”Tallotteri”

En pose indeholder 90 brikker, som er forsynet med tallene 1,2,3,….,90.

Følgende proces gentages indtil der kun er én brik tilbage:

> Vælg to vilkårlige brikker med tallene x og y (x ≥ y), og erstat dem med en ny brik med tallet x – y.

Processen stopper, når der kun er en brik tilbage. Kan der siges noget om tallet på den sidste brik ?

Transition system

Definition 1.3.1 A ***transition system*** S is a pair of the form

$$S = (C, T)$$

where C is the set of ***configurations*** and $T \subseteq C \times C$ is a relation, the ***transition relation***. □

Sequences generated by a transition system

Definition 1.3.3 Let $S = (C, T)$ be a transition system. S ***generates a set of sequences***, $\mathcal{S}(S)$, defined as follows:

1. the finite sequence $c_0, c_1, \ldots, c_n$ (for $n \geq 0$) belongs to $\mathcal{S}(S)$ if
   (ι) $c_0 \in C$
   (ιι) for all i with $1 \leq i \leq n$: $(c_{i-1}, c_i) \in T$
2. the infinite sequence $c_0, c_1, \ldots, c_n, \ldots$ belongs to $\mathcal{S}(S)$ if
   (ι) $c_0 \in C$
   (ιι) for all $i \geq 1$: $(c_{i-1}, c_i) \in T$

□

Processes generated by a transition system

Definition 1.3.5 Let $S = (C, T)$ be a transition system. The set of ***processes generated by*** S, written $\mathcal{P}(S)$, is the subset of $\mathcal{S}(S)$ containing

1. all infinite sequences of $\mathcal{S}(S)$
2. all finite sequences $c_0, c_1, \ldots, c_n$ $(n \geq 0)$ of $\mathcal{S}(S)$ for which it holds that there is no $c \in C$ with $(c_n, c) \in T$.

The final configuration of a finite process is called a ***dead configuration***. □

Transitionssystem: TalLotteri

$S_{TL} = (C_{TL}, T_{TL})$ hvor

$C_{TL} = \{[u,l] | (u \geq 0) \land (l \geq 0)\}$

$T_{TL} = \{([u,l],[u-2,l+1]) | u \geq 2\} \cup$
$\{([u,l],[u,l-1]) | (u \geq 1) \land (l \geq 1)\} \cup$
$\{([u,l],[u,l-1]) | l \geq 2\}$

"Brugervenlig" notation

Transition system TalLotteri

Configurations: $\{[u,l] | (u \geq 0) \land (l \geq 0)\}$

$[u,l] \rhd [u-2,l+1]$ $\boldsymbol{if}$ $u \geq 2$

$[u,l] \rhd [u,l-1]$ $\boldsymbol{if}$ $u,l \geq 1$

$[u,l] \rhd [u,l-1]$ $\boldsymbol{if}$ $l \geq 2$

Football

Transition system Football

Configurations: $\{[t, X, a, b] \mid 0 \leq t \leq 90, X \in \{A, B, R\}, a, b \in \mathbf{N}\}$

$$
\begin{array}{lcll}
[t, A, a, b] & \rhd & [t+2, B, a, b] & \textbf{if } t \leq 88 \\
[t, A, a, b] & \rhd & [t+2, B, a+1, b] & \textbf{if } t \leq 88 \\
[t, A, a, b] & \rhd & [t+1, B, a, b] & \textbf{if } t \leq 89 \\
[t, A, a, b] & \rhd & [t+1, B, a+1, b] & \textbf{if } t \leq 89 \\
[90, A, a, b] & \rhd & [90, R, a, b] & \\
[t, B, a, b] & \rhd & [t+2, A, a, b] & \textbf{if } t \leq 88 \\
[t, B, a, b] & \rhd & [t+2, A, a, b+1] & \textbf{if } t \leq 88 \\
[t, B, a, b] & \rhd & [t+1, A, a, b] & \textbf{if } t \leq 89 \\
[t, B, a, b] & \rhd & [t+1, A, a, b+1] & \textbf{if } t \leq 89 \\
[90, B, a, b] & \rhd & [90, R, a, b] &
\end{array}
$$

Induction principle

Induction principle Let $P(0), P(1), \ldots, P(n), \ldots$ be statements. If

a) $P(0)$ is true

b) for all $n \geq 0$ it holds that $P(n)$ implies $P(n+1)$,

then $P(n)$ is true for all $n \geq 0$.

