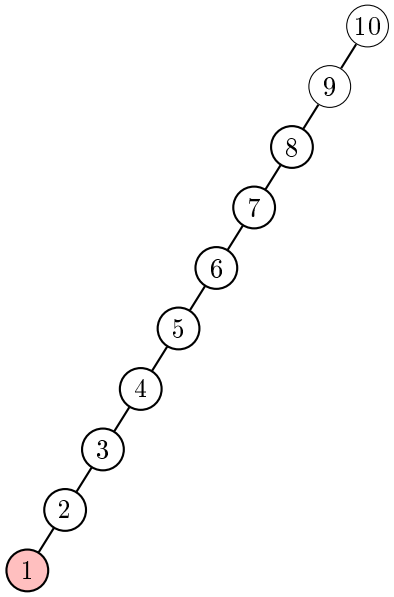
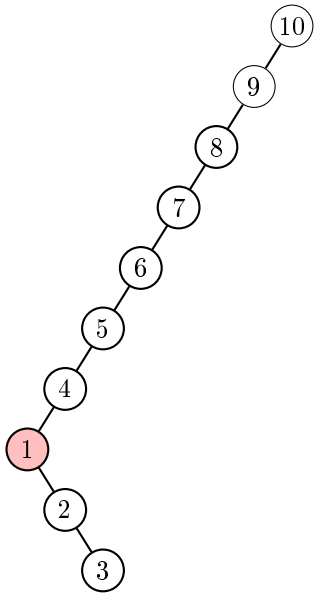
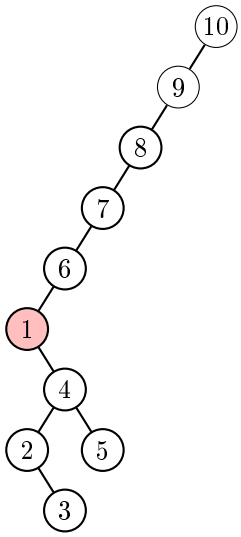
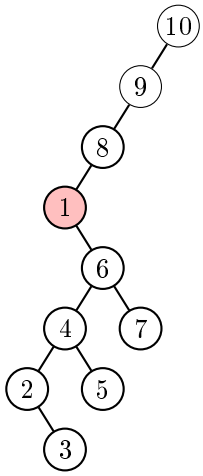
Example for using binTreeDL2Ti*k*Z

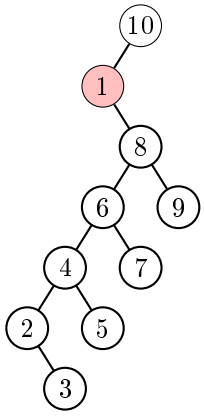
A splay tree example from 'Kozen' (Example 12.1, p. 60)

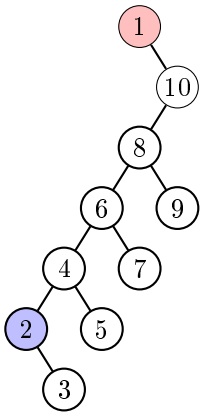


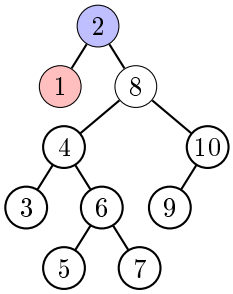



10
9
8
7
6
1
4
2
5
3


10
9
8
1
6
4
7
2
5
3


10
1
8
6
9
4
7
2
5
3


1
10
8
6
9
4
7
2
5
3


2
1
8
4
10
3
6
9
5
7

Instructions for the binTree-Description-Language

Instructions for tree representation

`new_tree`	Generate a new binary tree (current tree will be lost).
`draw_tree`	Will draw current tree (tikzpicture).

`highlight_red_node(<number>)`
`highlight_green_node(<number>)`
`highlight_blue_node(<number>)`
	Three instructions to highlight a node permanently.

`dont_highlight_node(<number>)`	Don't highlight node `<number>`.
`dont_highlight_all`	Don't highlight all nodes in tree.

Instructions for binary trees

`add_node(<number>)`	Add the node `<number>` in current tree. Multiple nodes with same number are allowed.
`remove_node(<number>)`	Remove one node with `<number>` from tree (replaced by his inorder successor).

Instructions for splay trees

`ST_splay_step(<number>)`	Apply on splay step for `<number>` on current tree (Zero, one or two rotations will be performed). Node with `<number>` need not be an element of the tree!
`ST_splay(<number>)`	As many splay steps as needed will be apllied. Node with `<number>`, inorder predecessor or inorder successor will be the root after this instruction.
`ST_member(<number>)`	Synonym for `ST_splay(<number>)` instruction.
`ST_insert(<number>)`	Add the node `<number>` in current tree under the rules for splay trees.
`ST_delete(<number>)`	Remove one node with `<number>` from tree under the rules for splay trees.