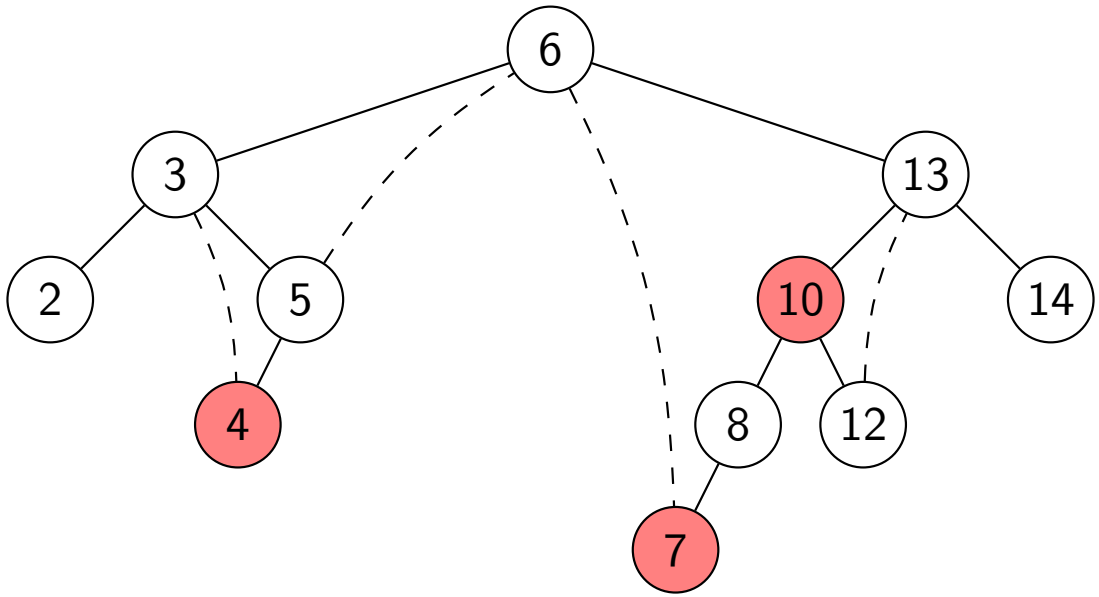

5
2
3
4
5
6
7
8
10
12
13
14
d
6
3
13
2
5
10
14
4
8
12
7

	Finger Search Trees *This paper*
Succ Pred	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$
FingerSearch	$\mathcal{O}(\lg d)$

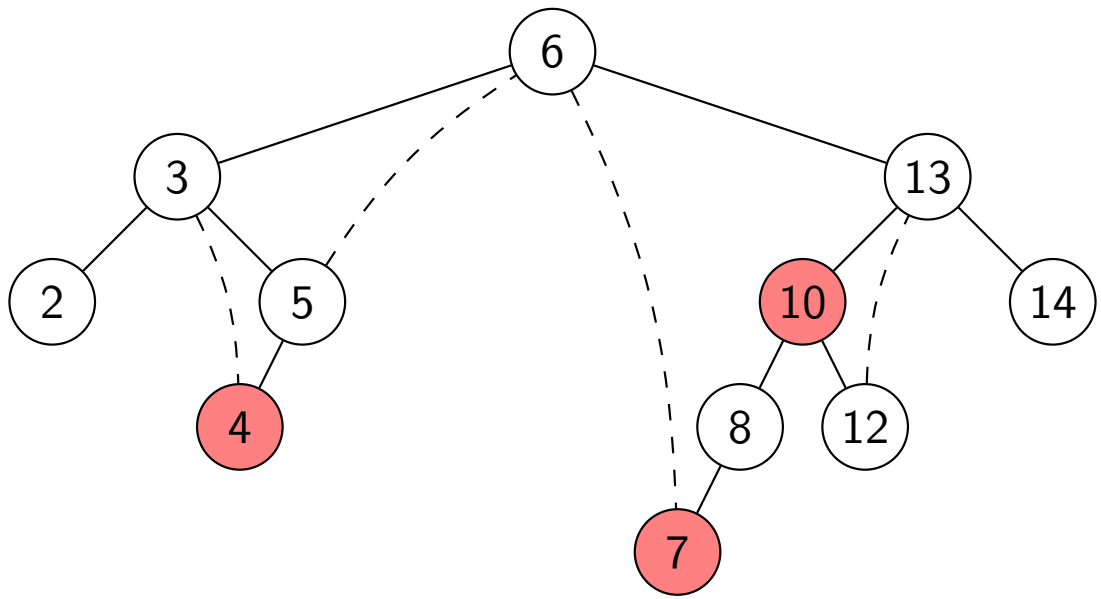
	Finger Search Trees *This paper*	Red-black Trees[1]
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(\lg n)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}(\lg n)$
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg n)$

[1] Guibas and Sedgewick. A dichromatic framework for balanced trees. 1978

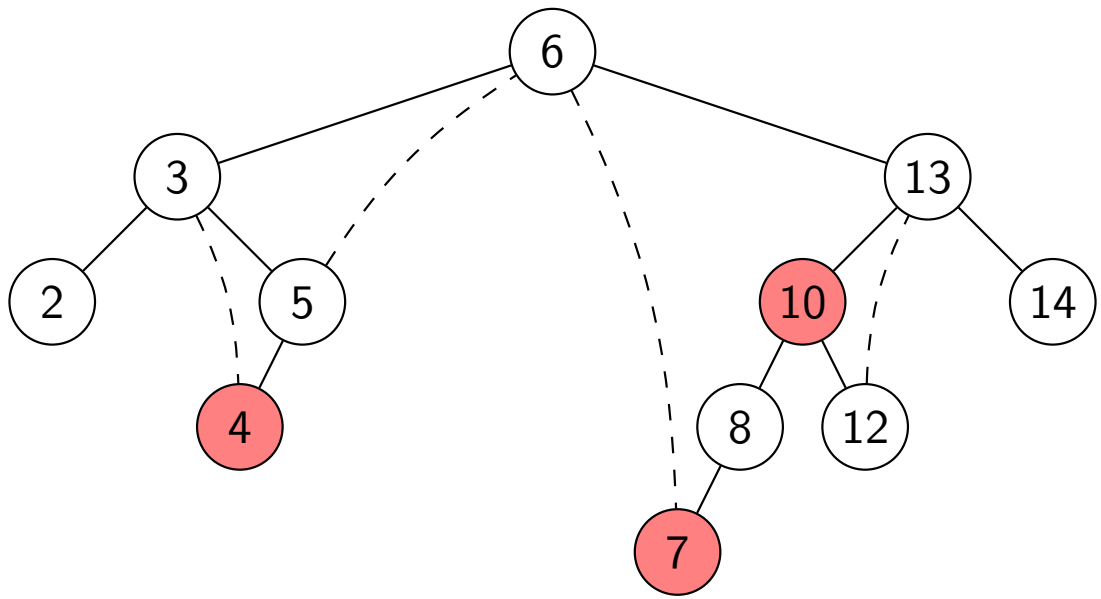

6
3
13
2
5
10
14
4
8
12
7


6
3
13
2
5
10
14
4
8
12
7

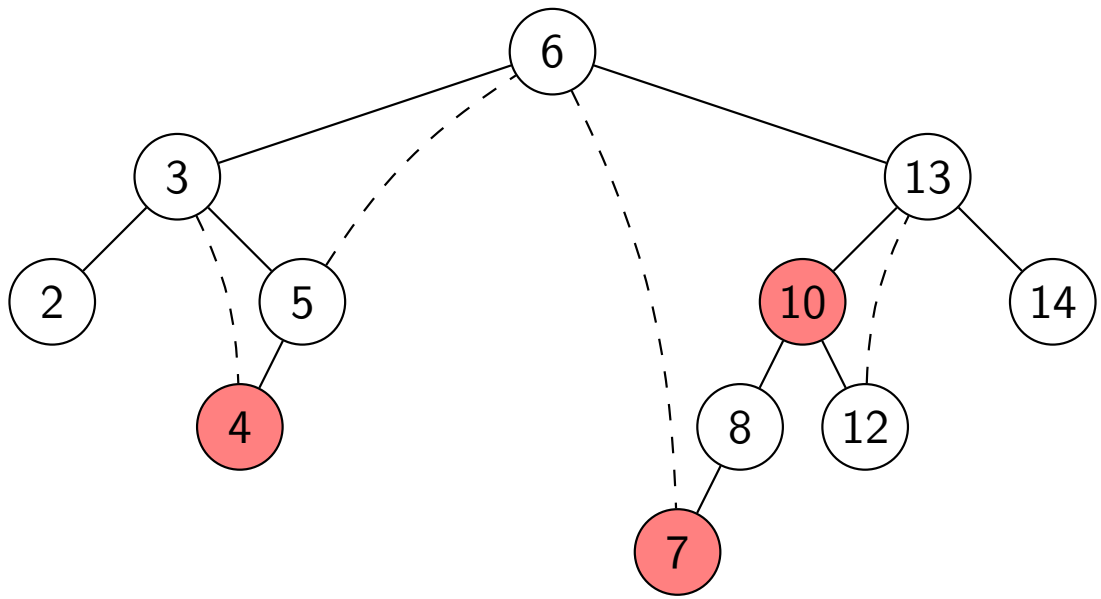
	This paper	Red-black Trees with links
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(\lg n)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}(\lg n)$
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg n)$



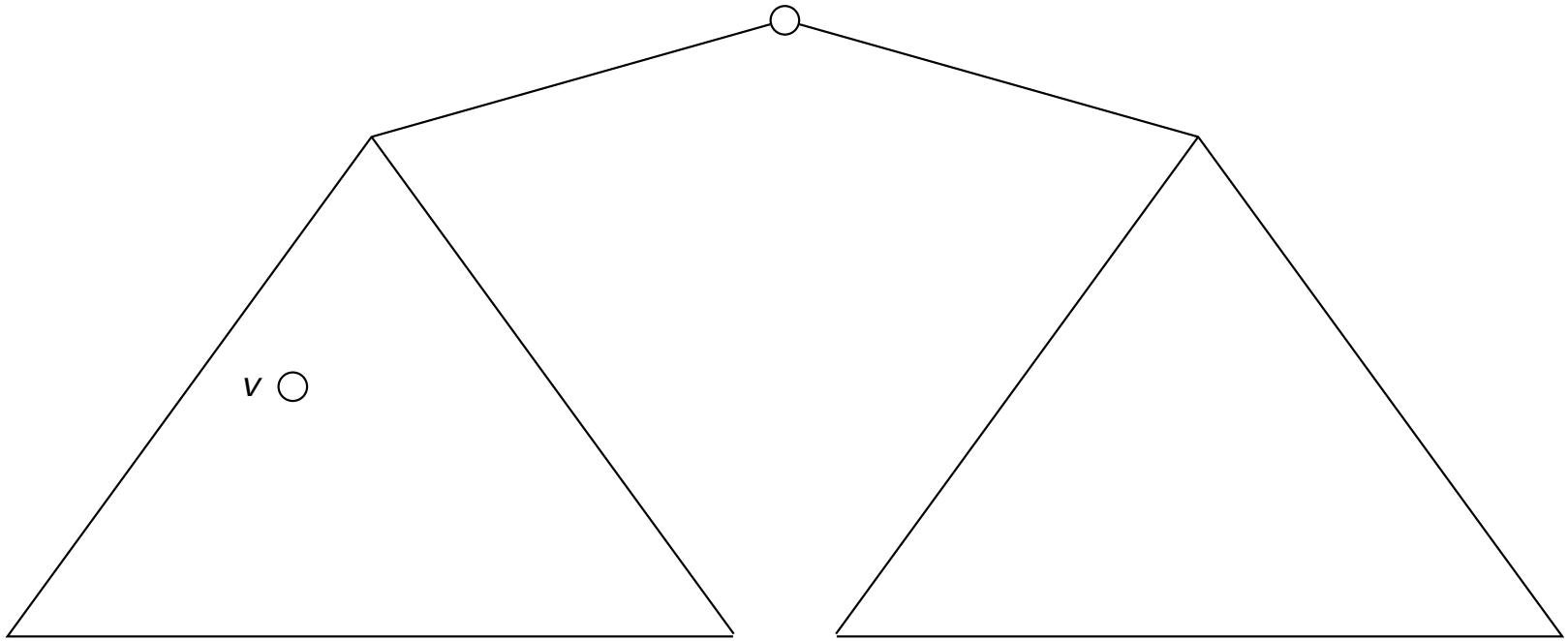
	This paper	Red-black Trees with links
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}(\lg n)$
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg n)$

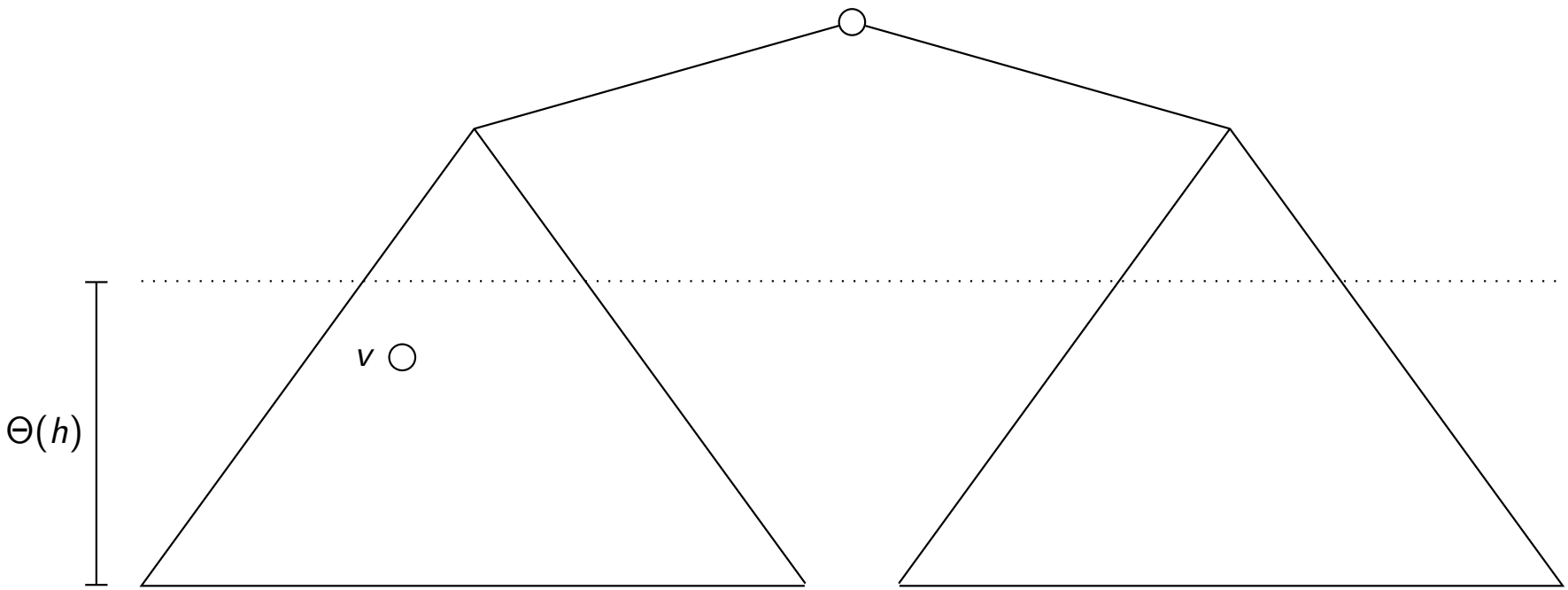


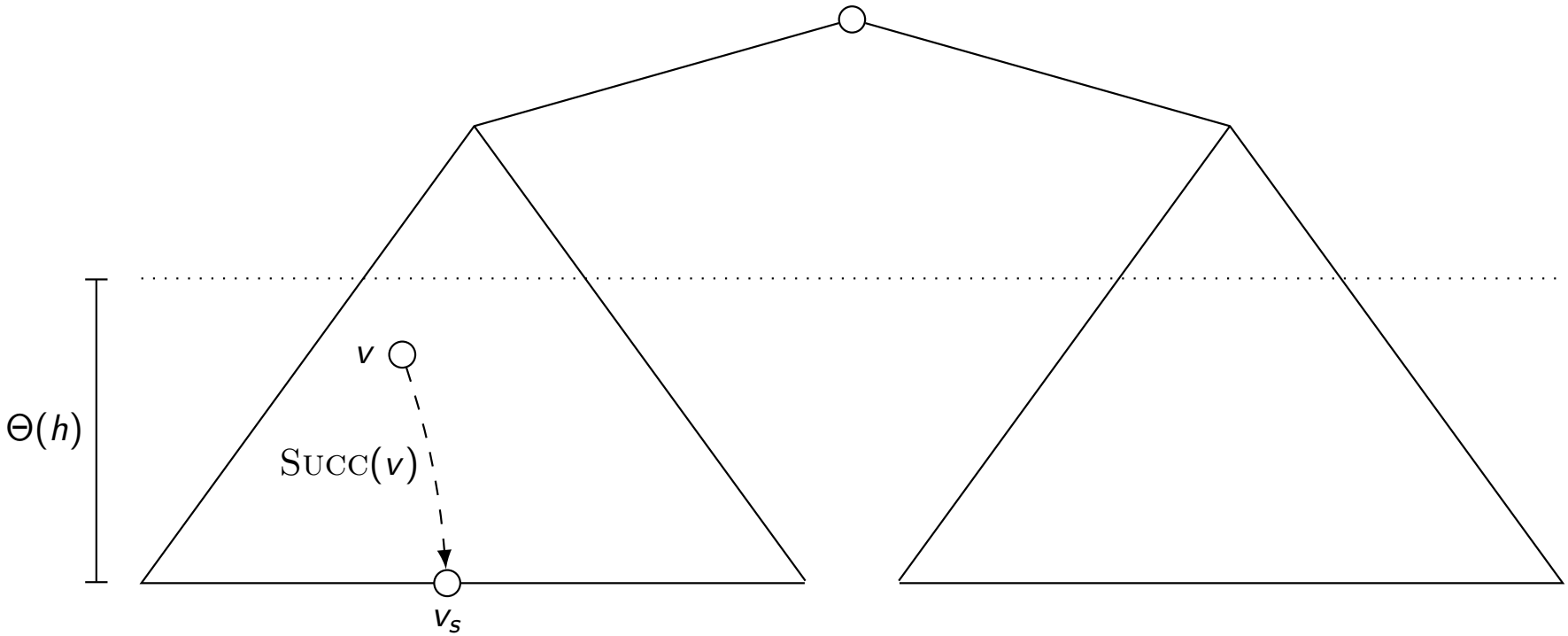
	This paper	Red-black Trees with links
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}_A(1)$ [2]
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg n)$