Invariance principle

Invariance principle for transition systems Let $S = (C, T)$ be a transition system and let $c_0 \in C$ be a configuration. If $I(c)$ is a statement about the configurations of the system, the following holds. If

a) $I(c_0)$ is true

b) for all $(c, c') \in T$ it holds that $I(c)$ implies $I(c')$

then $I(c)$ is true for any configuration c that occurs in a sequence starting with c_0.

Termination principle

Termination principle for transition systems Let $S = (C, T)$ be a transition system and let $\mu : C \to \mathbf{N}$ be a function. If

$$\text{for all } (c, c') \in T \text{ it holds that } \mu(c) > \mu(c')$$

then all processes in $\mathcal{P}(S)$ are finite.

Nim

Transition system Nim

Configurations: $\{A, B\} \times \mathbf{N}$

$$
\begin{array}{llll}
[A, n] & \rhd & [B, n-2] & \textbf{if } n \geq 2 \\
[A, n] & \rhd & [B, n-1] & \textbf{if } n \geq 1 \\
[B, n] & \rhd & [A, n-2] & \textbf{if } n \geq 2 \\
[B, n] & \rhd & [A, n-1] & \textbf{if } n \geq 1
\end{array}
$$

Towers of Hanoi

Transition system $\mathsf{Hanoi}(n)$

Configurations: $\{[A,B,C] \mid \{A,B,C\}$ a partition of $\{1,\ldots,n\}\}$

$$
\begin{array}{lll}
[A,B,C] & \triangleright & [A\setminus\{r\}, B\cup\{r\}, C] \ \ \textbf{if}\ (r=\min A)\wedge(r<\min B) \\
[A,B,C] & \triangleright & [A\setminus\{r\}, B, C\cup\{r\}] \ \ \textbf{if}\ (r=\min A)\wedge(r<\min C) \\
[A,B,C] & \triangleright & [A\cup\{r\}, B\setminus\{r\}, C] \ \ \textbf{if}\ (r=\min B)\wedge(r<\min A) \\
[A,B,C] & \triangleright & [A, B\setminus\{r\}, C\cup\{r\}] \ \ \textbf{if}\ (r=\min B)\wedge(r<\min C) \\
[A,B,C] & \triangleright & [A\cup\{r\}, B, C\setminus\{r\}] \ \ \textbf{if}\ (r=\min C)\wedge(r<\min A) \\
[A,B,C] & \triangleright & [A, B\cup\{r\}, C\setminus\{r\}] \ \ \textbf{if}\ (r=\min C)\wedge(r<\min B)
\end{array}
$$

Transitionssystem Counting

Configurations: $\{[i,j] \mid i,j \geq 1\}$

$$[i,j] \triangleright [i,2j] \quad \textbf{if} \quad j \leq 10$$
$$[i,j] \triangleright [i-1,j-10] \quad \textbf{if} \quad j > 10 \ \wedge \ i > 1$$

dADS1 eksamensopgave august 2006, opgave 20

Når startkonfigurationen er [10,10], hvilken af følgende er ***ikke*** en invariant for transitionssystemet ?

a) $1 \leq j \leq 10$
b) $1 \leq i \leq 10$
c) $10 \leq j \leq 20$
d) $j \geq i$
e) $20\,i - j < 200$
f) ved ikke

