1. 添加

第一种情况：

A

P

NIL

NIL

或

=

替换

插入到黑色结点，直接插入就行

第二种情况：

插入到红色结点，其父结点必定为黑色，并且父结点的另一子结点要么为黑色NIL要么为红色

NIL

A

L

NIL

替换

P

NIL

或

=

此时，先将L涂为黑色，分情况

1. A父结点的另一子结点为红色结点

经过P的左子树路径多了一个黑色结点，因此把P涂成红色，但经过P的右子树路径就少了一个黑色结点，因此把R涂成黑色，然后再把P当做新插入结点往上依次递归返回操作

NIL

A

L

P

R

1. 父结点的另一子结点为黑色（NIL）

经过P的左子树路径多了一个黑色结点，因此把P涂成红色，但经过P的右子树路径就少了一个黑色结点，因此需要进行旋转操作

NIL

A

L

P

NIL

如果A插入到的是L的左子结点，则进行右单转操作，如果A插入到的是L的左子结点，则进行左右单转操作（另一侧同理）

A

L

P

NIL

NIL

1. 删除

实际是删除叶子结点问题（删除非叶子结点可以（根据左右子树高度，用前继结点或）后继结点的值替换被删结点值，然后简化为删除叶子结点问题）

第一种情况：

删除到红色结点，直接删除就行

L

P

第二种情况：

删除到黑色结点，父结点为红色，经过P右子树的路径必有一个黑色结点，经过P左子树的路径少一个黑色结点

或

或

L

P

R

NIL

NIL

1. 如果A结点为红色，则进行右左双转

或

P

R

NIL

A

或

L

P

R

NIL

A

P

或

L

R

NIL

A

然后将P涂黑

P

或

L

R

NIL

A

B

如果B结点是黑色（NIL）也可将R涂红，A涂黑

1. 如果A结点为黑色（NIL），则进行左单转

或

L

R

NIL

P

或

L

R

NIL

P

B

如果B结点为黑色（NIL），也可不旋转将P涂黑，R涂红

第三种情况：

删除到黑色结点，父结点为黑色，经过P左子树的路径少一个黑色结点

或

或

L

P

R

NIL

NIL

或

R

NIL

NIL

NIL

NIL

或

或

或

或

1. 如果R为红色，只需把一个结点给过去填补另一侧的黑色结点，操作很多举个例子：

A

B

P

R

1. 如果A子结点全为黑色（NIL），左单转并将A涂红，R涂黑

R

P

A

1. A右子结点为红色，右左双转，将C涂黑

A

R

或

C

P

1. A右子结点为黑色（NIL），右左双转

A

R

或

然后再左单转

A

或

R

R

如果B子结点全为黑也可以将B涂红，R涂黑

A

R

或

B

1. 如果R为黑色，子结点非全黑（NIL）

A

B

P

R

如果A结点为红色，则右左双转将A涂黑

P

或

L

R

NIL

A

如果B结点红色，则左单转将B涂黑

P

或

L

B

NIL

R

如果AB结点同时为红色，可右左双转将A涂黑，也可左单转将B涂黑

1. 如果R为黑色，子结点全黑（NIL），先对P树进行内部红黑平衡，将R涂红，然后往上递归（经过P的路径黑色结点少一个）

A

B

P

R