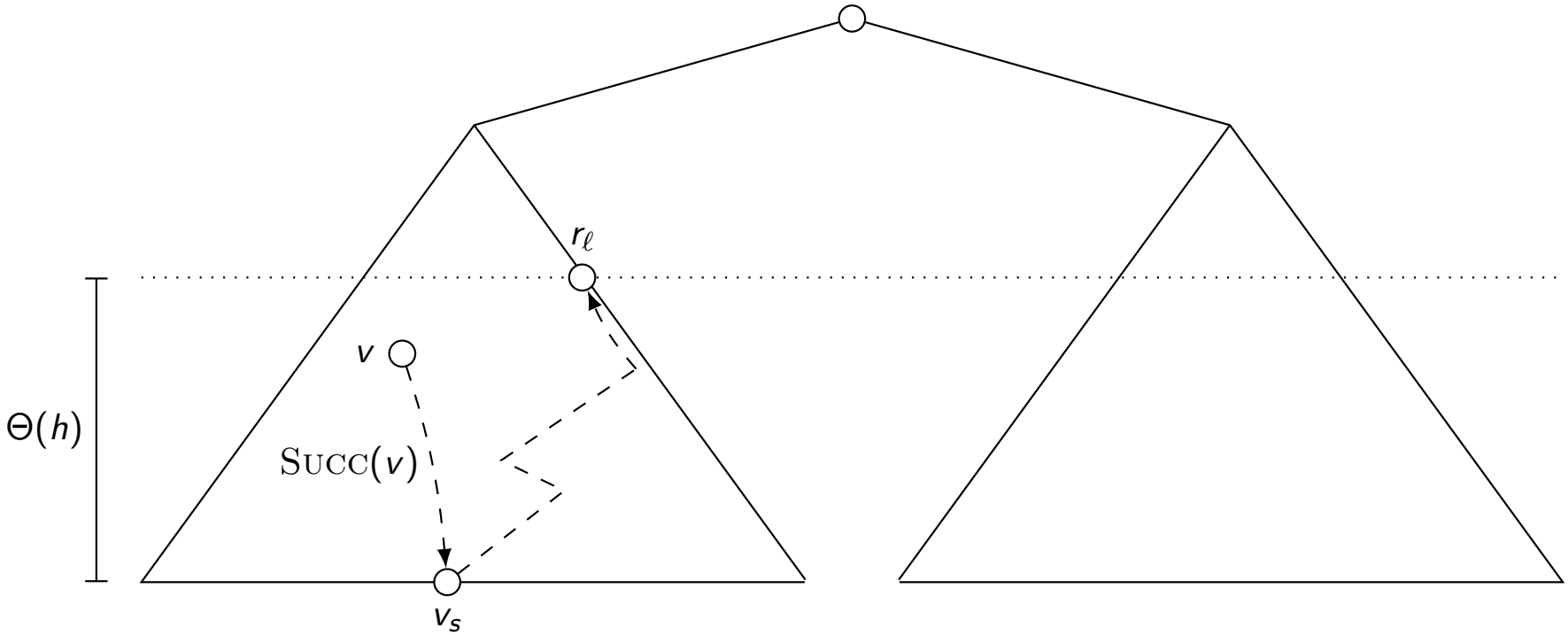


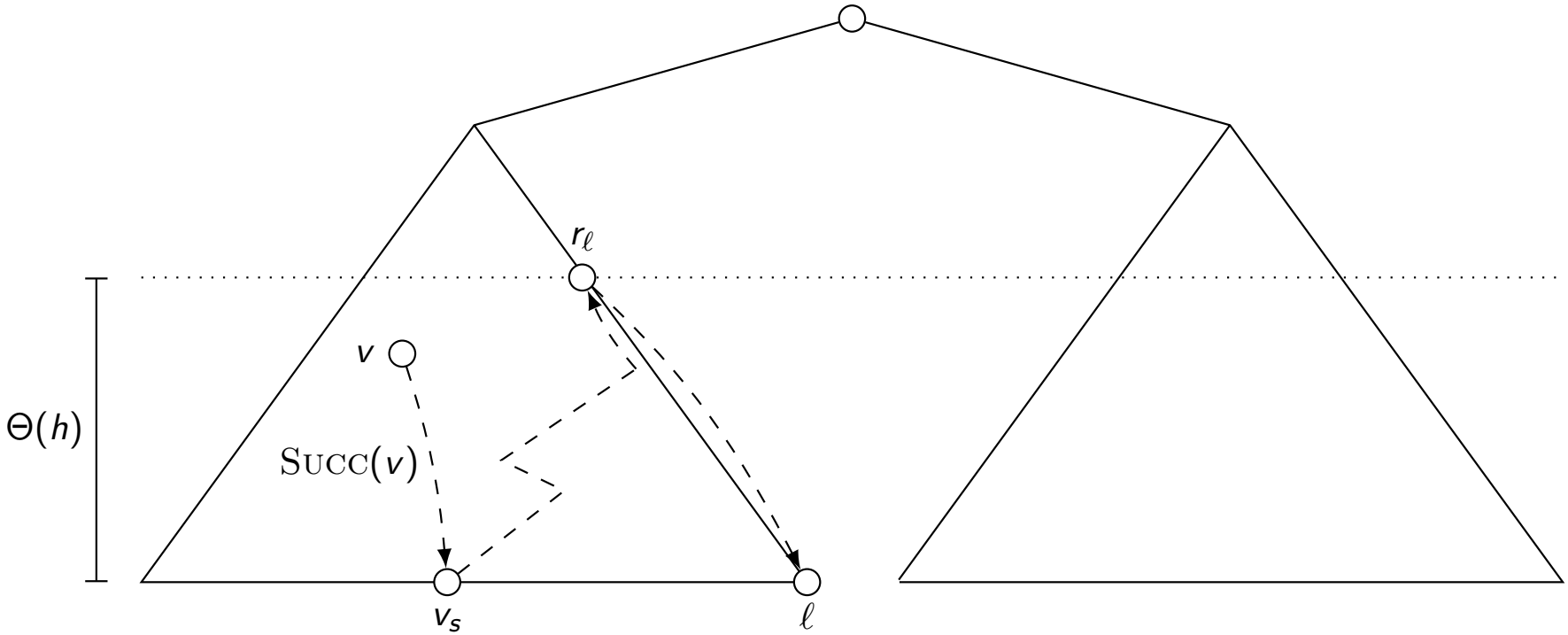
[2] Tarjan. Amortized computational complexity. 1985

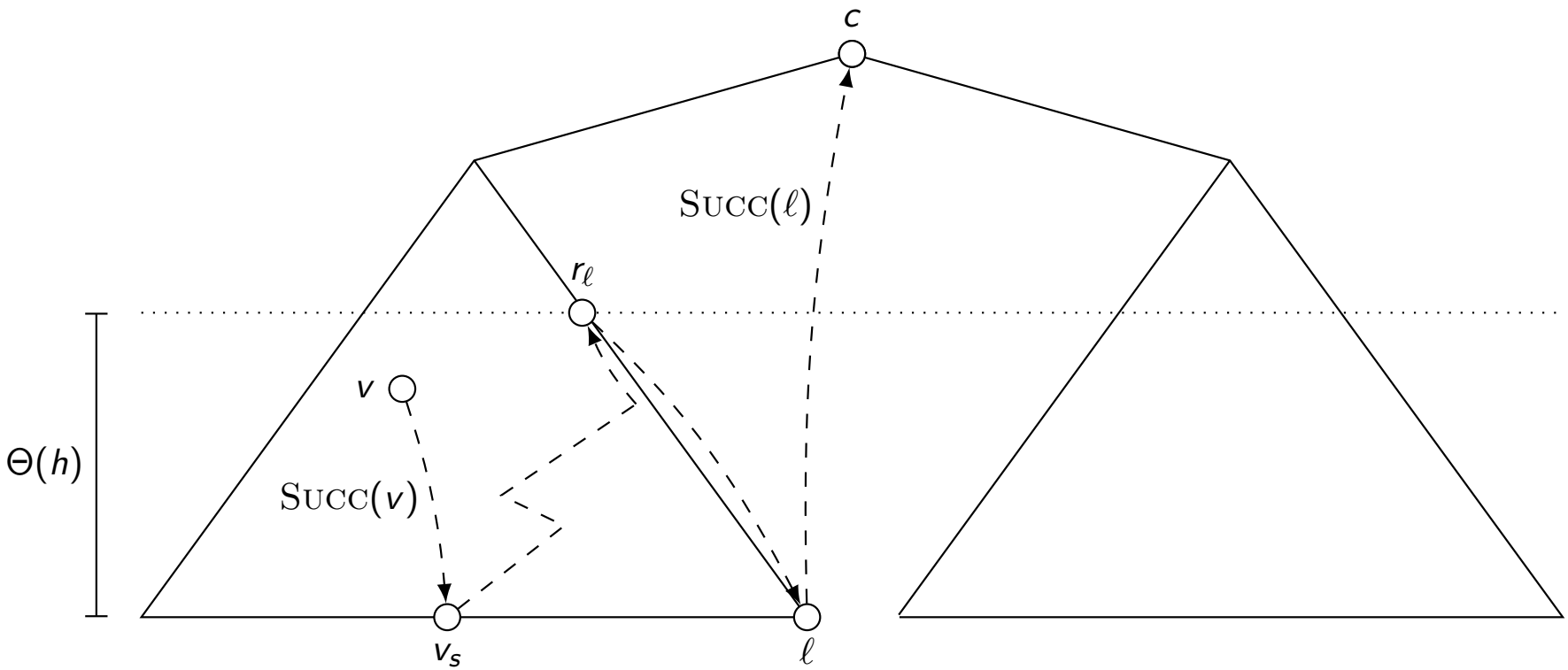

v

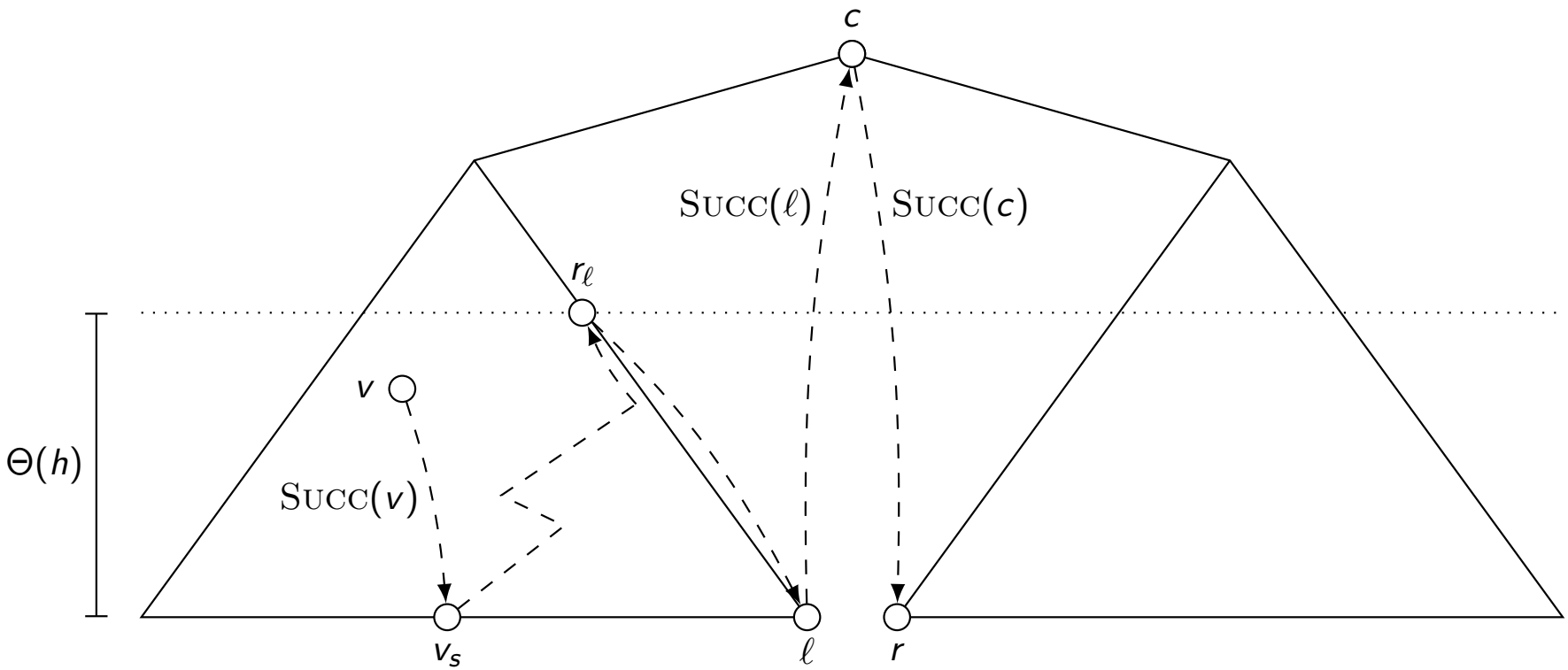

v
Θ(h)