Transitionssystem Counting

Configurations: $\{[i,j] \mid i,j \geq 1\}$

$[i,j] \triangleright [i,2j] \quad \textbf{if} \quad j \leq 10$

$[i,j] \triangleright [i-1,j-10] \quad \textbf{if} \quad j > 10 \quad \wedge \quad i > 1$

dADS1 eksamensopgave august 2006, opgave 20

Når startkonfigurationen er [10,10], hvilken af nedenstående er en termineringsfunktion for transistionssystemet ?

a) $\mu(i, j) = i$
b) $\mu(i, j) = i - j$
c) $\mu(i, j) = 20\, i - j$
d) $\mu(i, j) = i^2 - j$
e) $\mu(i, j) = j - i$
f) ved ikke

Euclid's algorithm

Transition system Euclid

Configurations: $\{[m, n] \mid m, n \geq 1\}$

$[m, n] \;\triangleright\; [m - n, n] \quad \textbf{if } m > n$

$[m, n] \;\triangleright\; [m, n - m] \quad \textbf{if } m < n$

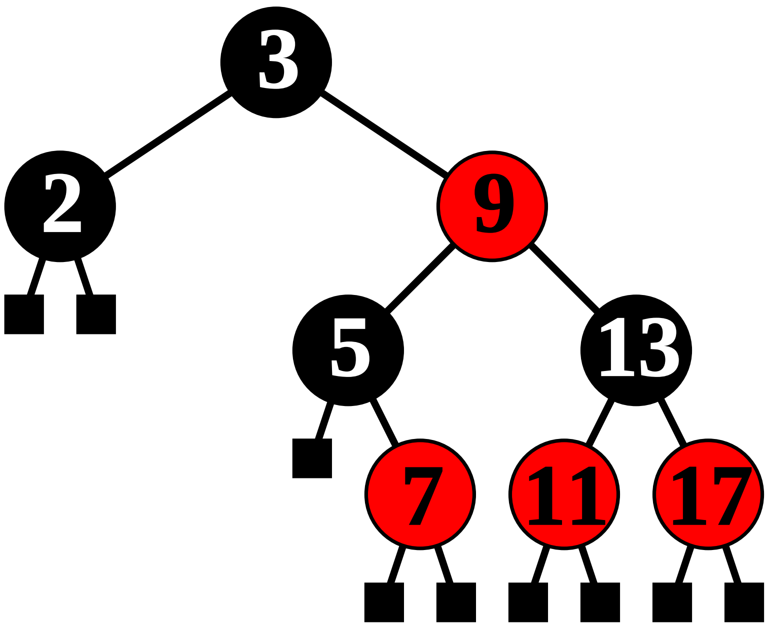
Red-black tree

Definition 1.5.7 A ***red-black tree*** is binary search tree in which all internal nodes are colored either red or black, in which the leaves are black, and

Invariant I_2 Each red node has a black parent.

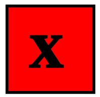
Invariant I_3 There is the same number of black nodes on all paths from the root to a leaf.

□



Insertion

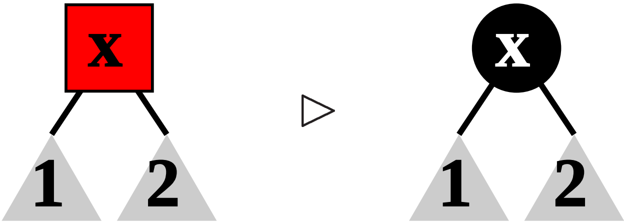
Illegitimate red node:



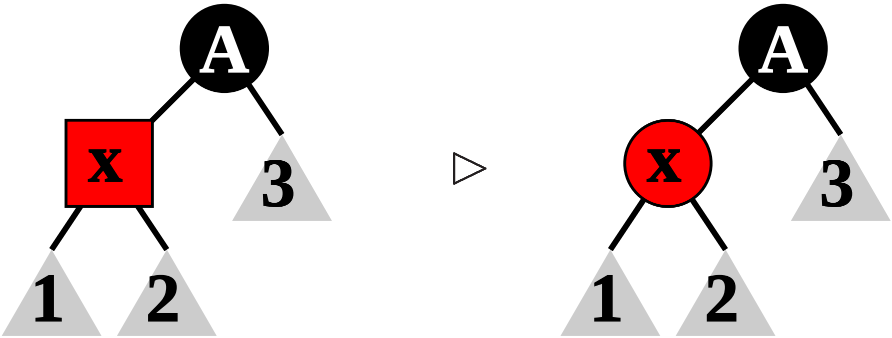
Invariant I_2'**:** Each *legitimate* red node has a black parent.

Insertion: transitions 1 and 2

The illegitimate node is the root of the tree:

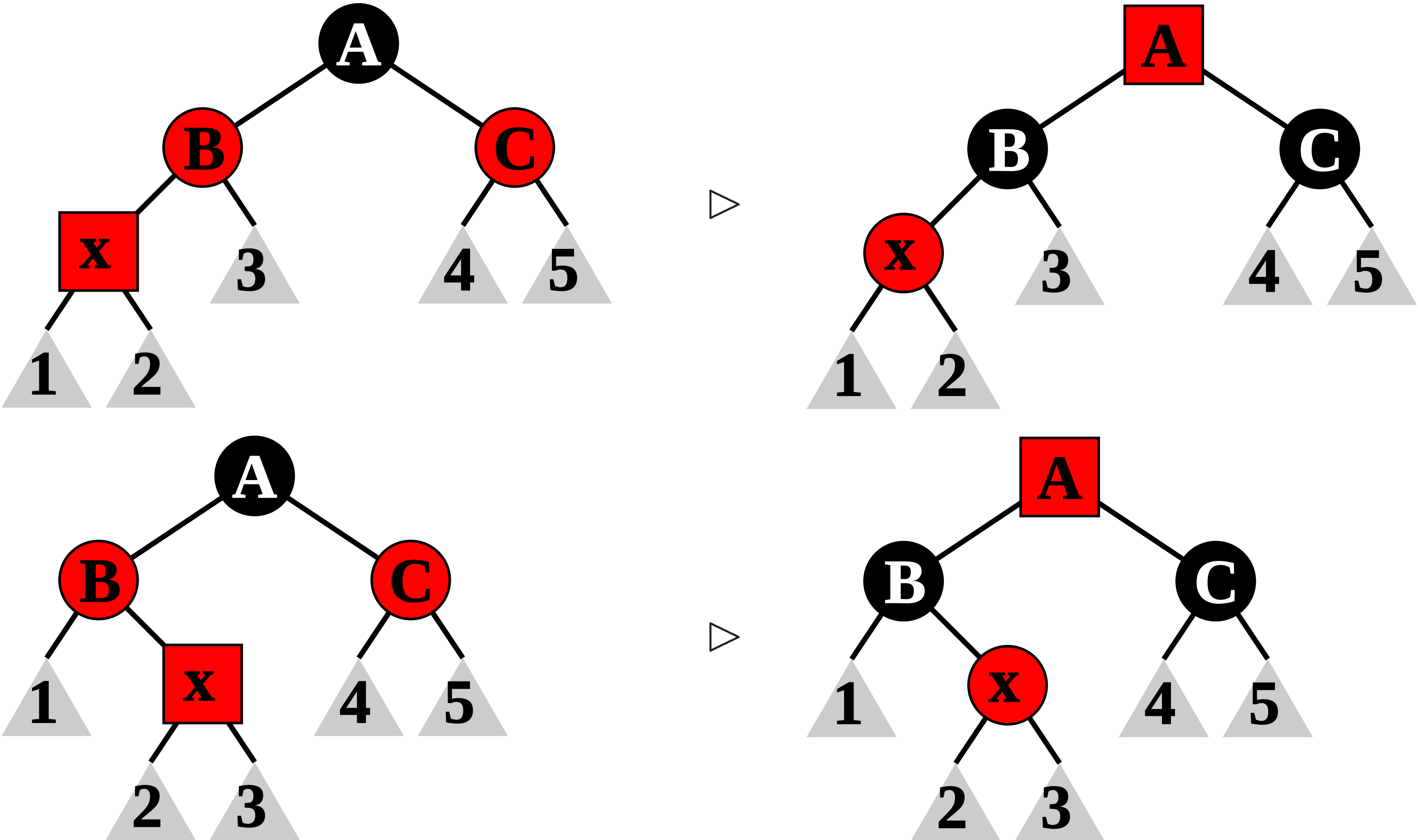


The illegitimate node has a black father:



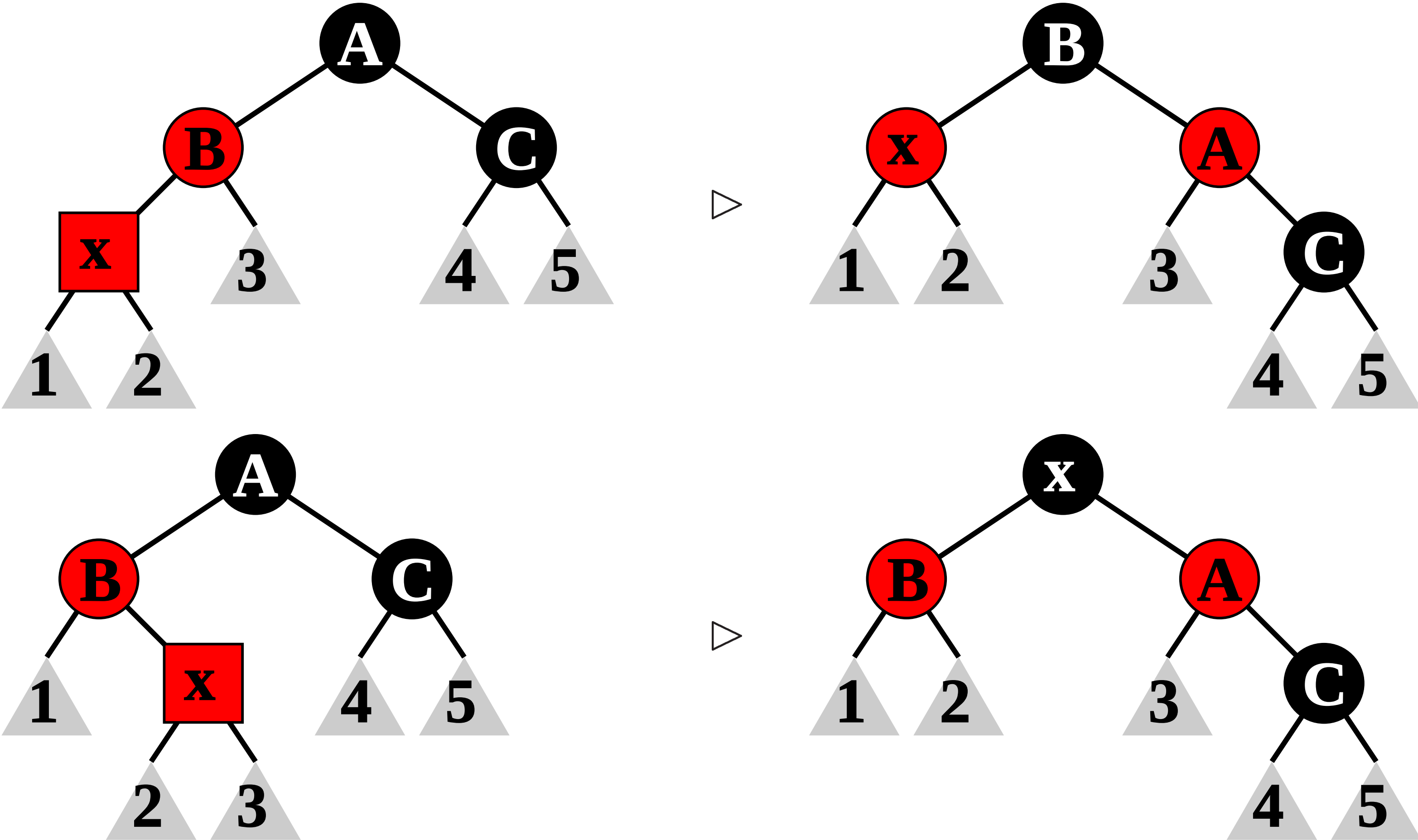
Insertion: transitions 3.1 and 3.2

The illegitimate node has a red father and a red uncle:



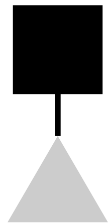
Insertion: transitions 4.1 and 4.2

The illegitimate node has a red father and a black uncle:



Deletion