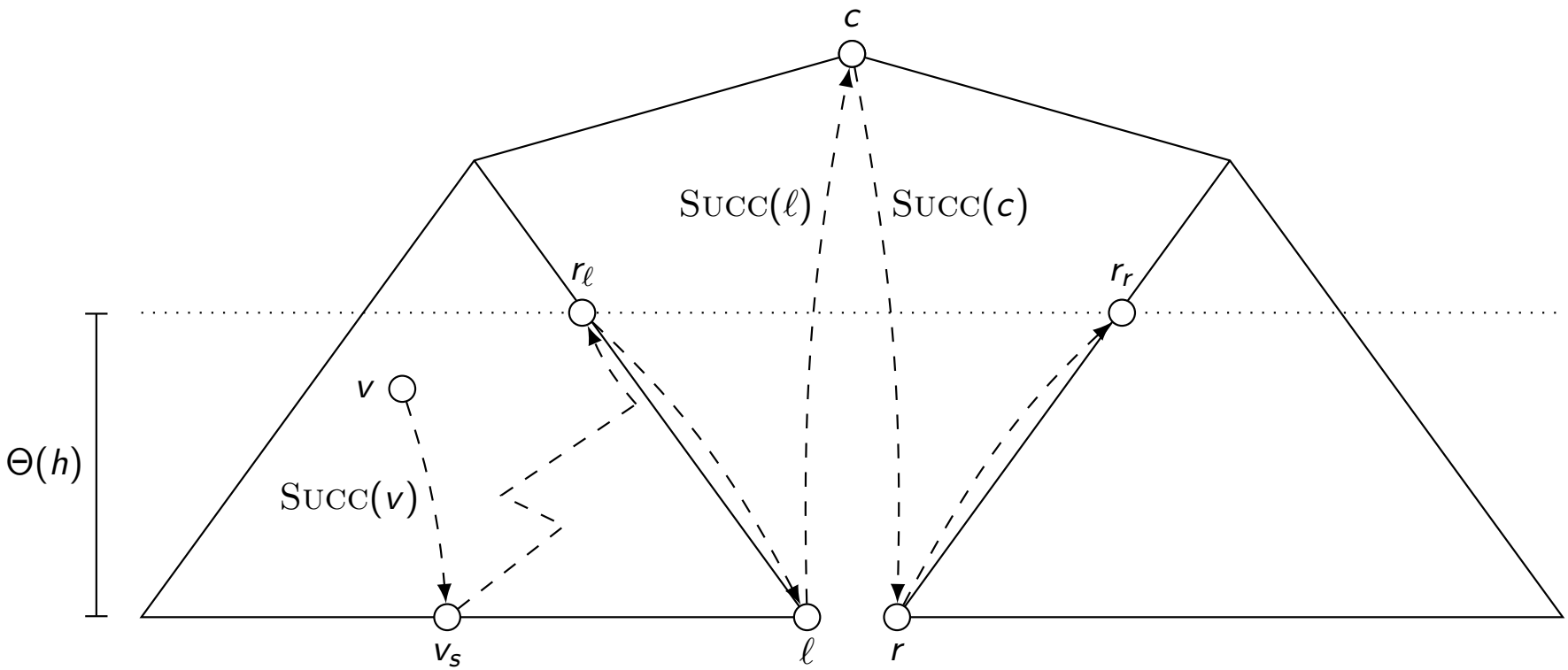

$\Theta(h)$
v
$\textsc{Succ}(v)$
v_s

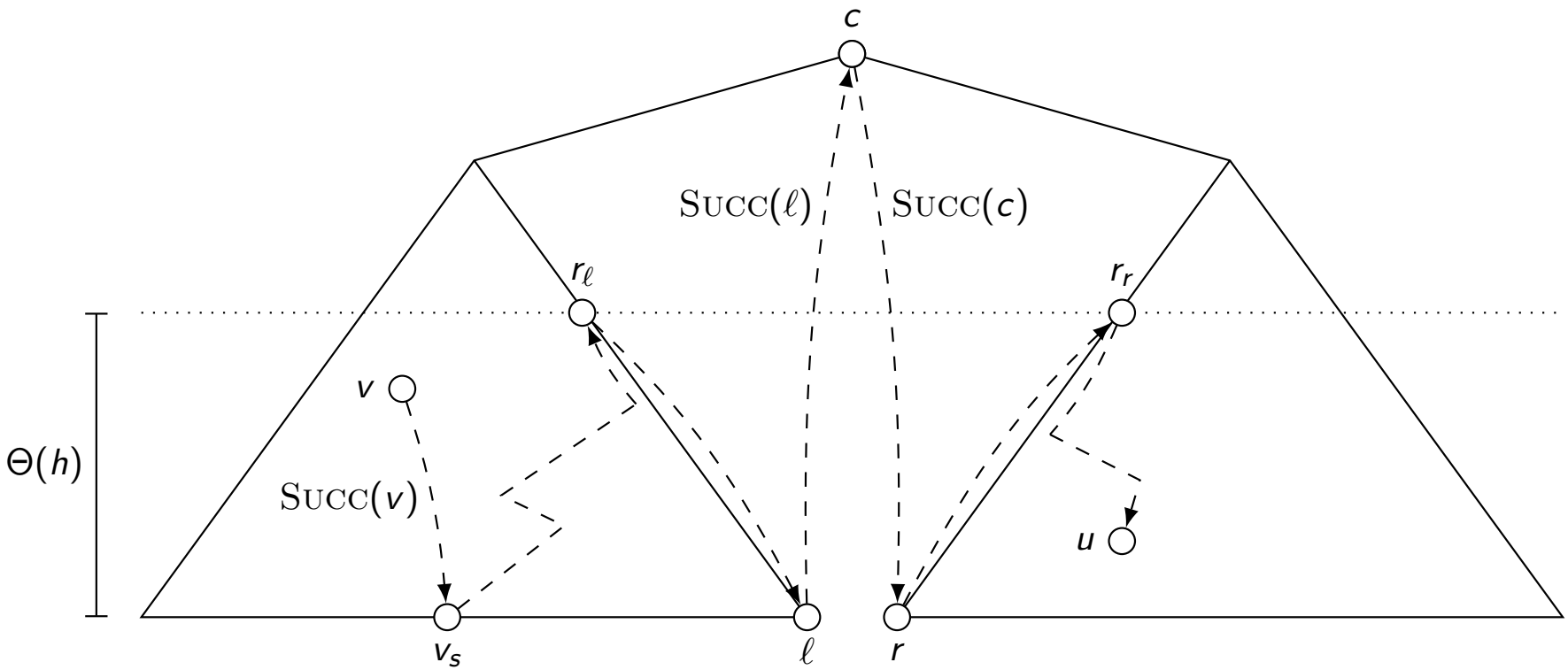

r_ℓ
v
$\Theta(h)$
$\textsc{Succ}(v)$
v_s

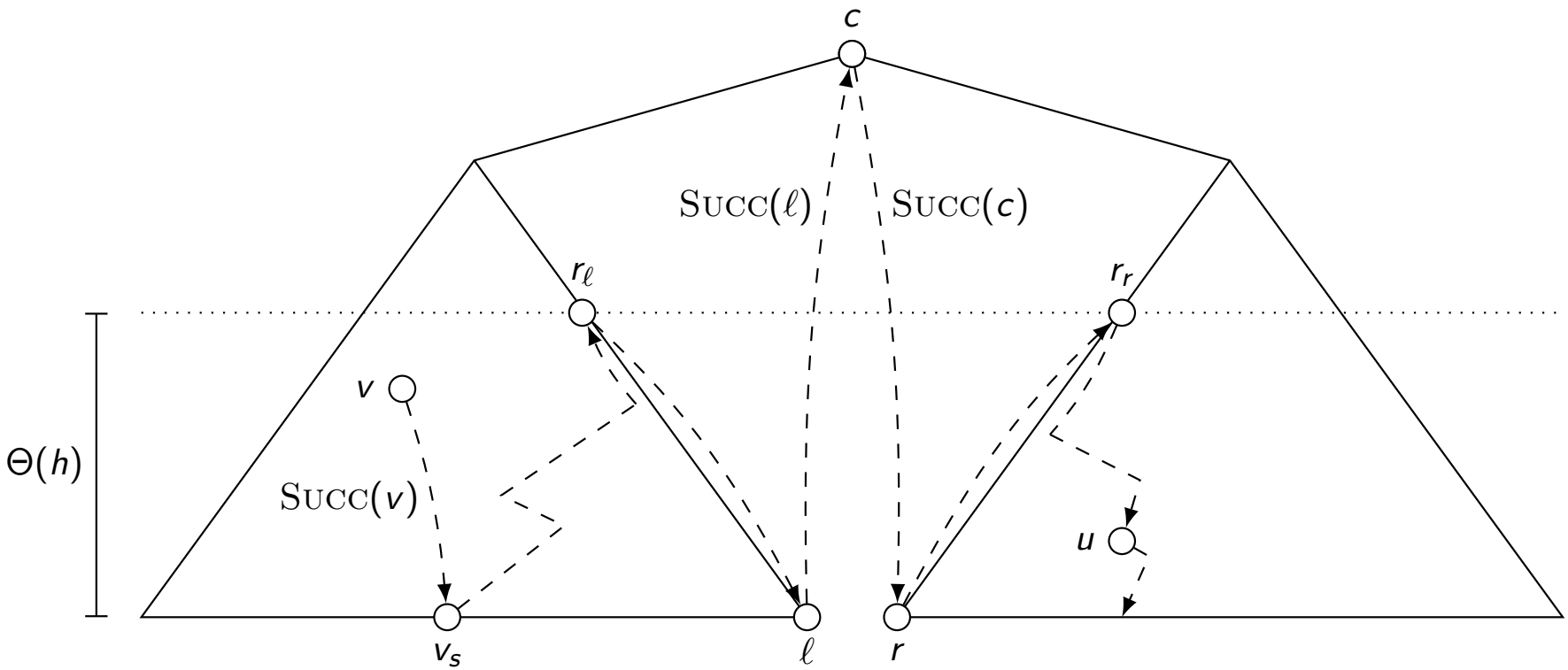

r_ℓ
v
$\Theta(h)$
$\textsc{Succ}(v)$
v_s
ℓ

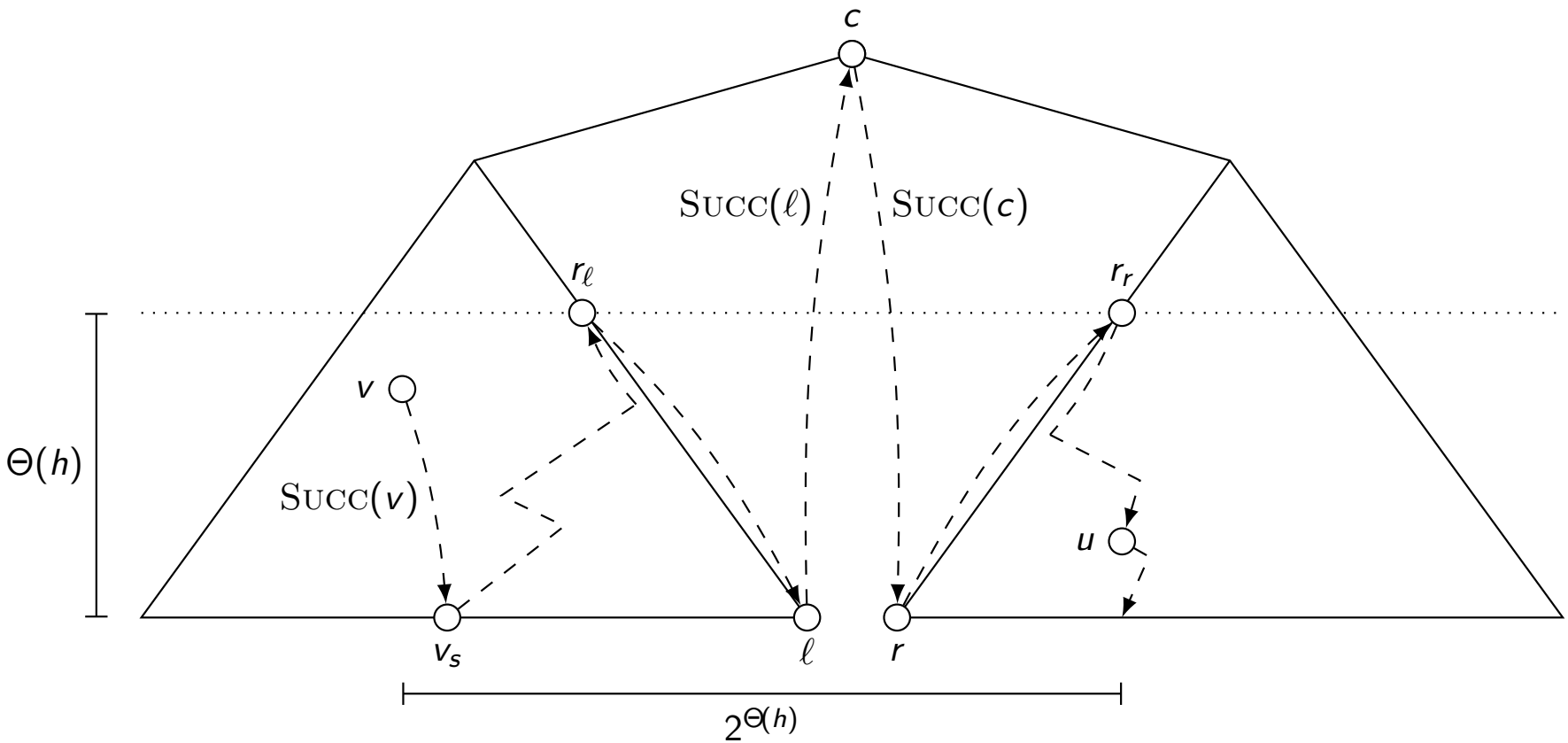

c
Succ(ℓ)
r_ℓ
v
Θ(h)
Succ(v)
v_s
ℓ


c
Succ(ℓ)
Succ(c)
r_ℓ
v
Θ(h)
Succ(v)
v_s
ℓ
r

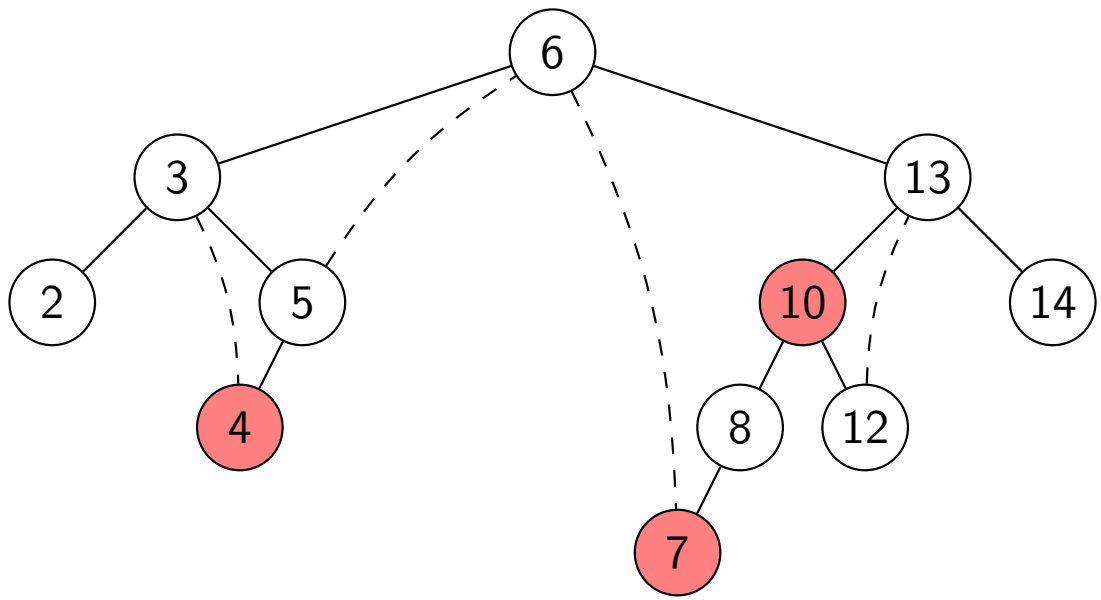

c
$\textsc{Succ}(\ell)$
$\textsc{Succ}(c)$
r_ℓ
r_r
v
$\Theta(h)$
$\textsc{Succ}(v)$
v_s
ℓ
r


c
$\textsc{Succ}(\ell)$
$\textsc{Succ}(c)$
r_ℓ
r_r
$\Theta(h)$
v
$\textsc{Succ}(v)$
u
v_s
ℓ
r


c
$\textsc{Succ}(\ell)$
$\textsc{Succ}(c)$
r_ℓ
r_r
$\Theta(h)$
v
$\textsc{Succ}(v)$
u
v_s
ℓ
r

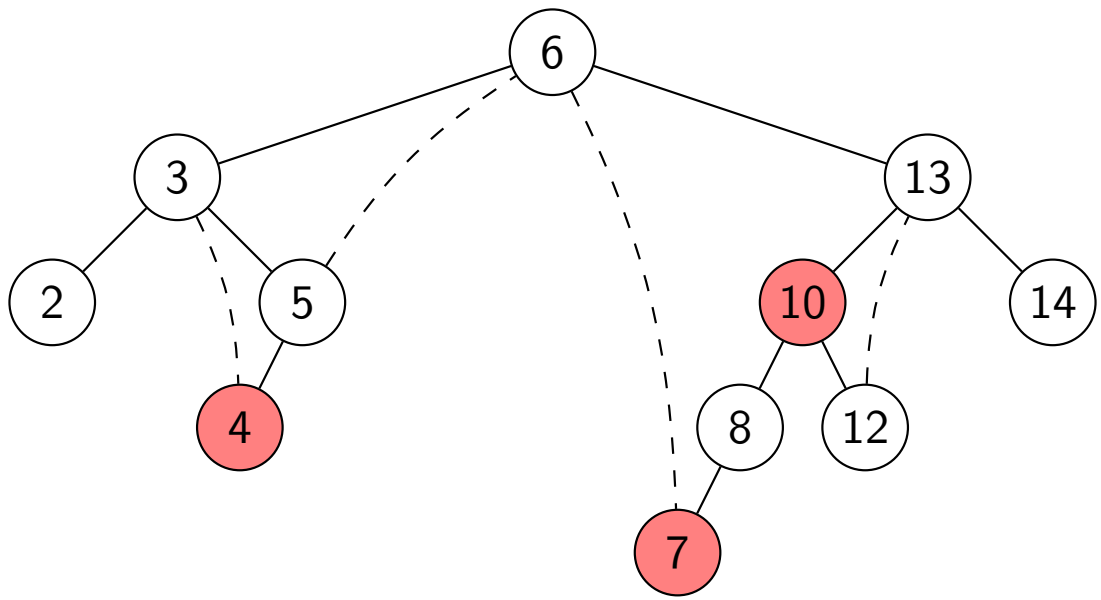

c
Succ(ℓ)
Succ(c)
r_ℓ
r_r
v
Succ(v)
u
Θ(h)
v_s
ℓ
r
$2^{\Theta(h)}$

	This paper	Red-black Trees with links
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}_A(1)$ [2]
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg n)$



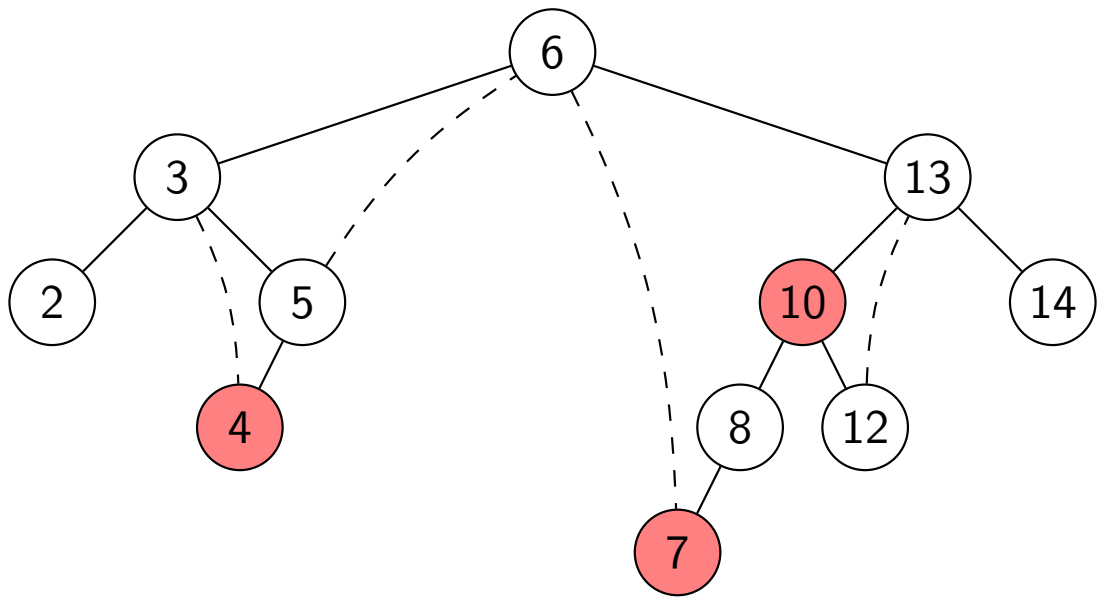
[2] Tarjan. Amortized computational complexity. 1985

	This paper	Red-black Trees with links
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}_A(1)$ [2]
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg d)$



[2] Tarjan. Amortized computational complexity. 1985

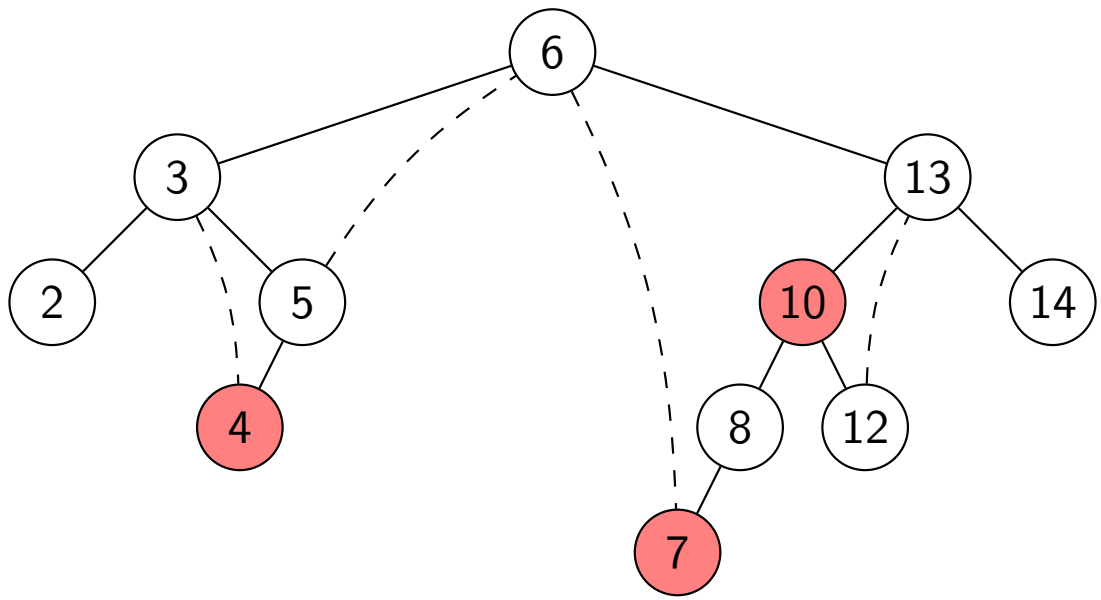
	This paper	Red-black Trees with links	Level-linked (a, b)-trees[3]
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}_A(1)$ [2]	$\mathcal{O}_A(1)$
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg d)$



[2] Tarjan. Amortized computational complexity. 1985

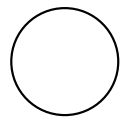
[3] Huddleston and Mehlhorn. A new data structure for representing sorted lists. 1982

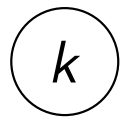
	This paper	Red-black Trees with links	Level-linked (a, b)-trees[3]
Succ Pred	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$	$\mathcal{O}_A(1)$ [2]	$\mathcal{O}_A(1)$
FingerSearch	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg d)$	$\mathcal{O}(\lg d)$

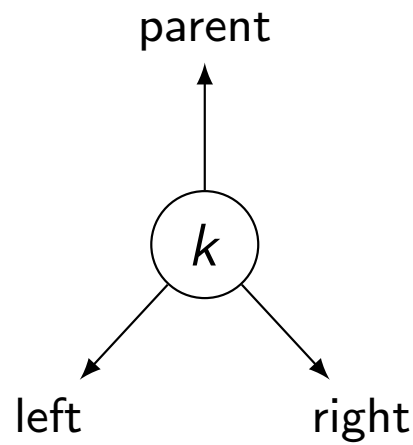


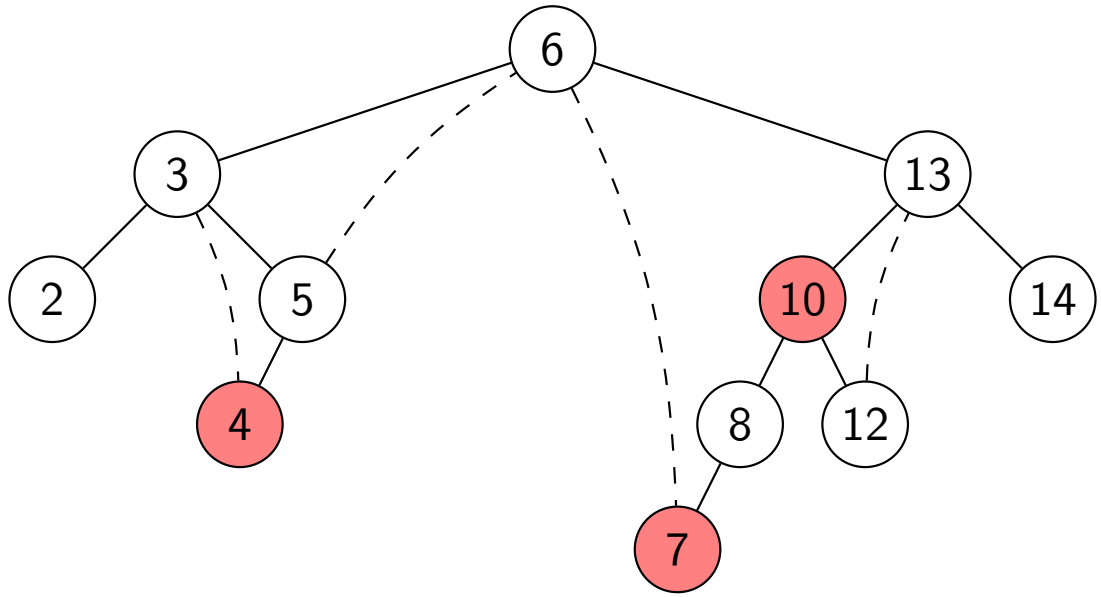
[2] Tarjan. Amortized computational complexity. 1985

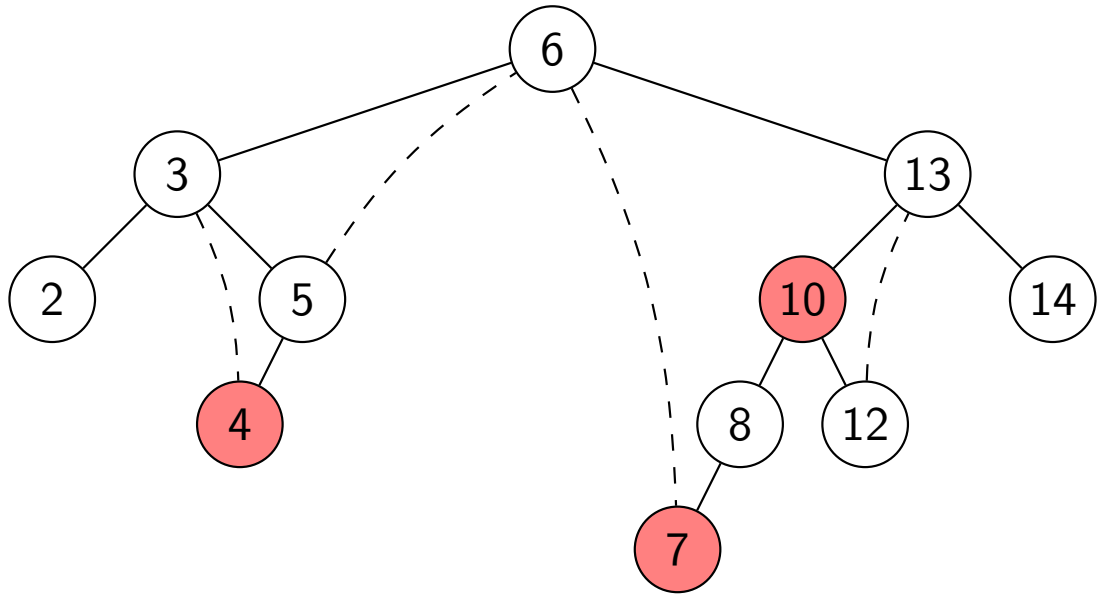
[3] Huddleston and Mehlhorn. A new data structure for representing sorted lists. 1982



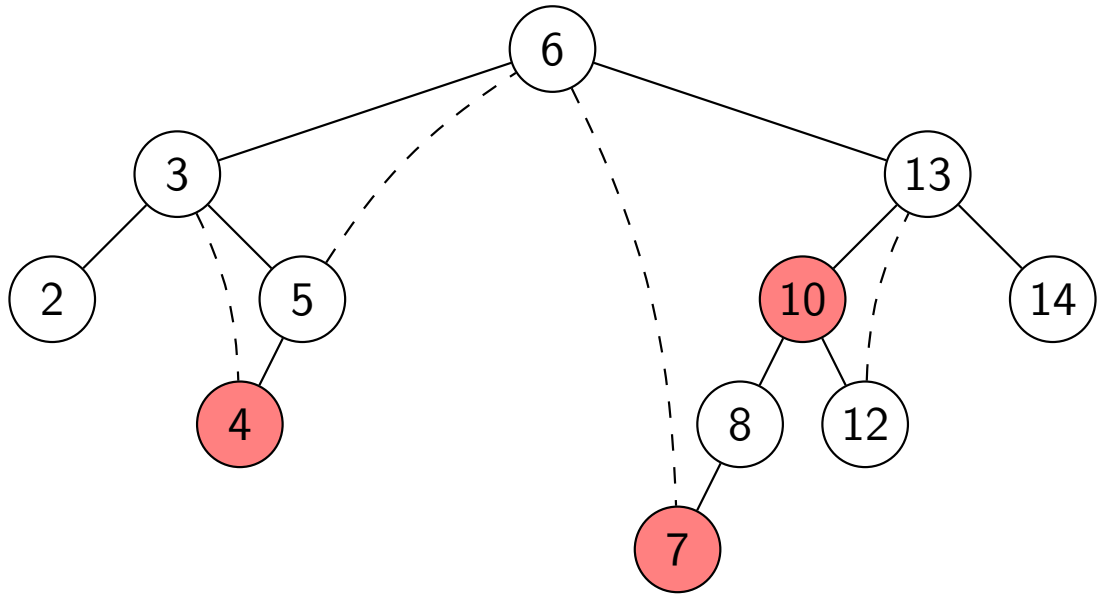

k


parent
k
left
right

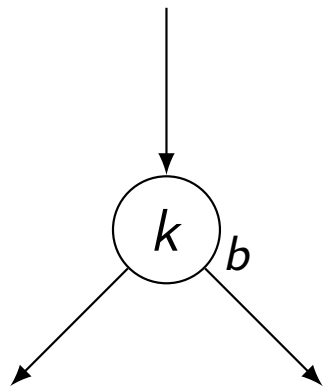

6
3
13
2
5
10
14
4
8
12
7



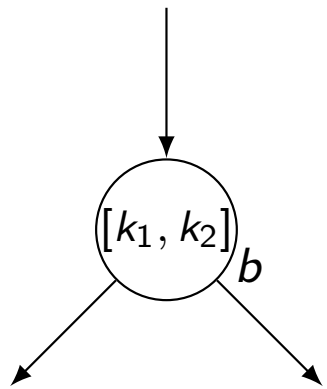
1. Bit marking red/black



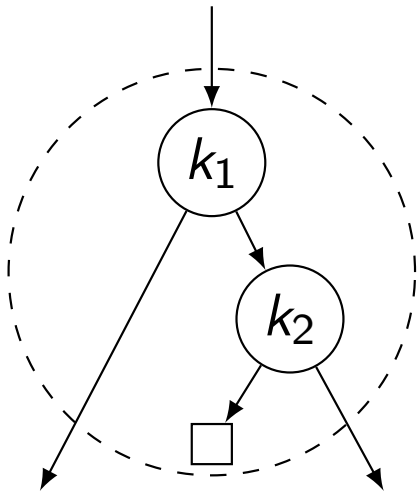
1. Bit marking red/black
2. Pointers to Succ and Pred



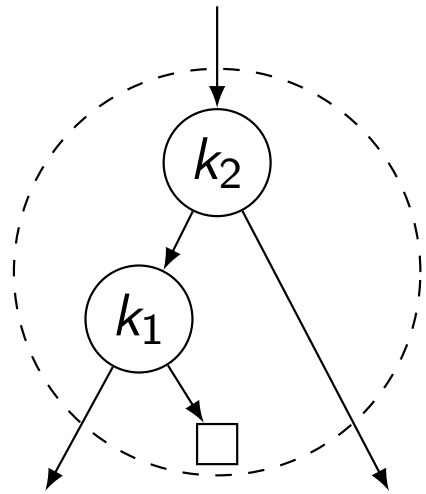
Brown. A storage scheme for height-balanced trees. 1978.



Brown. A storage scheme for height-balanced trees. 1978.

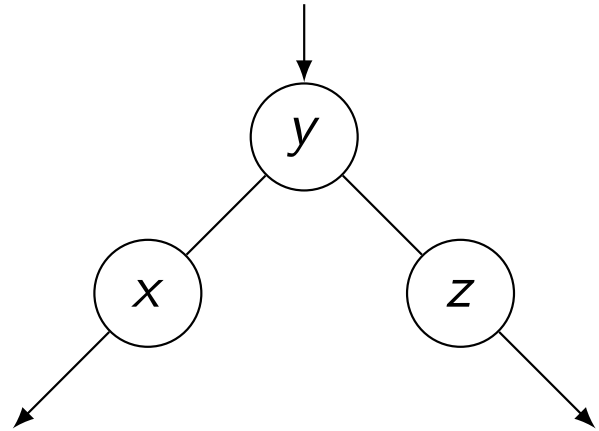


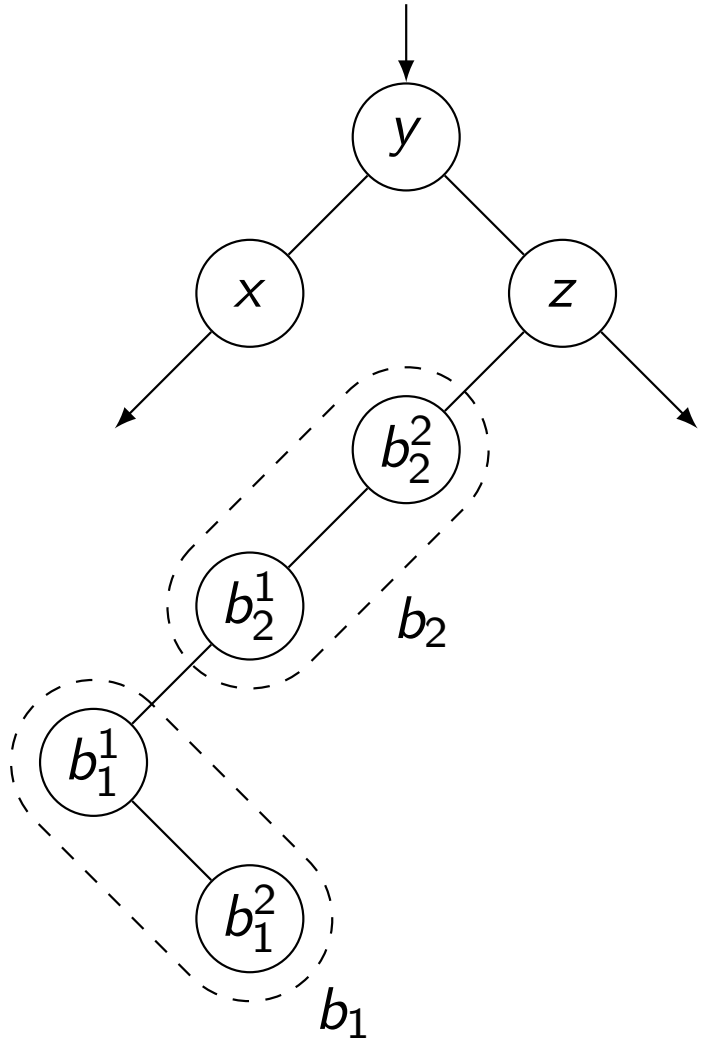
$b = 0$

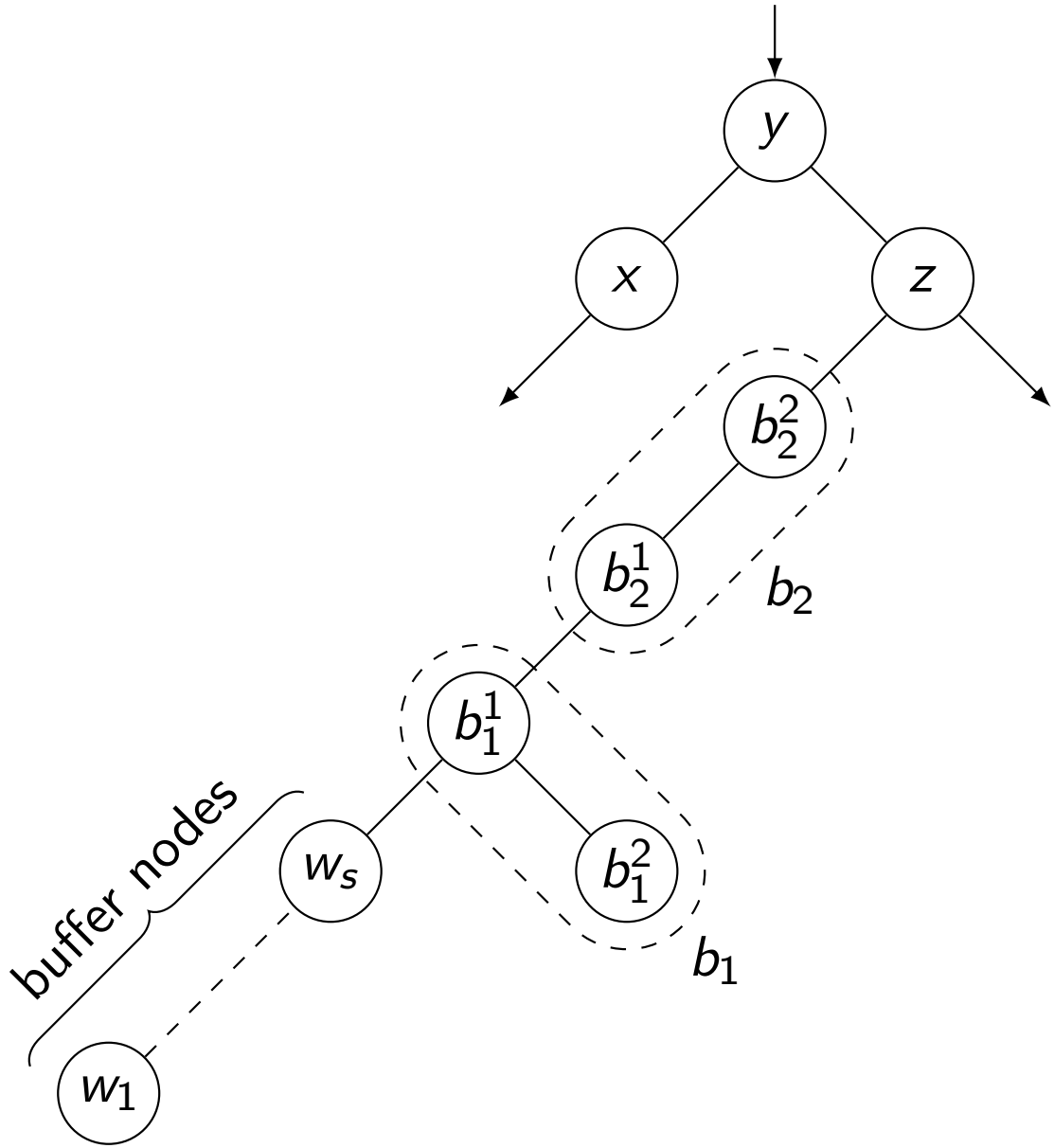


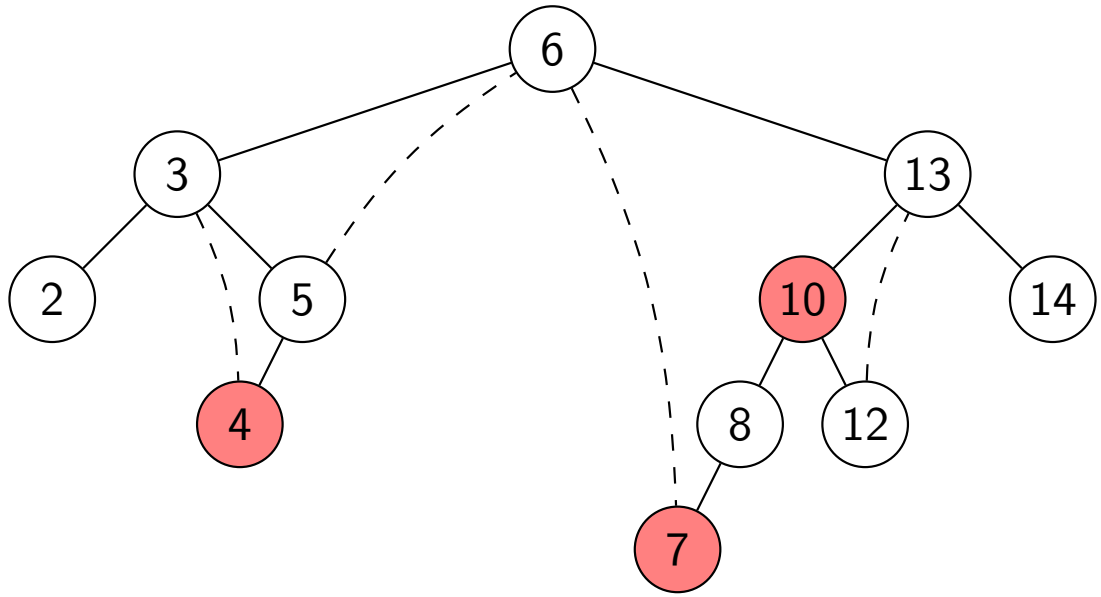
$b = 1$

Brown. A storage scheme for height-balanced trees. 1978.

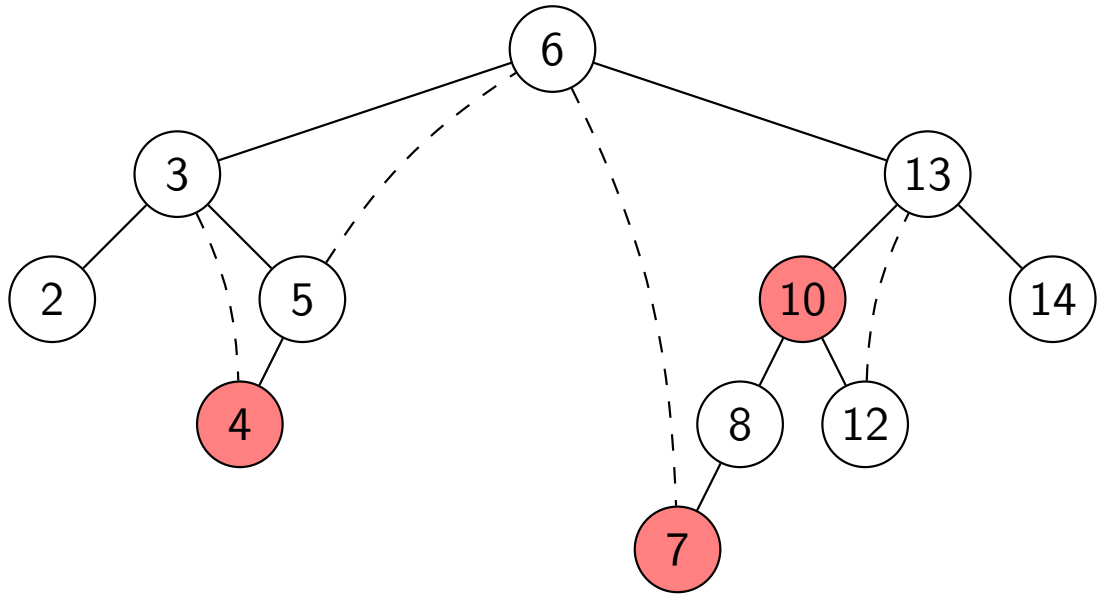

y
x
z


y
x
z
b_2^2
b_2^1
b_2
b_1^1
b_1^2
b_1


y
x
z
b_2^2
b_2^1
b_2
b_1^1
w_s
b_1^2
b_1
buffer nodes
w_1

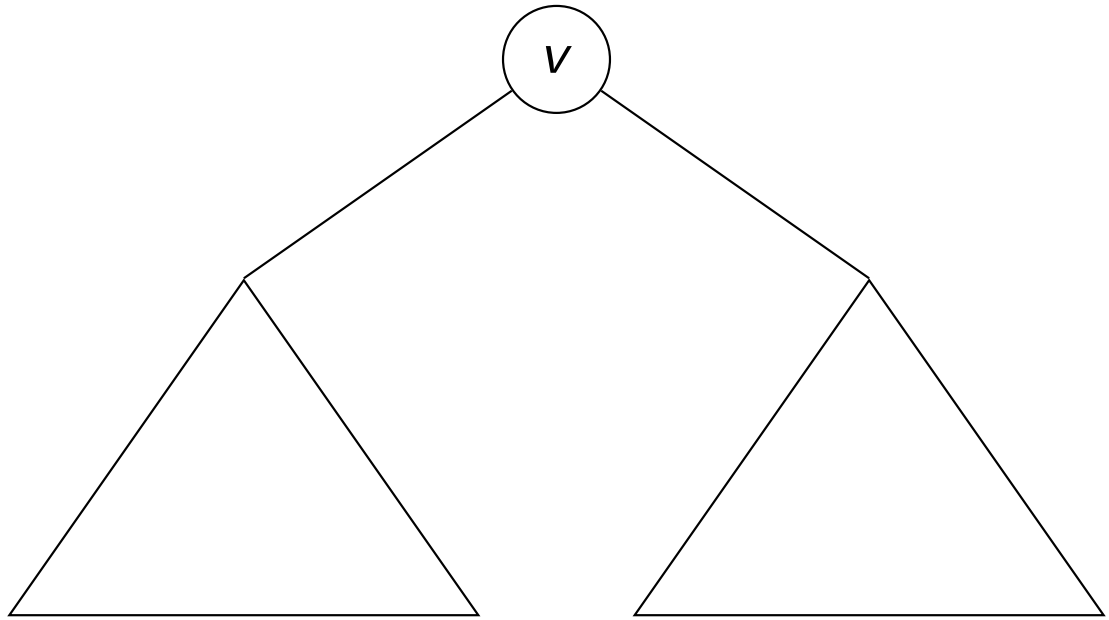


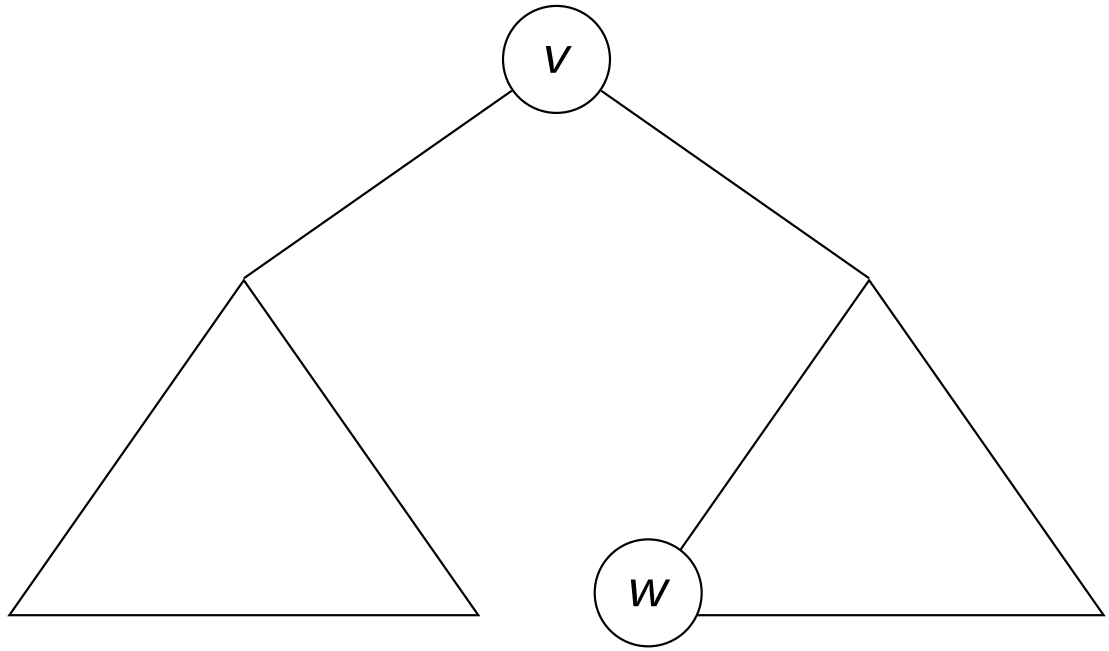
1. Bit marking red/black
2. Pointers to Succ and Pred

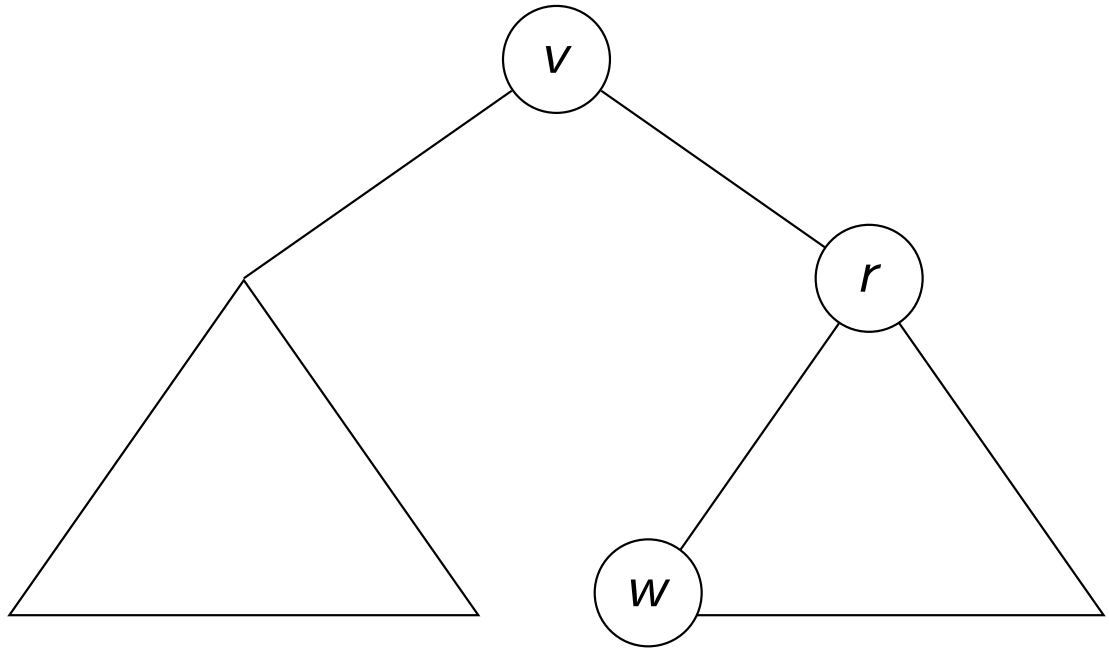


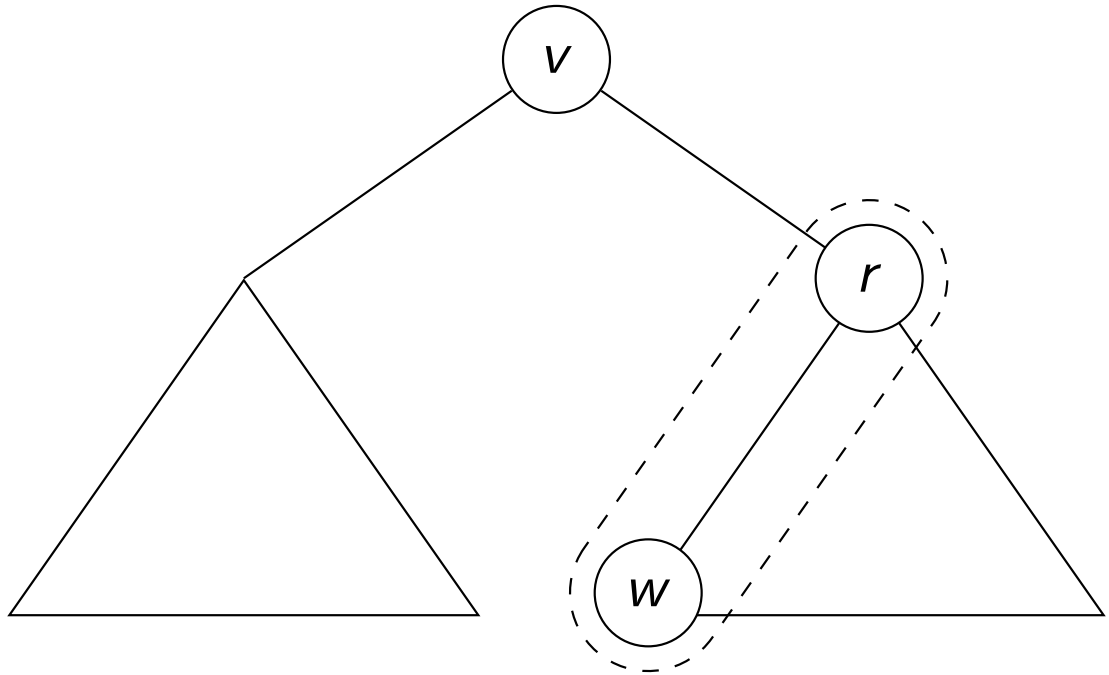
✓ 1. Bit marking red/black

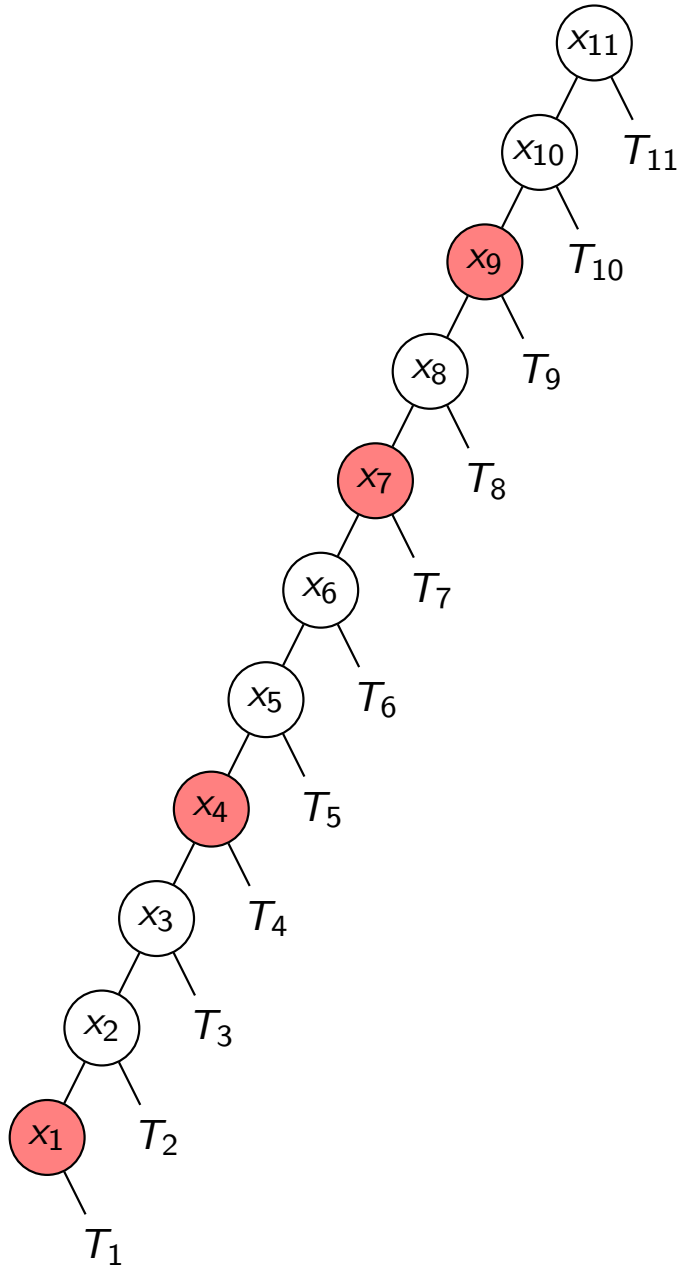
2. Pointers to SUCC and PRED

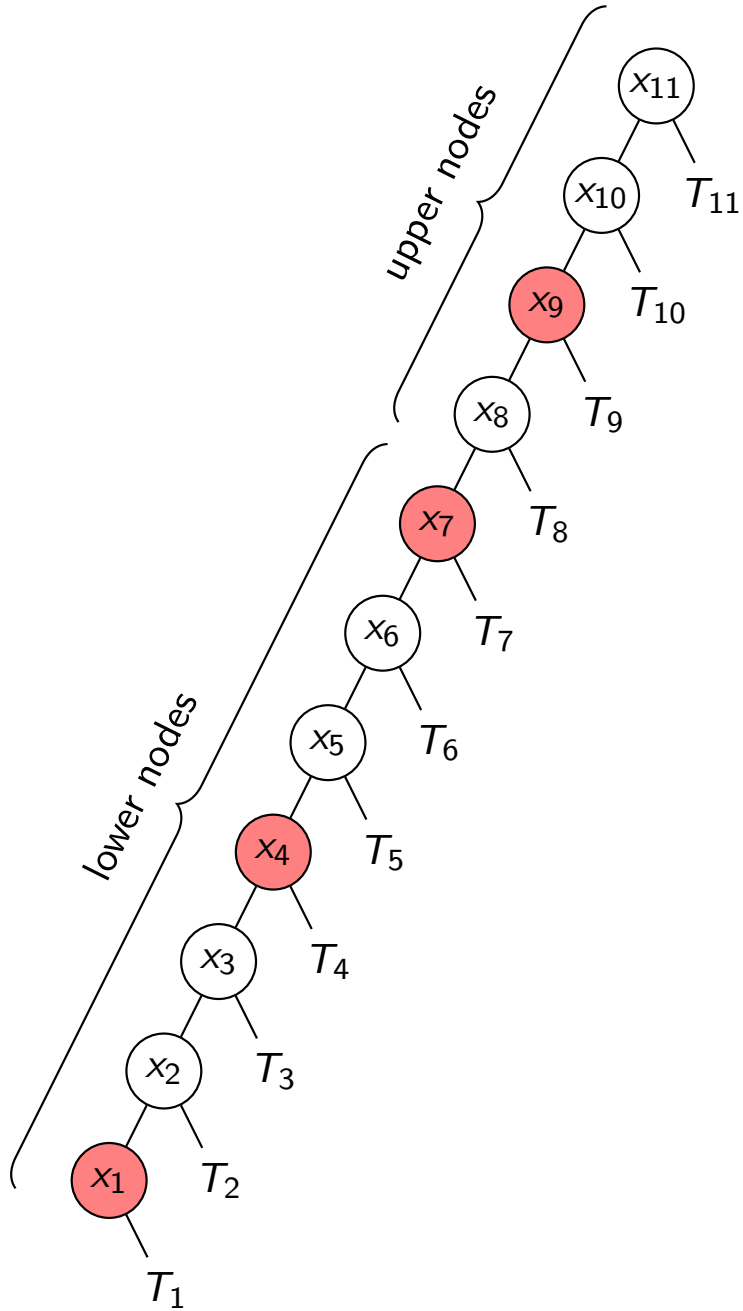

v

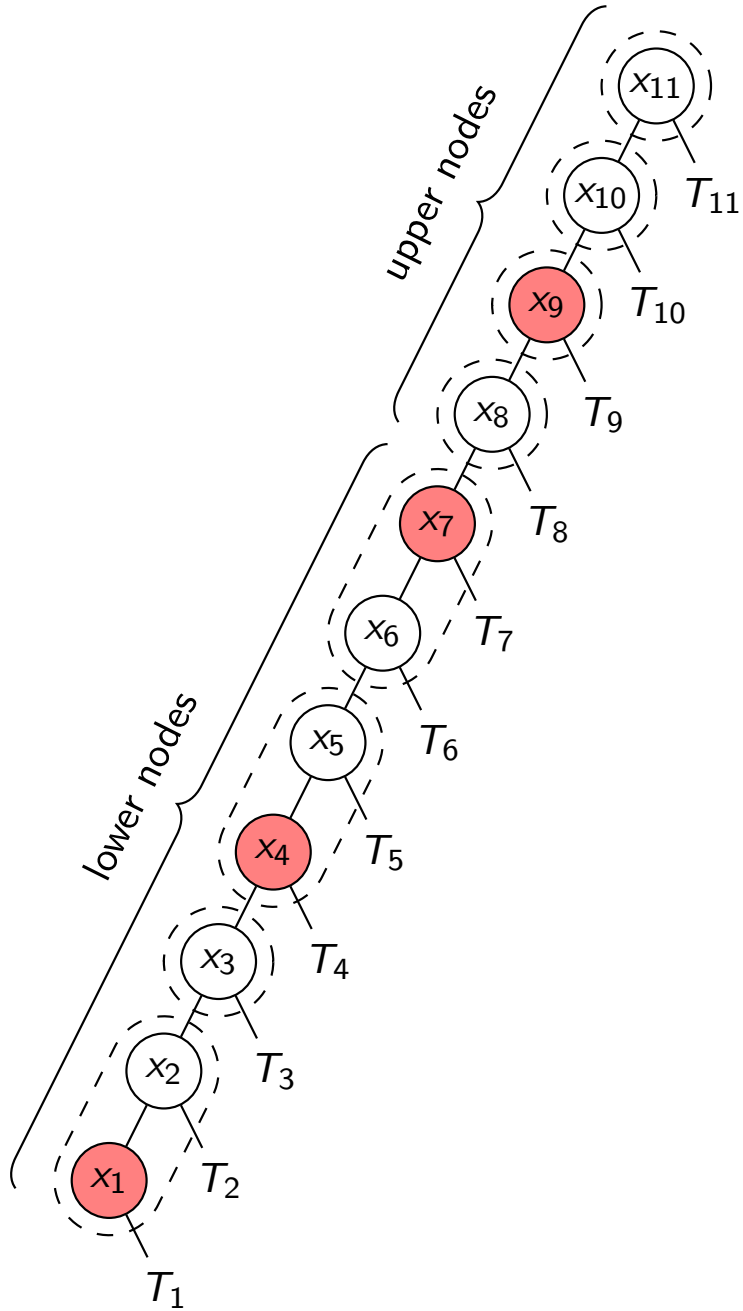

v
w


v
r
w

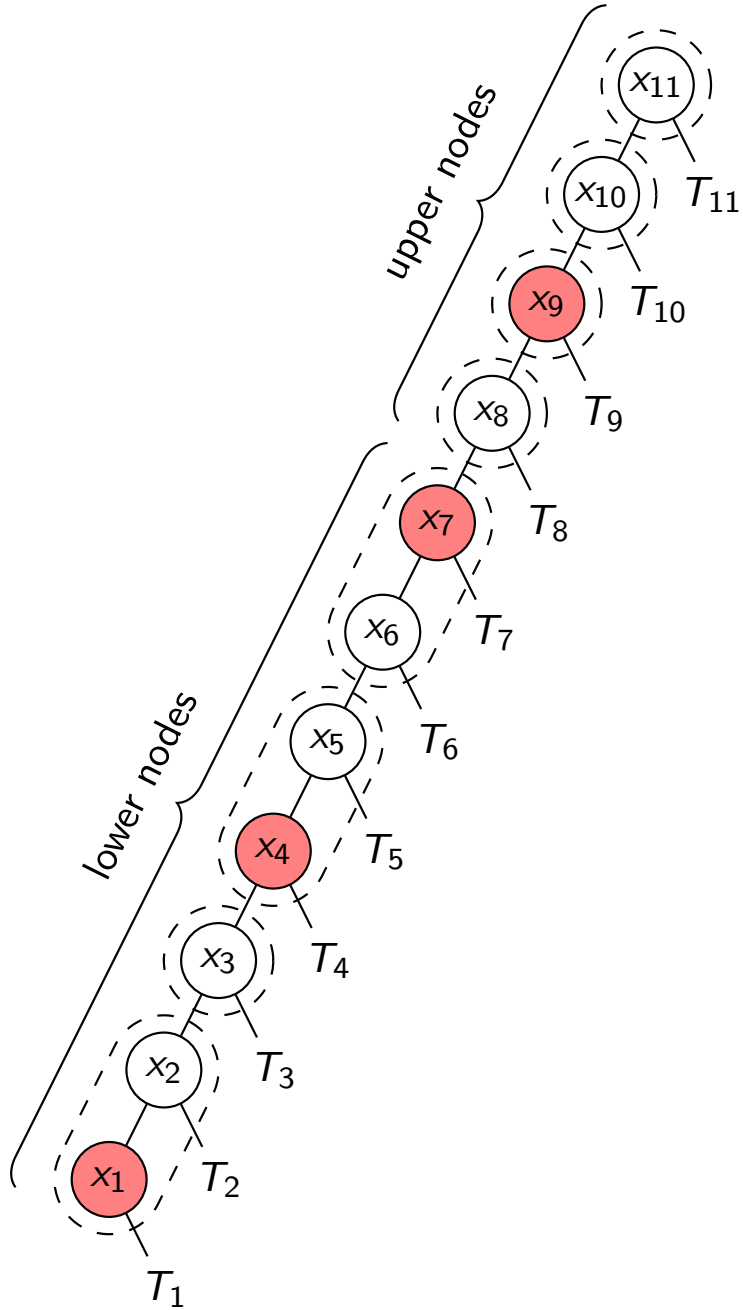

v
r
w

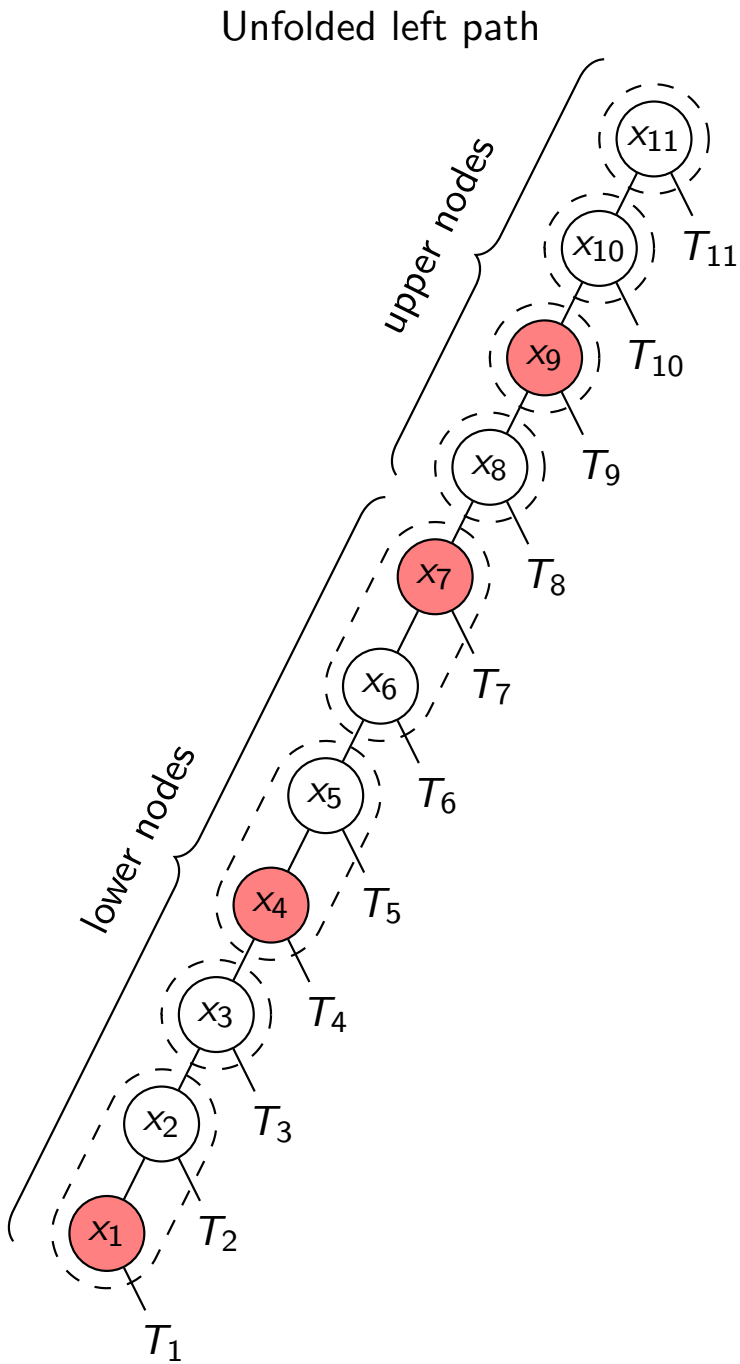

x_{11}
x_{10}
T_{11}
x_9
T_{10}
x_8
T_9
x_7
T_8
x_6
T_7
x_5
T_6
x_4
T_5
x_3
T_4
x_2
T_3
x_1
T_2
T_1


upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1


upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1

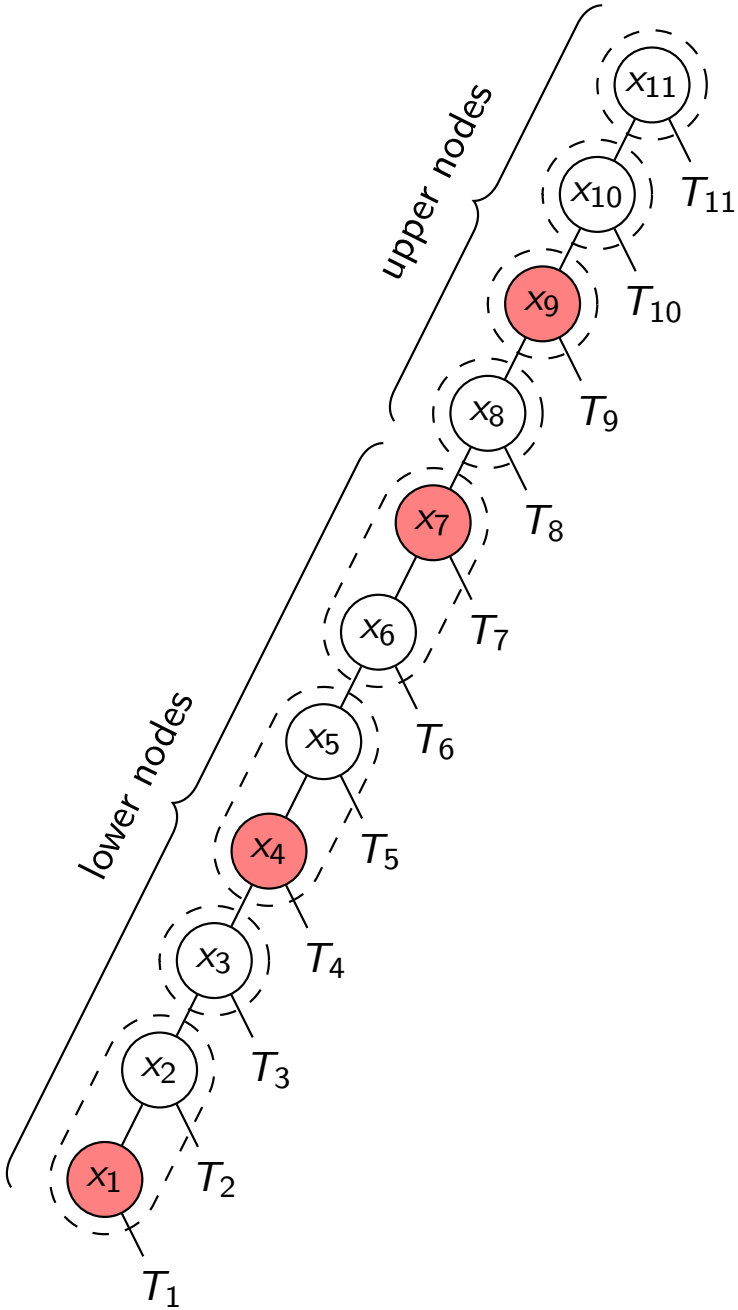
Unfolded left path

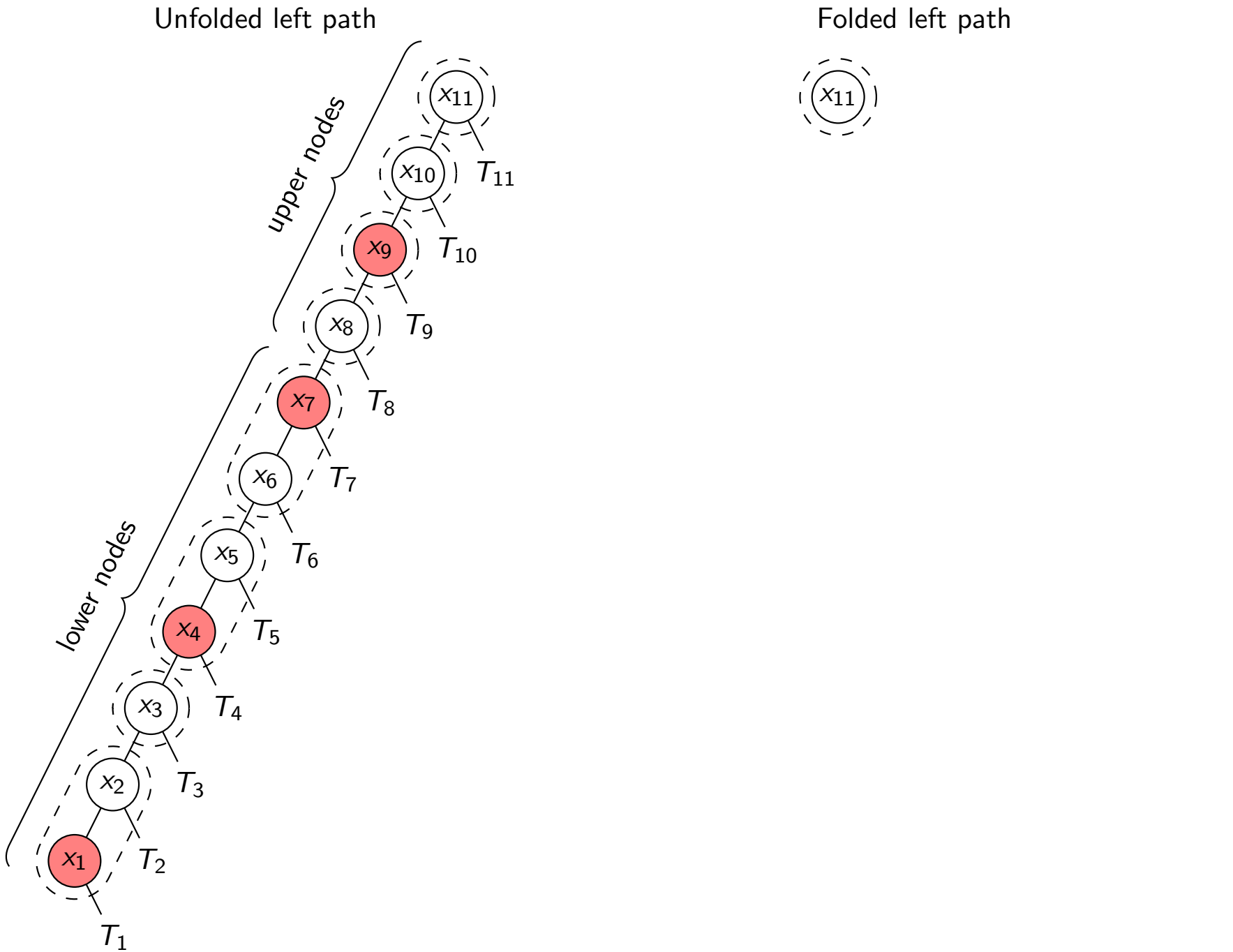



Unfolded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1

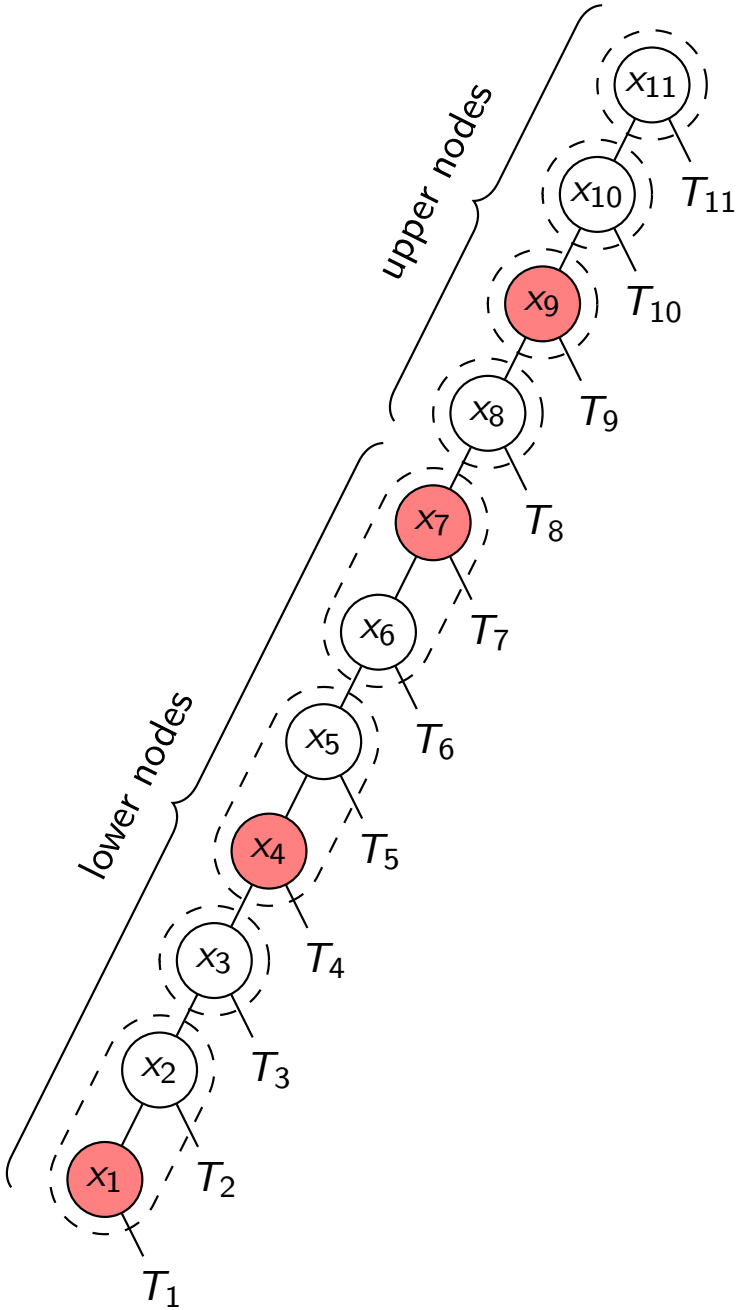
Unfolded left path

Folded left path

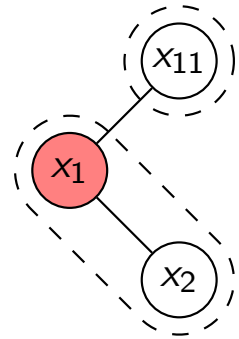



Unfolded left path
Folded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1

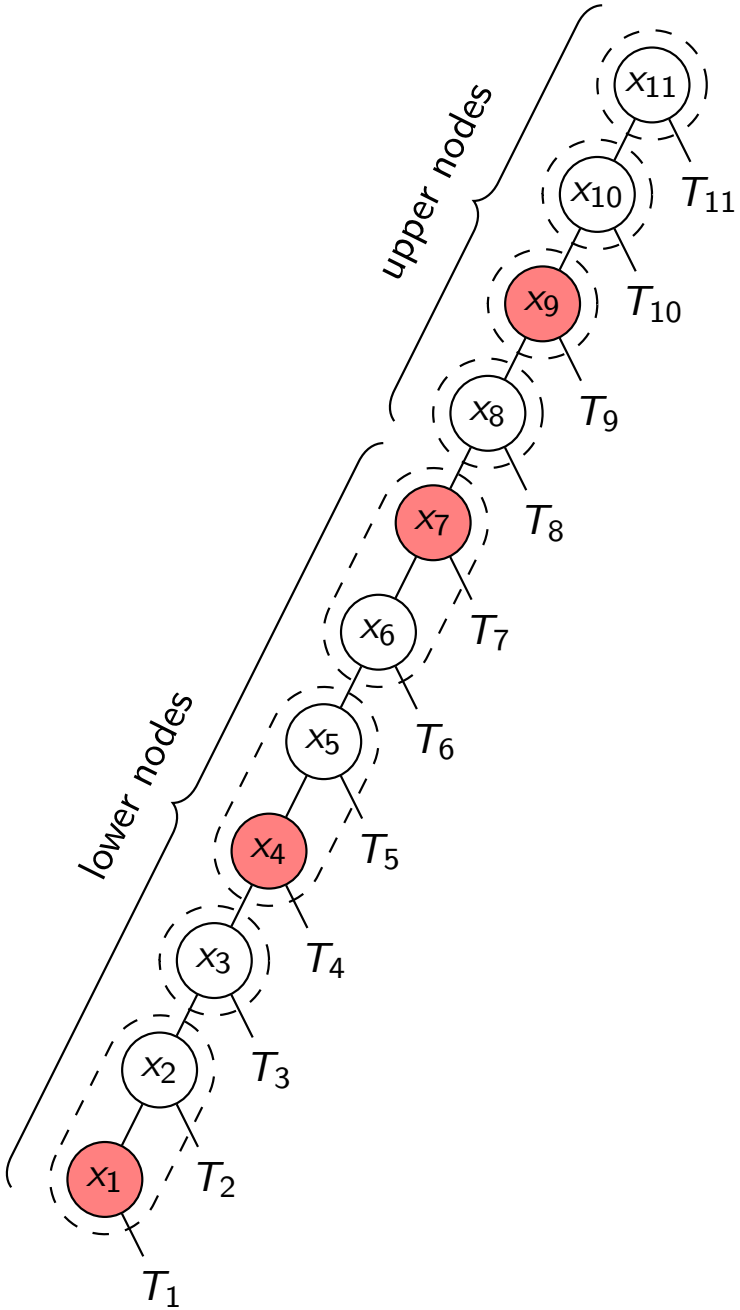
Unfolded left path



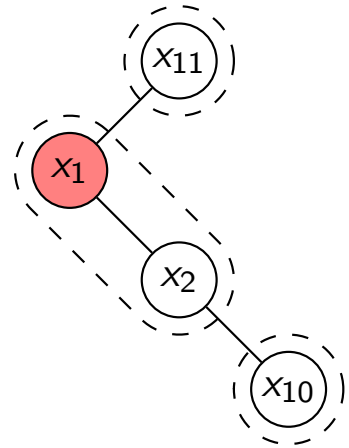
Folded left path



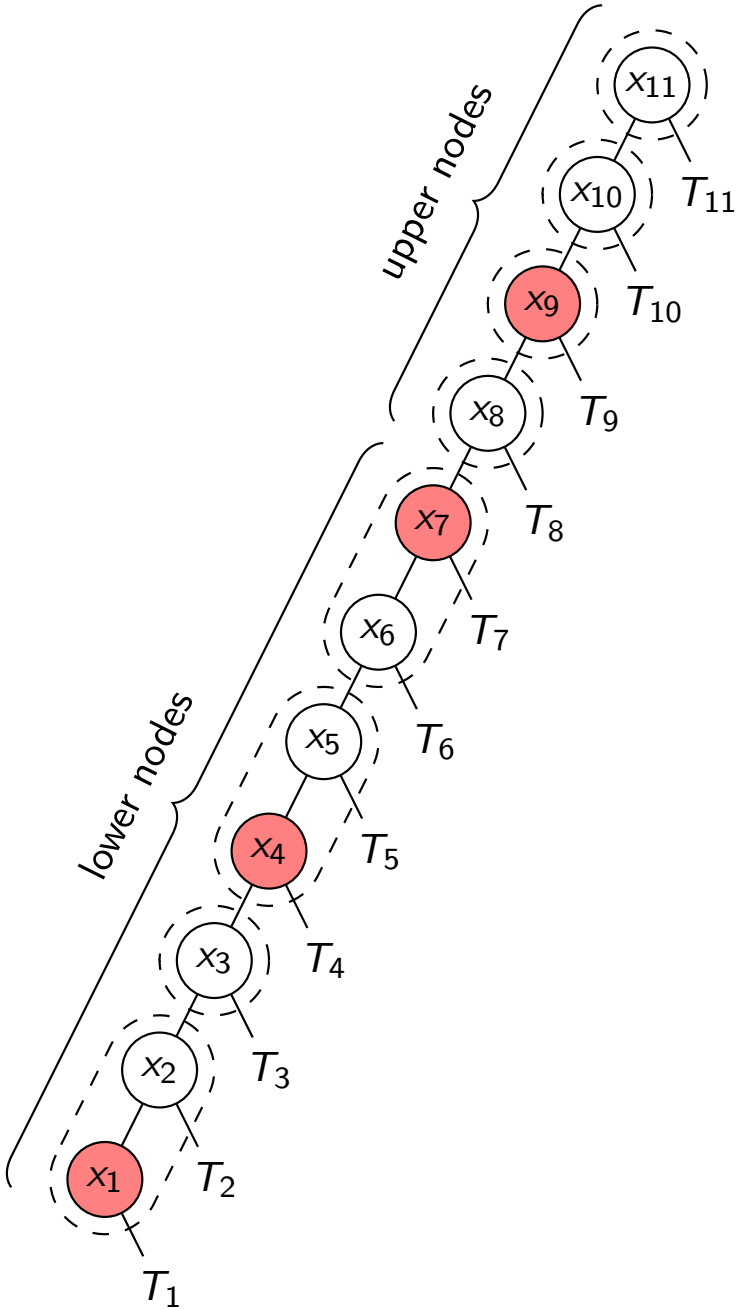
Unfolded left path



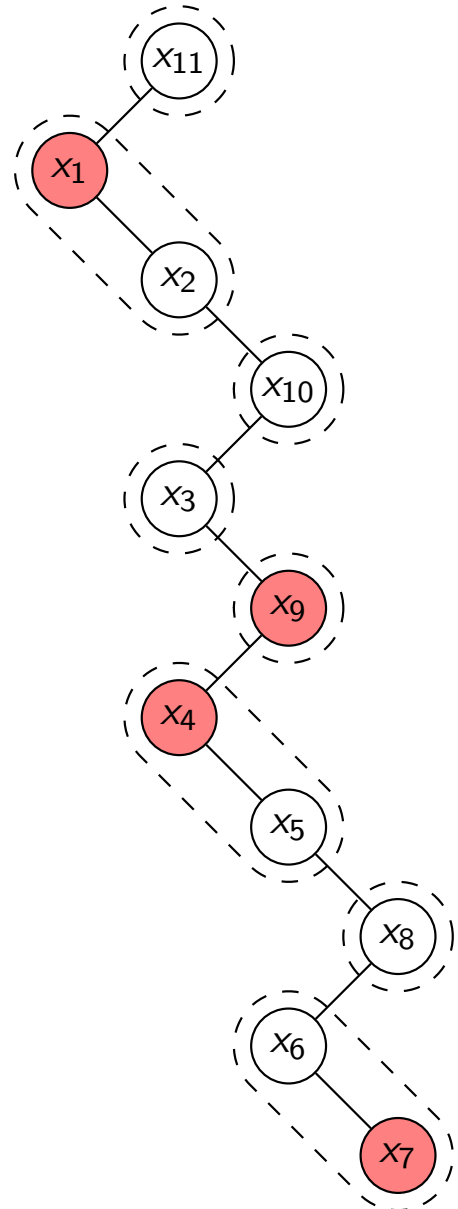
Folded left path

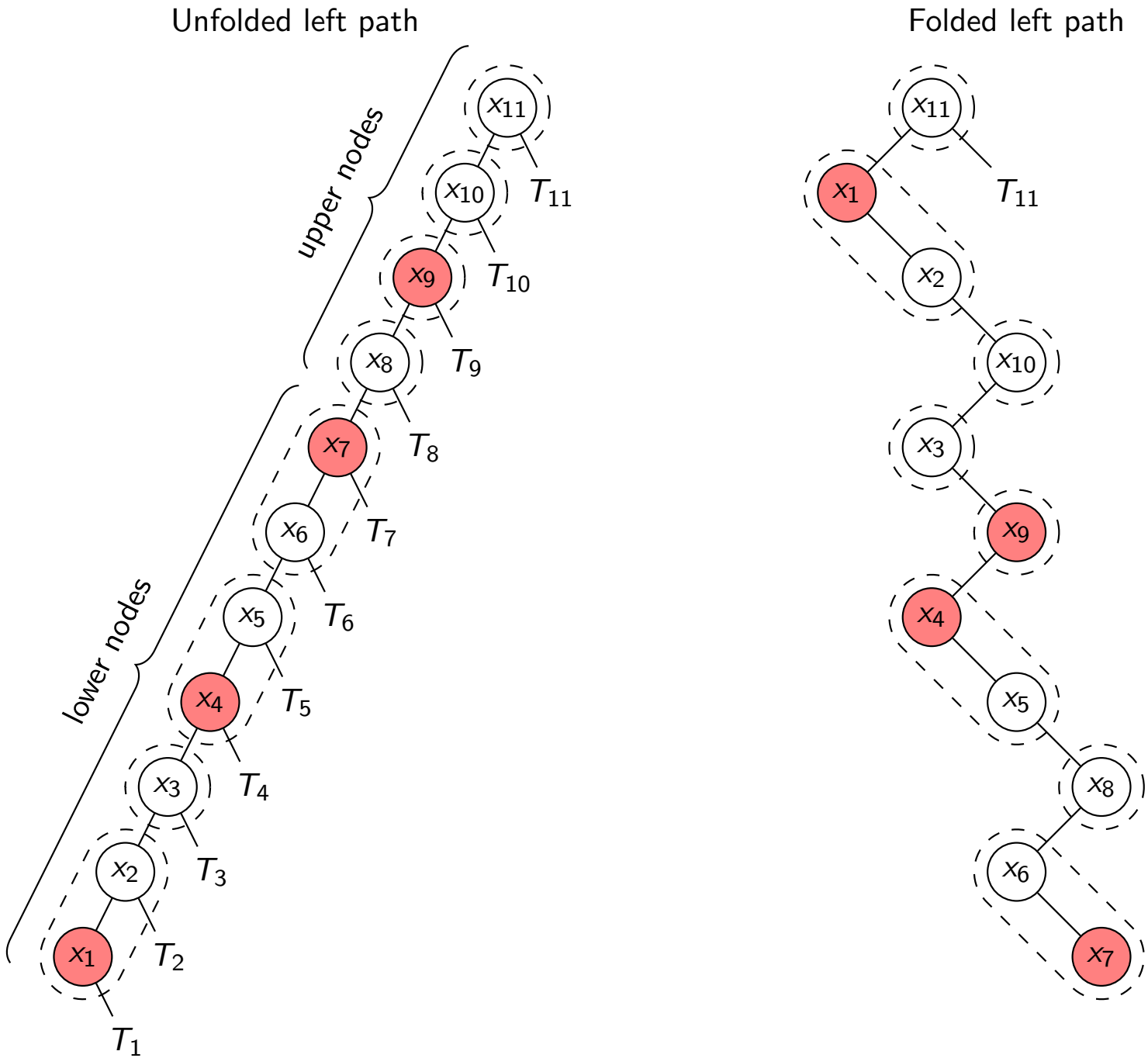


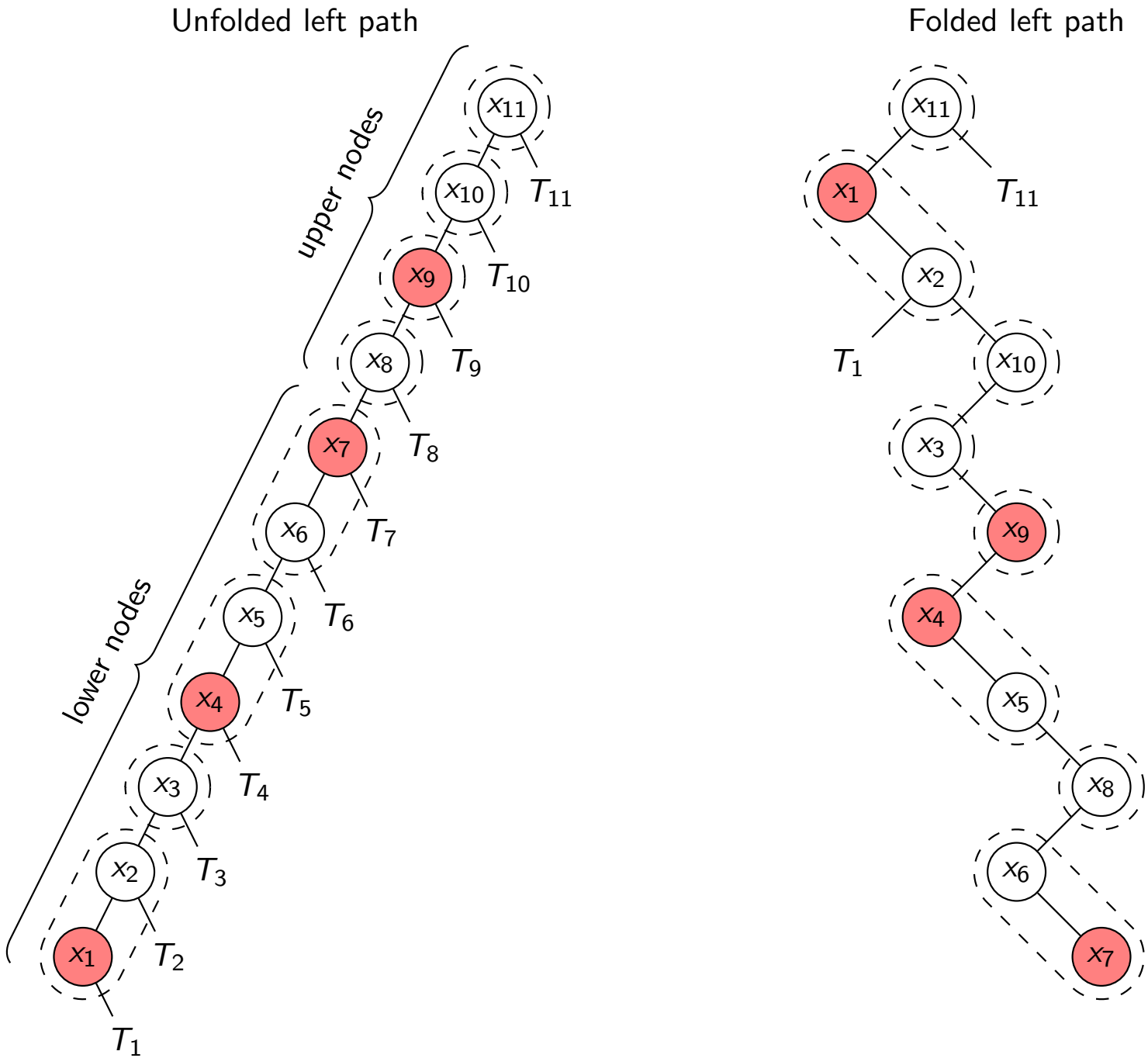
Unfolded left path

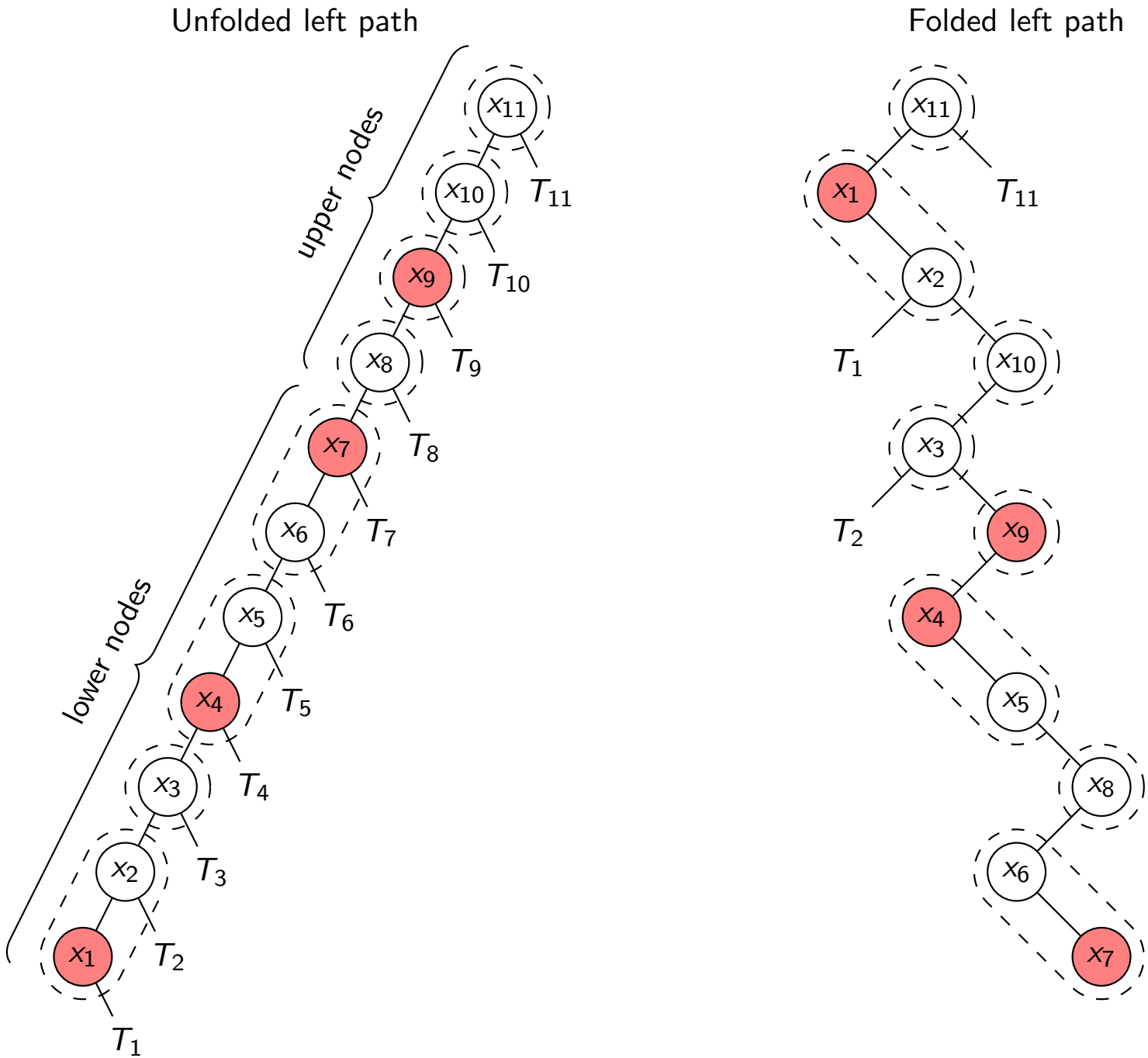


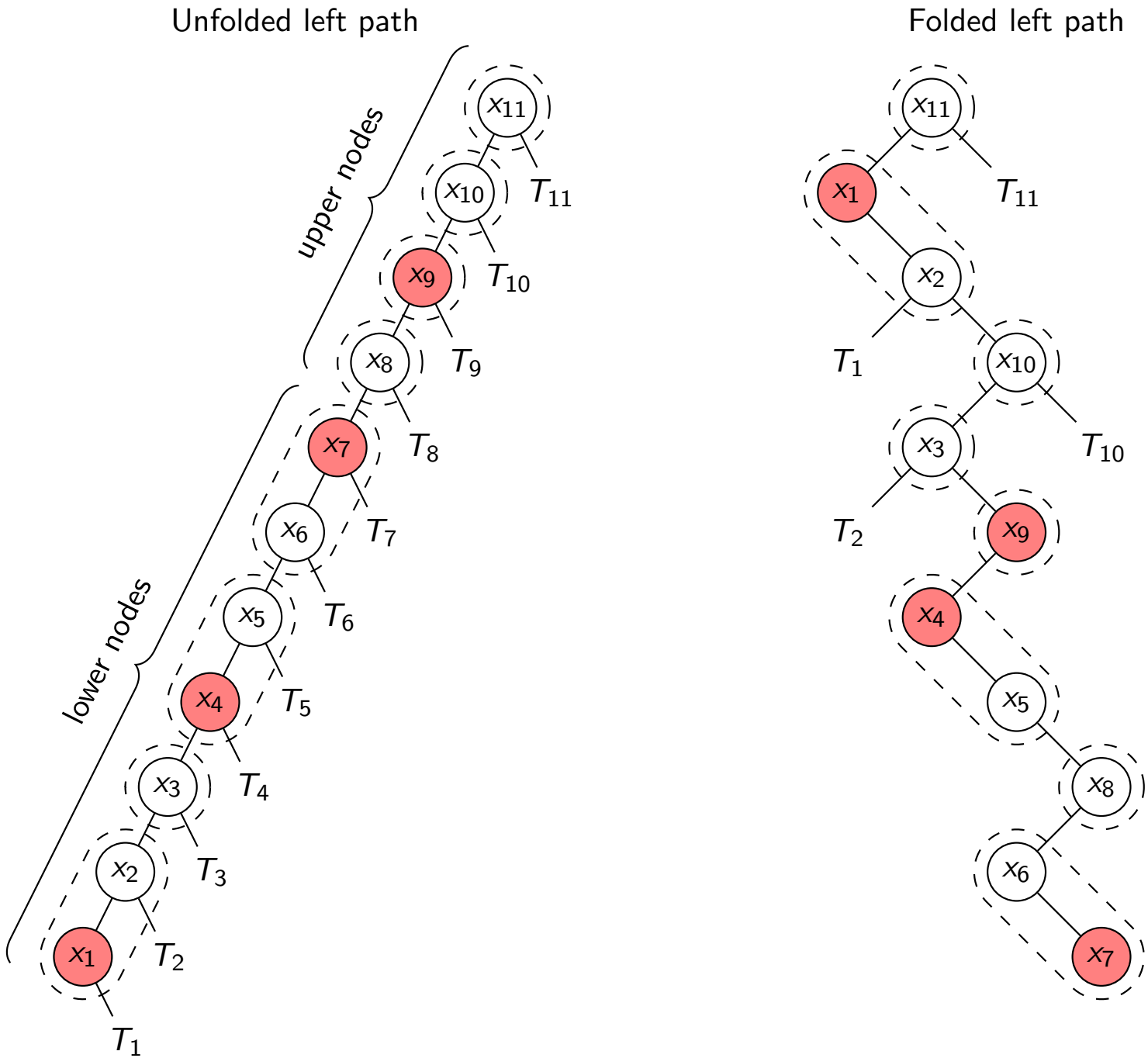
Folded left path

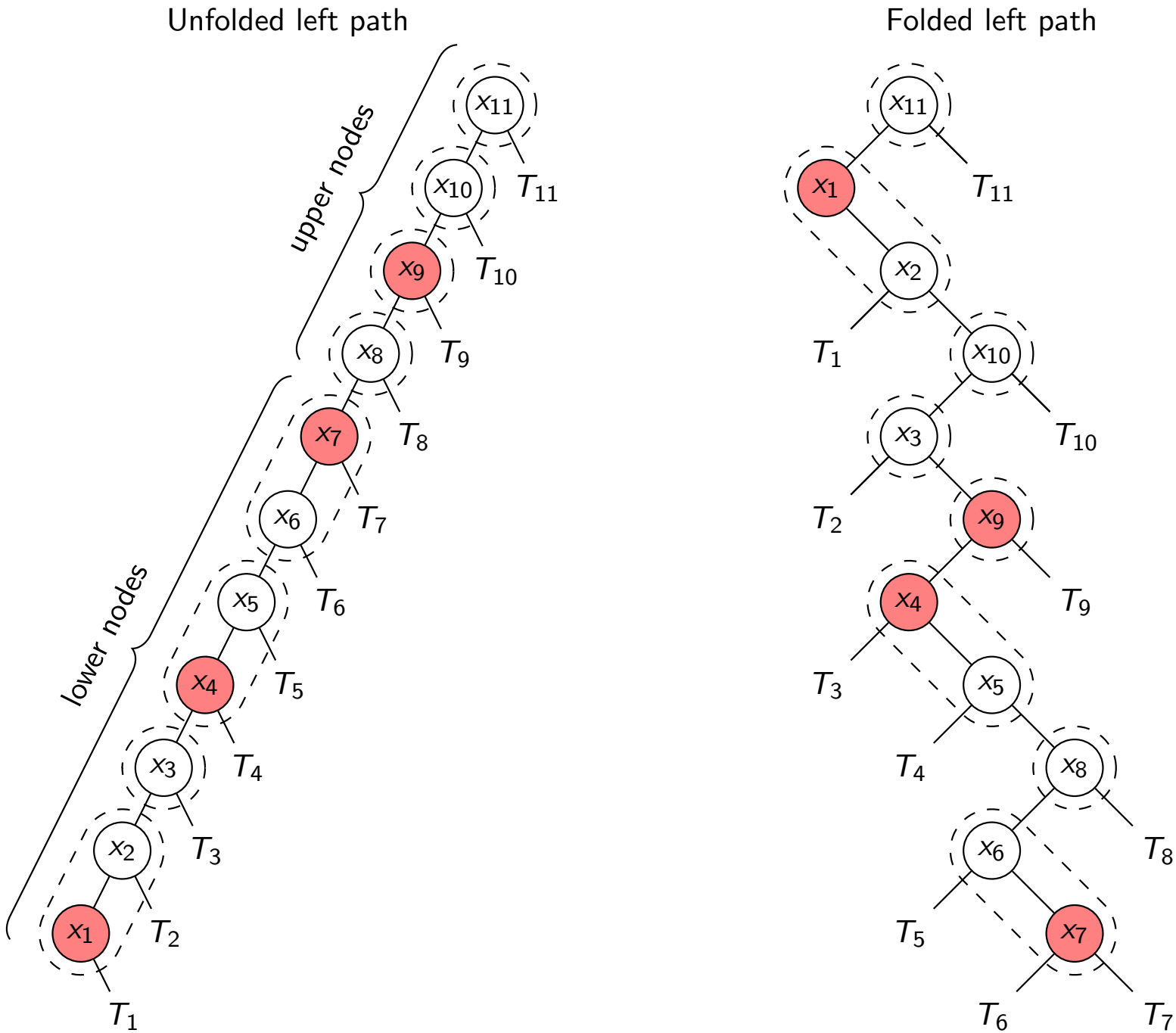


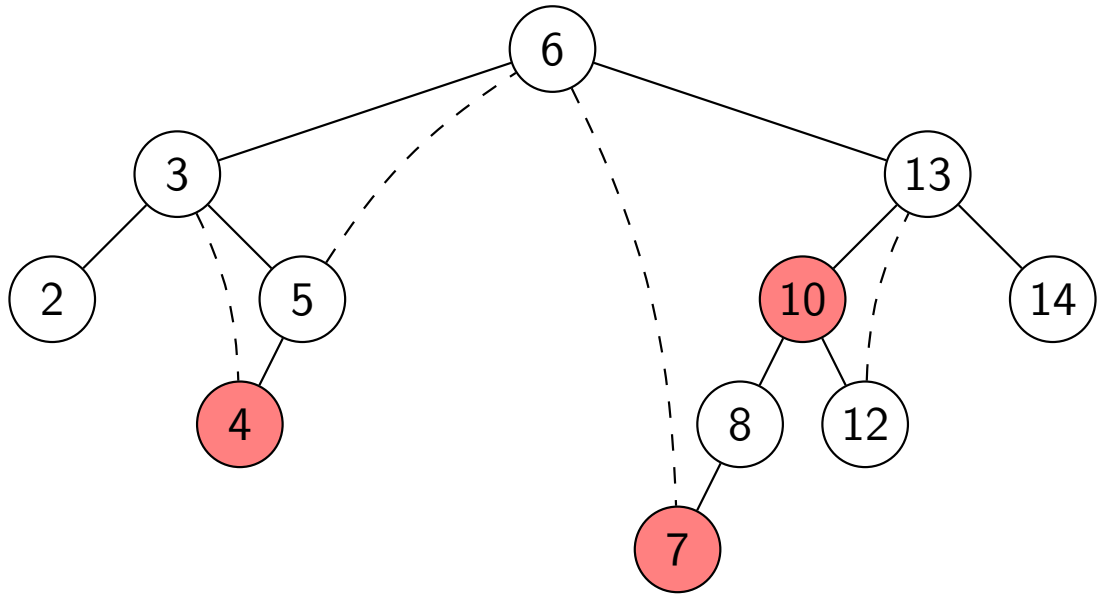

Unfolded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1
Folded left path
x_{11}
x_1
T_{11}
x_2
x_{10}
x_3
x_9
x_4
x_5
x_8
x_6
x_7


Unfolded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1
Folded left path
x_{11}
x_1
T_{11}
x_2
T_1
x_{10}
x_3
x_9
x_4
x_5
x_8
x_6
x_7


Unfolded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1
Folded left path
x_{11}
x_1
T_{11}
x_2
T_1
x_{10}
x_3
T_2
x_9
x_4
x_5
x_8
x_6
x_7

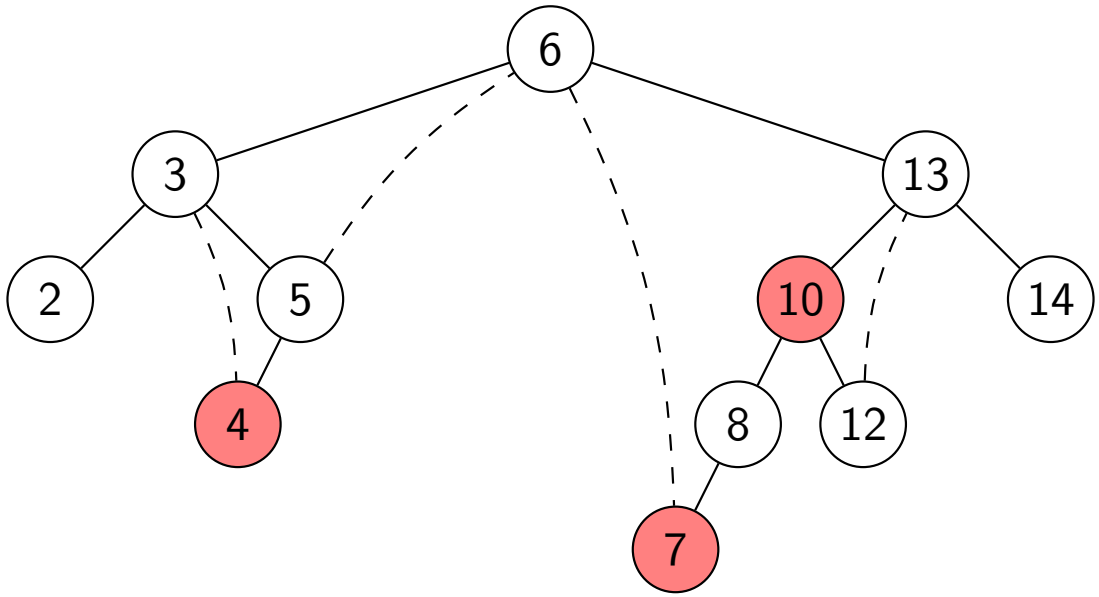

Unfolded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1
Folded left path
x_{11}
x_1
x_2
x_{10}
x_3
x_9
x_4
x_5
x_8
x_6
x_7
T_{11}
T_1
T_{10}
T_2


Unfolded left path
upper nodes
lower nodes
x_{11}
x_{10}
x_9
x_8
x_7
x_6
x_5
x_4
x_3
x_2
x_1
T_{11}
T_{10}
T_9
T_8
T_7
T_6
T_5
T_4
T_3
T_2
T_1
Folded left path
x_{11}
x_1
T_{11}
x_2
T_1
x_{10}
x_3
T_{10}
T_2
x_9
x_4
T_9
T_3
x_5
T_4
x_8
x_6
T_8
T_5
x_7
T_6
T_7



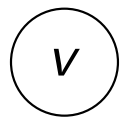
✓ 1. Bit marking red/black

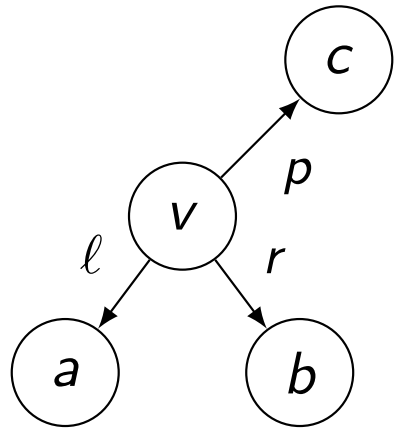
2. Pointers to SUCC and PRED

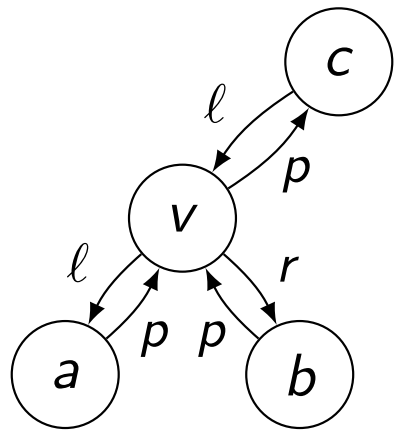


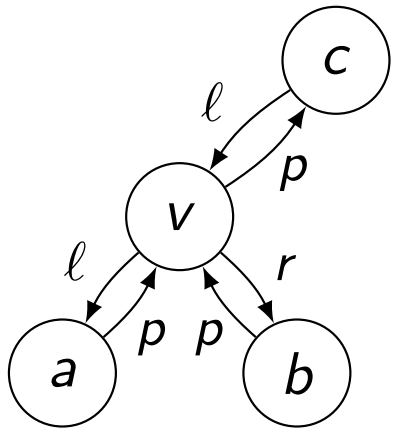
- ✓ 1. Bit marking red/black
- ✓ 2. Pointers to SUCC and PRED

1 key		1 key
5 pointers	$\longrightarrow$	3 pointers
1 bit		0 bits

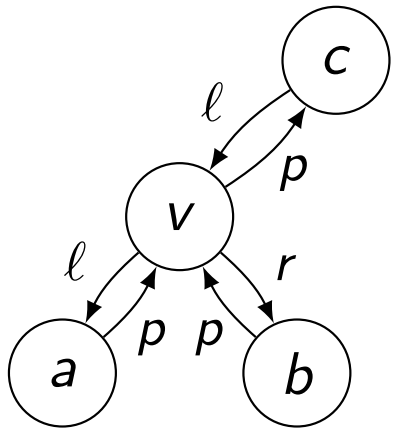

v

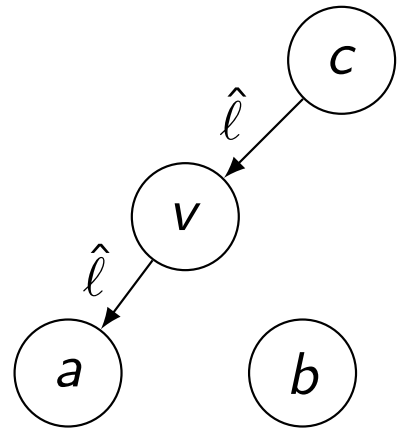

c
p
v
ℓ
r
a
b

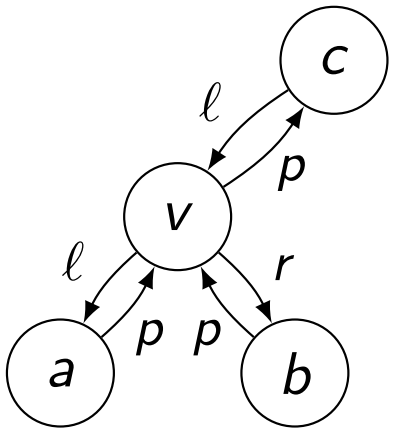

c
ℓ
p
v
ℓ
r
p
p
a
b

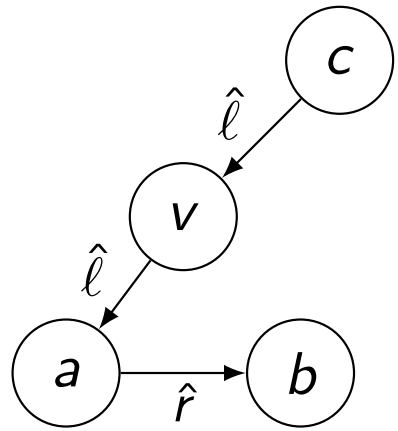

c
ℓ
p
v
ℓ
r
p
p
a
b

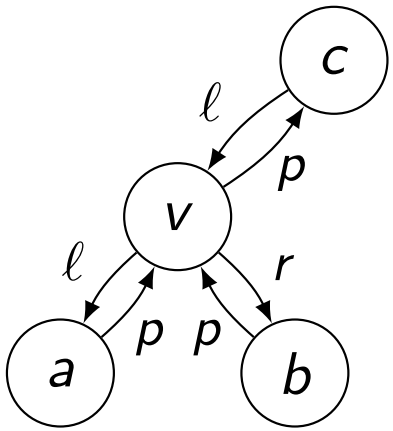

c
ℓ
p
v
ℓ
r
p
p
a
b
c
v
a
b

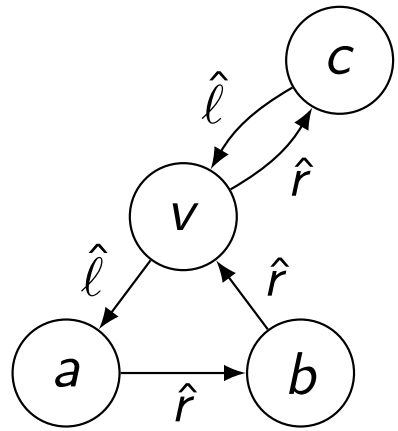

c
ℓ
p
v
ℓ
r
p
p
a
b

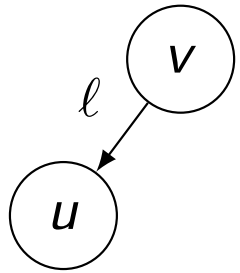

c
ℓ̂
v
ℓ̂
a
b

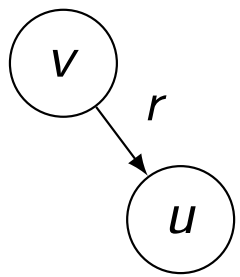

c
ℓ
p
v
ℓ
r
p
p
a
b

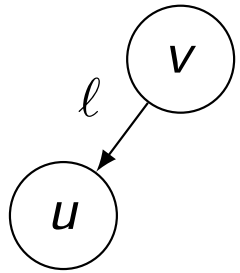

c
ℓ̂
v
ℓ̂
a
r̂
b

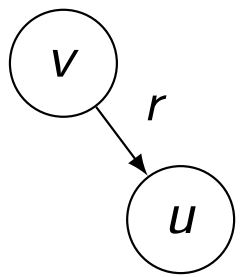

c
ℓ
p
v
ℓ
r
p
p
a
b

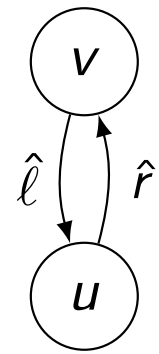

c
ℓ̂
r̂
v
ℓ̂
r̂
a
b
r̂

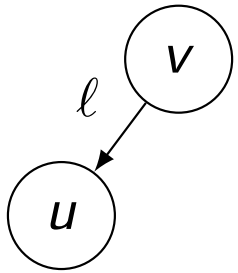

v
ℓ
u

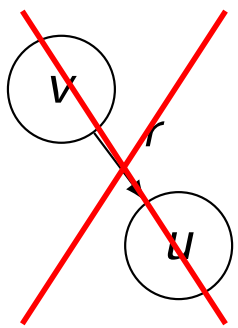

v
r
u

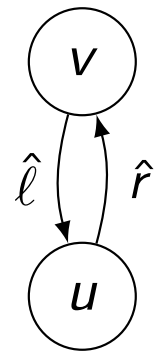

v
ℓ
u

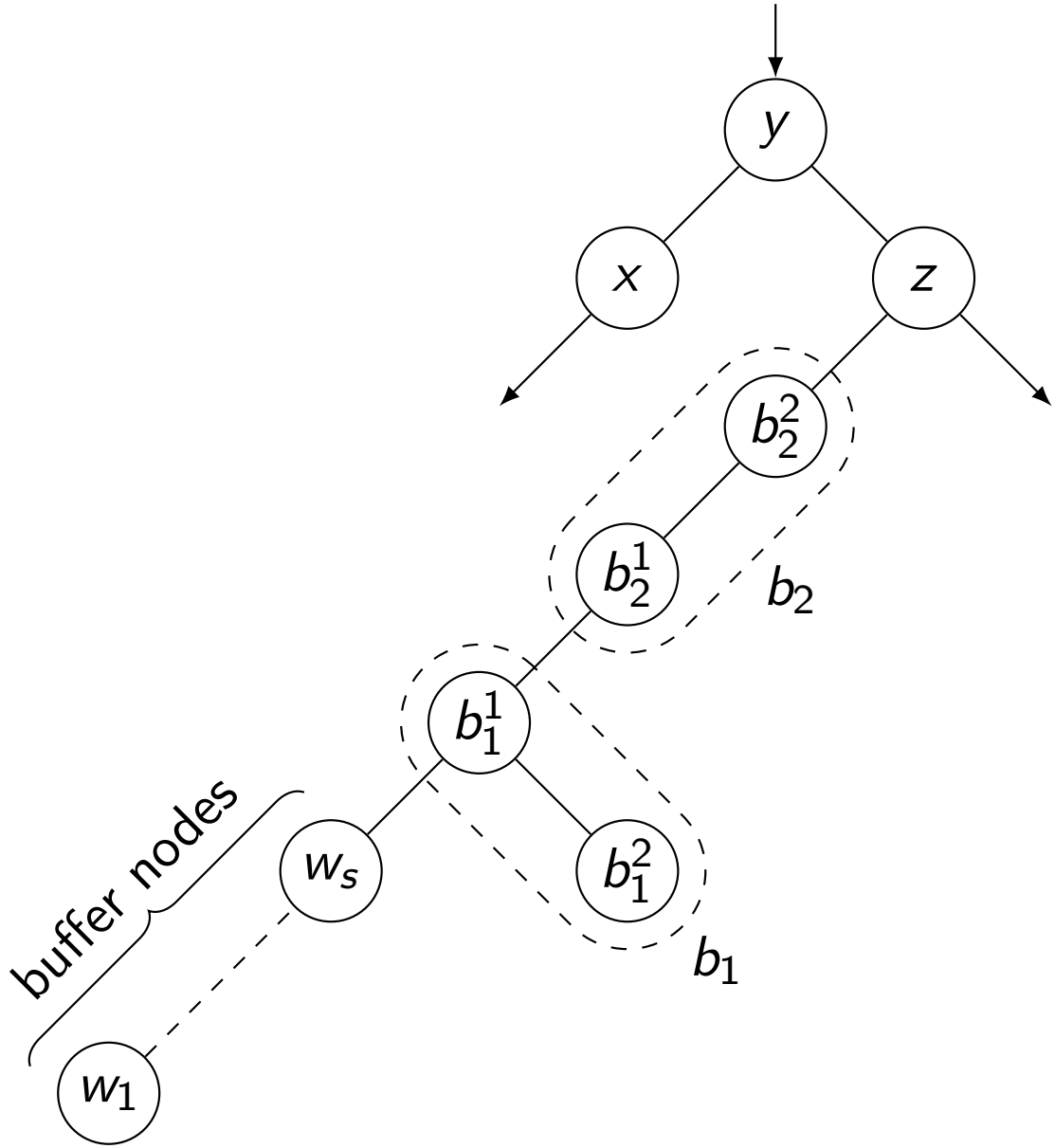

v
r
u


v
ℓ̂
r̂
u


v
ℓ
u


v
r
u


v
ℓ̂
r̂
u

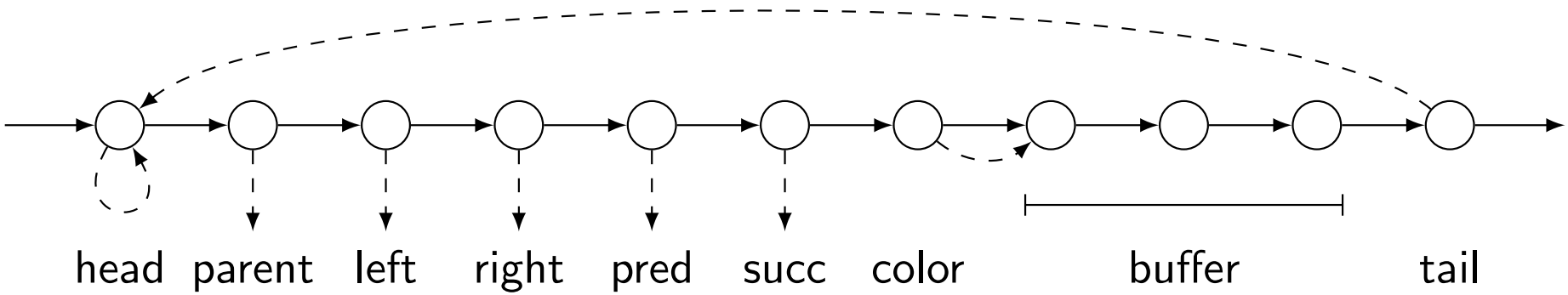

y
x
z
b_2^2
b_2^1
b_2
b_1^1
buffer nodes
w_s
b_1^2
b_1
w_1

1 key		1 key
5 pointers	$\longrightarrow$	3 pointers
1 bit		0 bits

1 key		1 key
5 pointers	$\longrightarrow$	3 pointers
1 bit		0 bits

		1 key
	$\longrightarrow$	2 pointers
		0 bits

	This paper
Succ Pred	$\mathcal{O}(1)$
InsertSucc InsertPred Delete	$\mathcal{O}_A(1)$
FingerSearch	$\mathcal{O}(\lg d)$


head
parent
left
right
pred
succ
color
buffer
tail