Illegitimate black node:

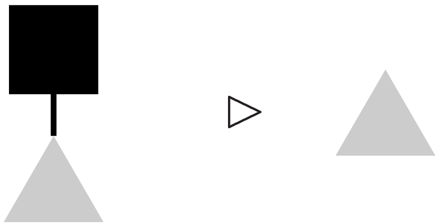


Invariant I_1' The tree satisfies I_1 if we remove the illegitimate node.

Invariant I_2' Each red node has a *legitimate* black father.

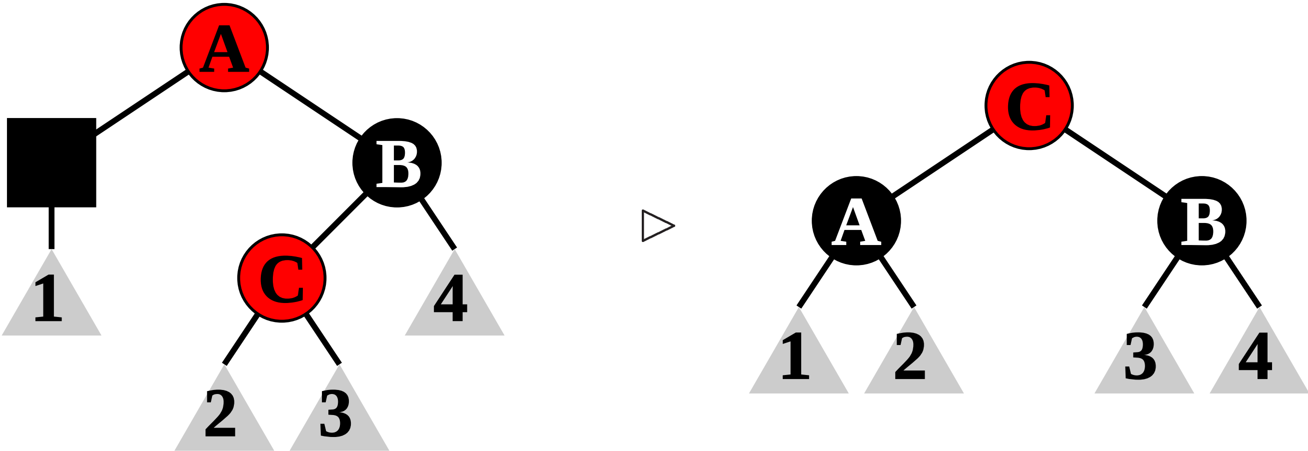
Deletion: transition 1

The illegitimate node is the root:

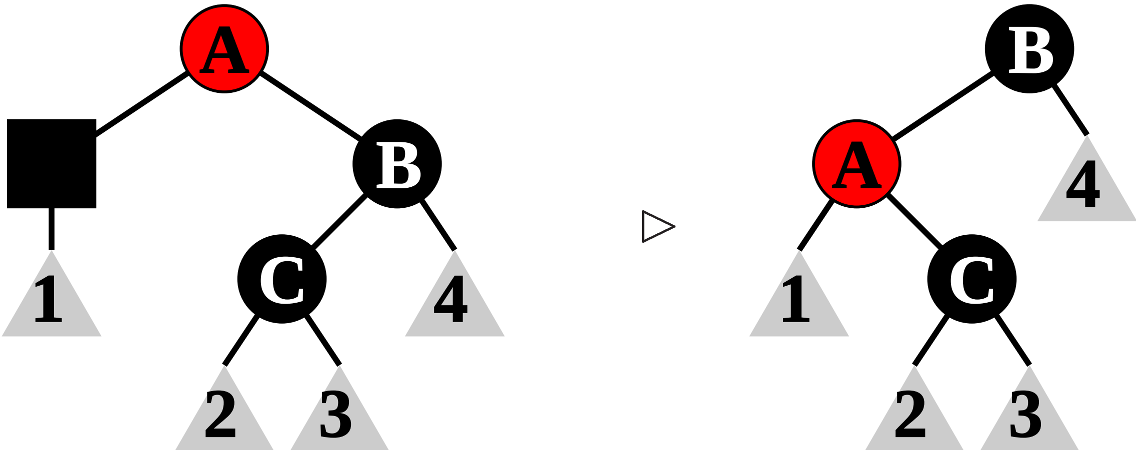


Deletion: transitions 2 and 3

The illegitimate node has a red father and a red closer nephew:

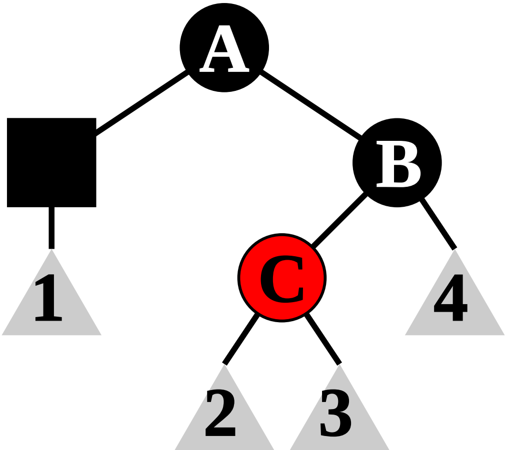


The illegitimate node has a red father and a black closer nephew:

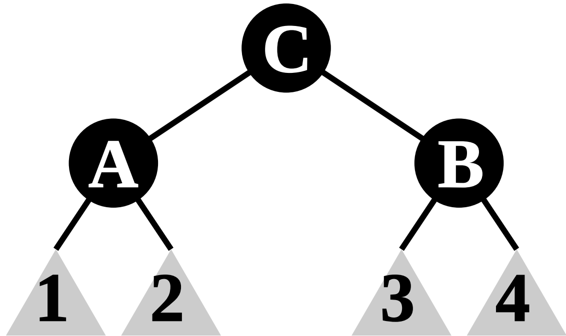


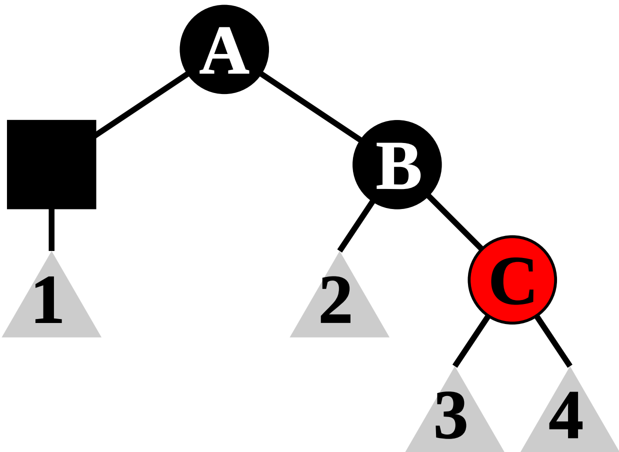
Deletion: transitions 4.1 and 4.2

The illegitimate node has a black father, a black sibling and one red nephew:

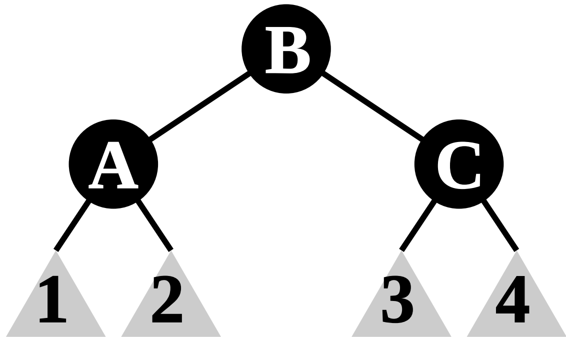


▷



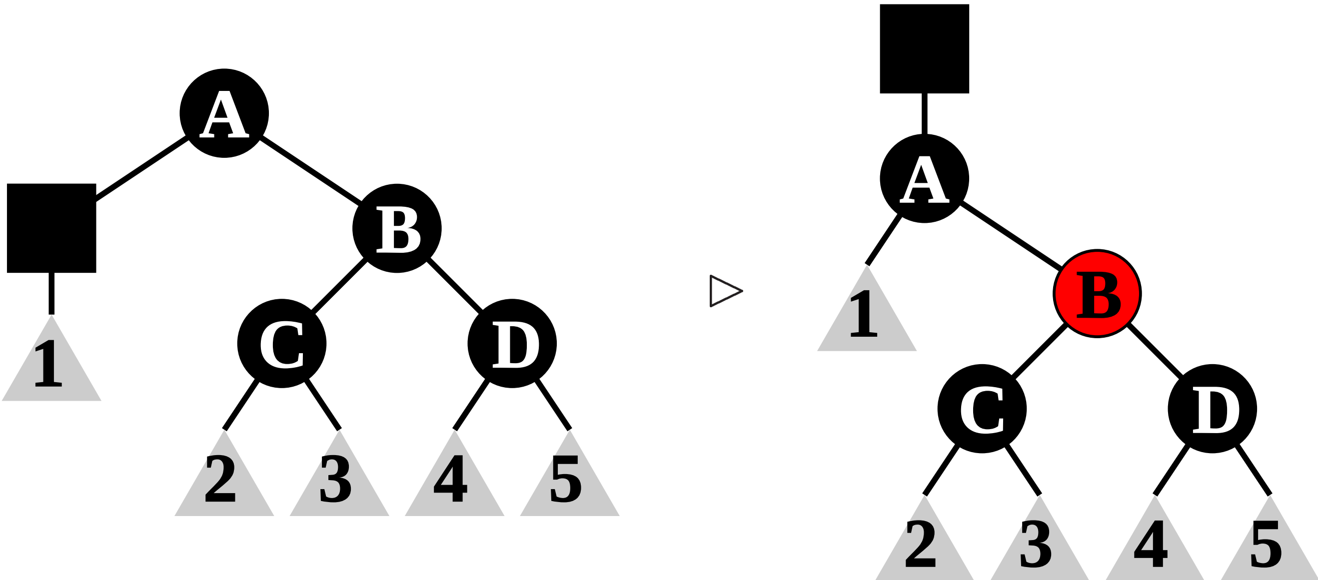


▷



Deletion: transition 5

The illegitimate node has a black father, a black sibling and two black nephews



Deletion: transition 6

The illegitimate node has a black father and a red sibling: