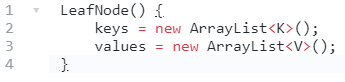
# DD路径分析

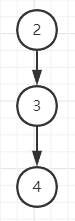
## LeafNode

1. LeafNode

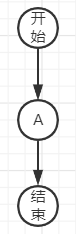
程序代码：



程序流图：



DD路径：

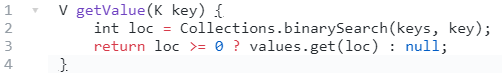


测试用例：

1.1 TestLeafNode

1. getValue

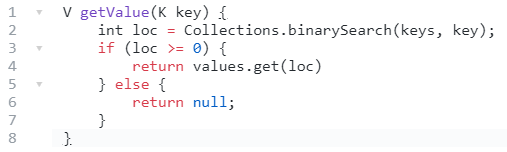
程序代码：



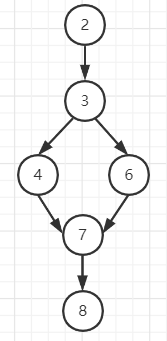
由于代码中有三元表达式，我们将三元表达式拆分成if-then-else的格式。

此外，假定java的API一定返回正确的值，因此第一行代码我们认为是一条语句。

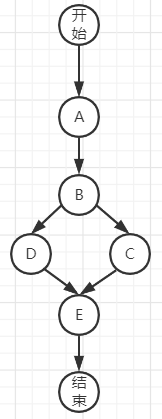
拆分后的程序代码：



程序流图：

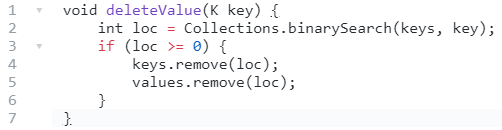


DD路径：

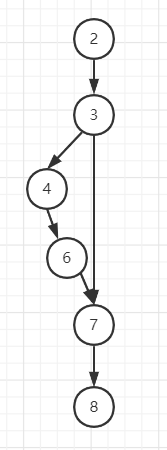


1. deleteValue

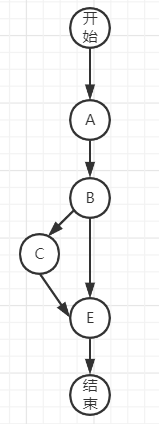
程序代码



程序流图：

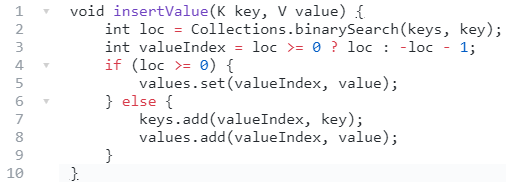


DD路径：

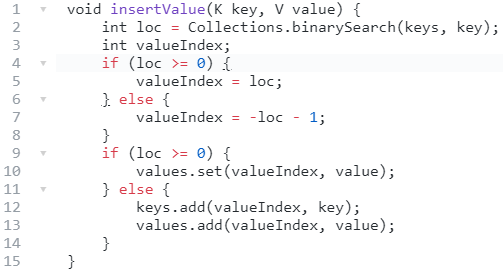


1. insertValue

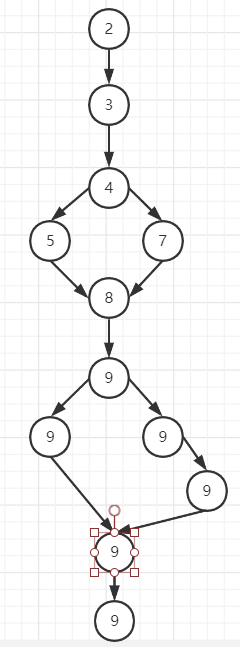
程序代码：



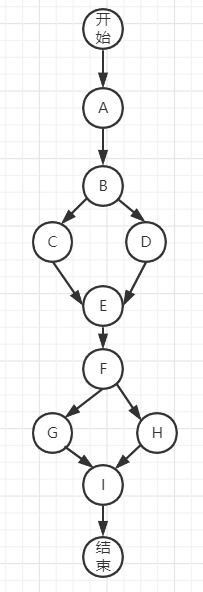
将三目表达式改成if-then-else的形式，代码如下



程序流图：

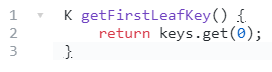


DD路径：

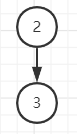


1. getFirstLeafKey

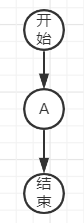
程序代码：



程序流图：

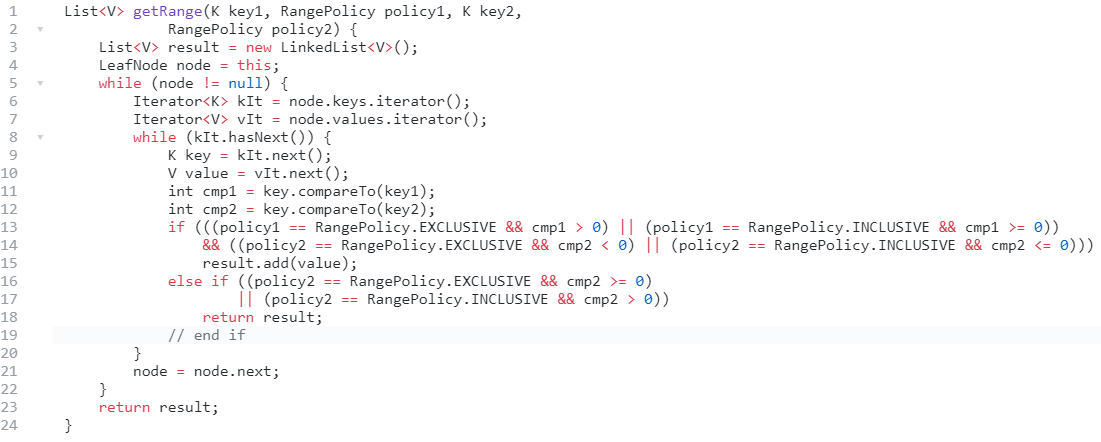


DD路径：

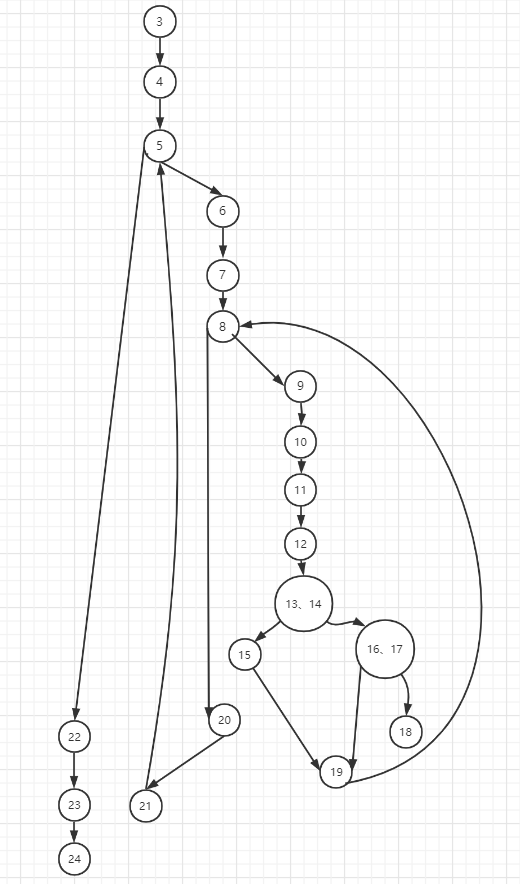


1. getRange

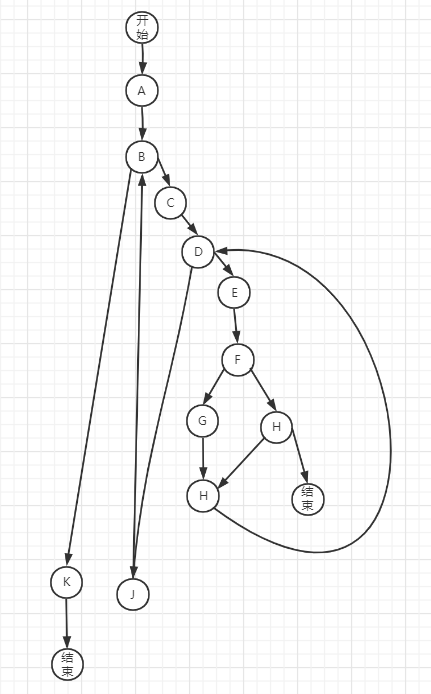
程序代码：



程序流图：

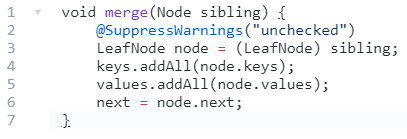


DD路径：

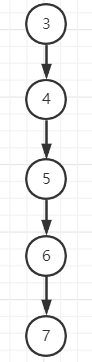


1. merge

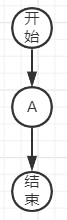
程序代码：



程序流图：

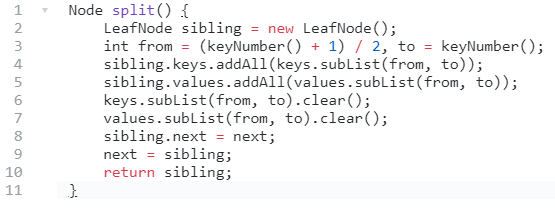


DD路径：

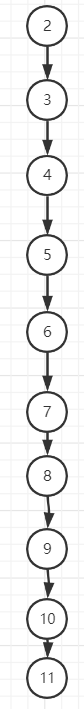


1. split

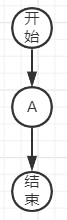
程序代码：



程序流图：

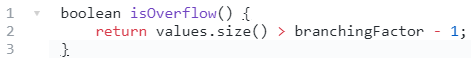


DD路径：

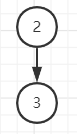


1. isOverflow

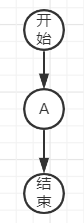
程序代码：



程序流图：

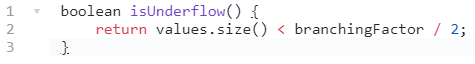


DD路径：

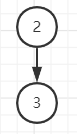


1. isUnderflow

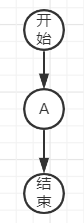
程序代码：



程序流图：



DD路径：



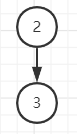
## BPlusTree

1. BPlusTree

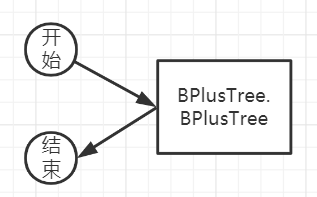
程序代码：



程序流图：

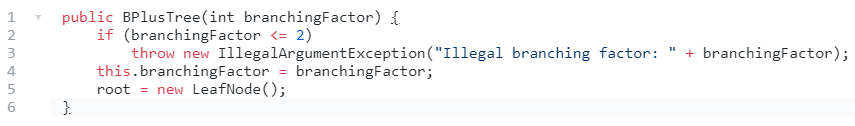


DD路径：

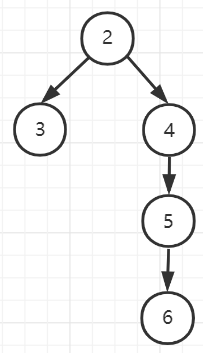


1. BPlusTree

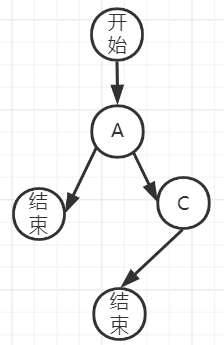
程序代码：



程序流图：



DD路径：

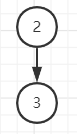


1. Search

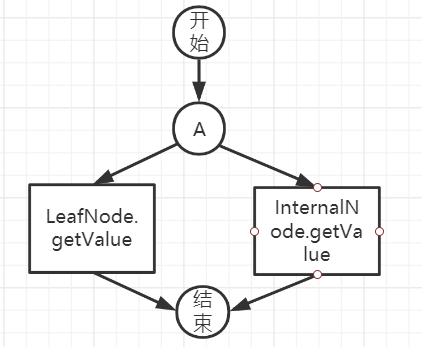
程序代码：



程序流图：

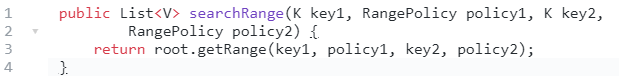


DD路径：

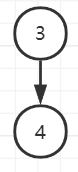


1. SearchRange

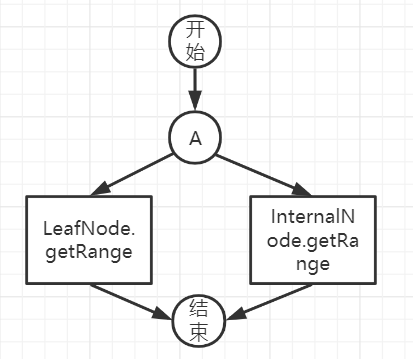
程序代码：



程序流图：

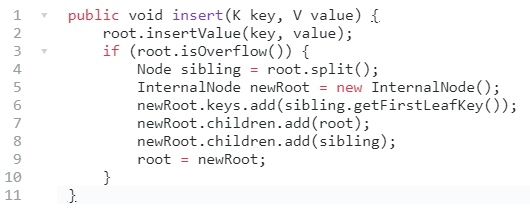


DD路径：

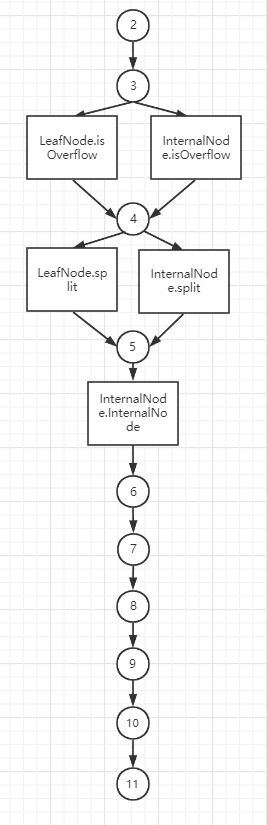


1. insert

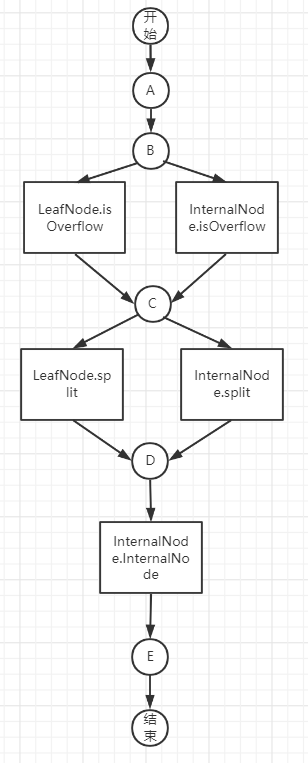
程序代码：



程序流图：

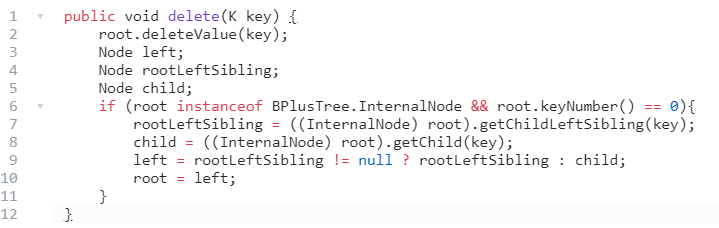


DD路径：

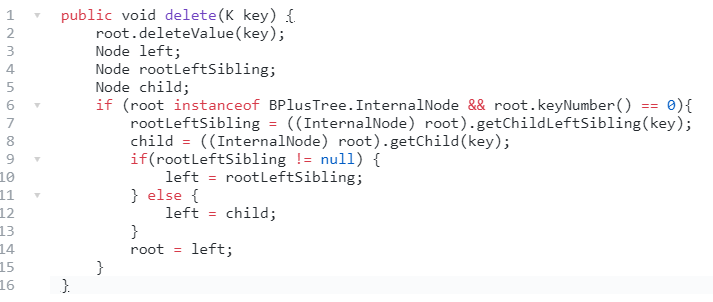


1. delete

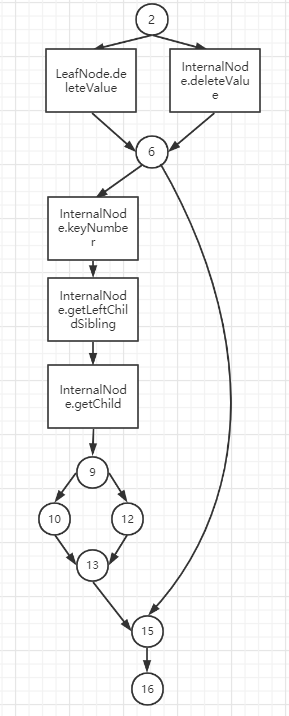
程序代码：



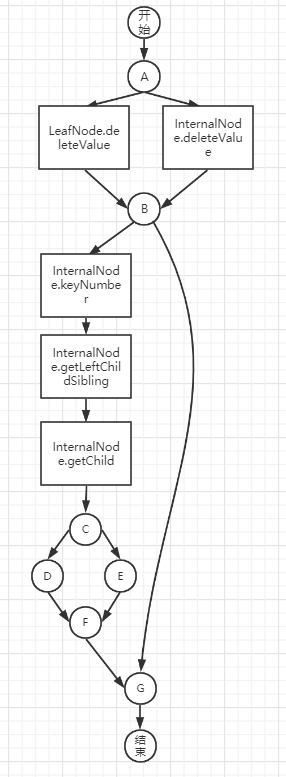
重写三目表达式之后的程序代码：



程序流图：



DD路径：

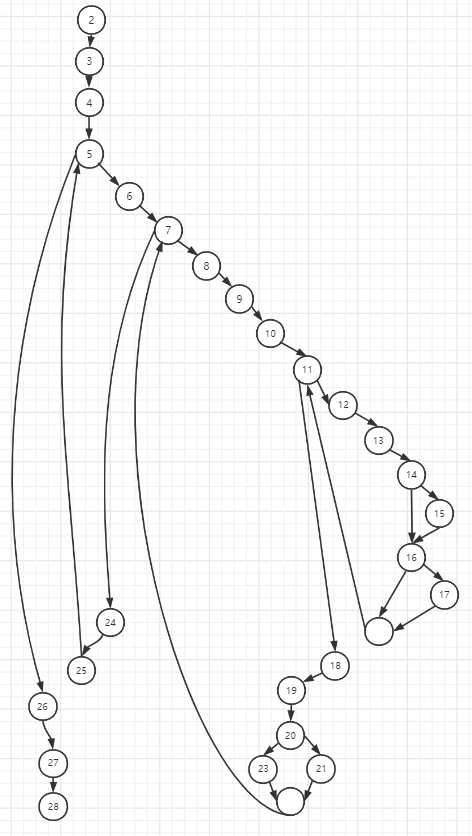


1. toString

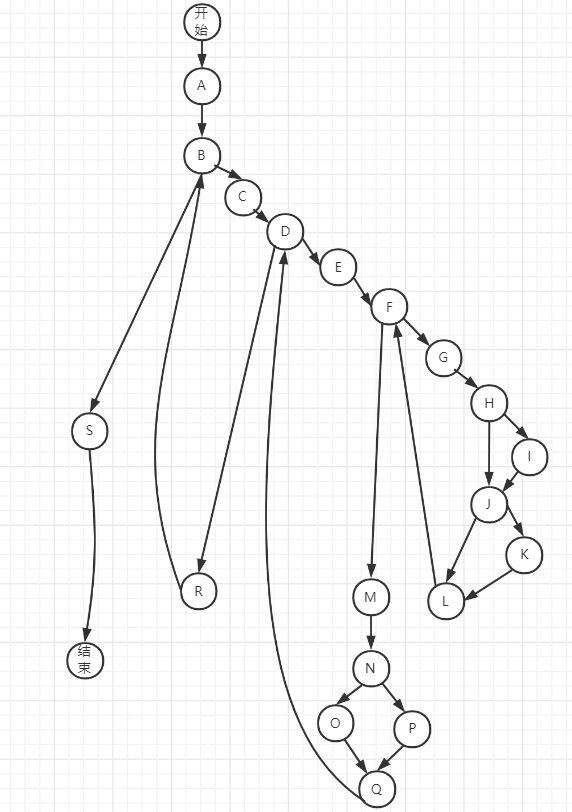
程序代码：



程序流图：



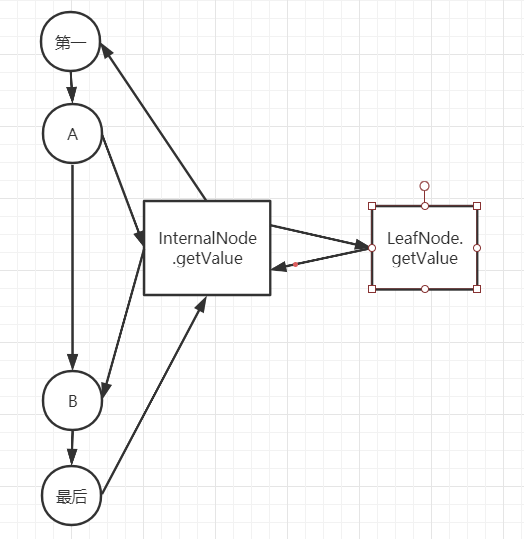
DD路径：



## InternalNode

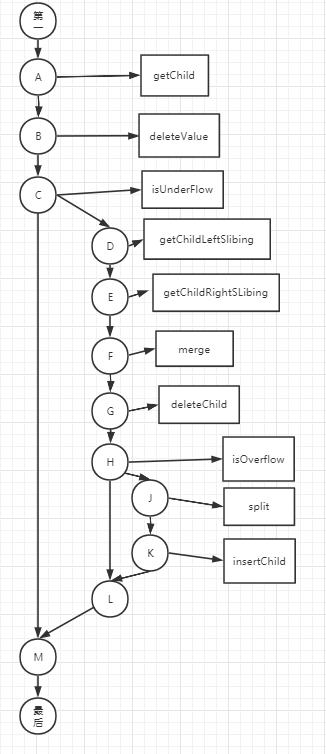
1. getValue

DD路径：



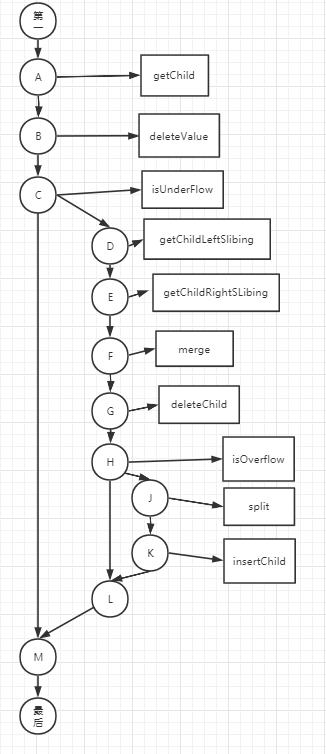
1. deleteValue

DD路径：



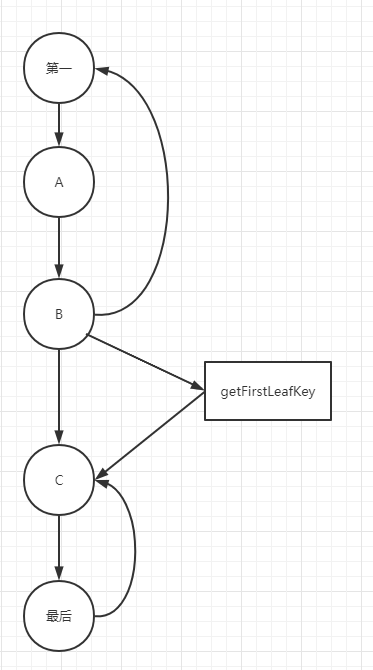
1. insertValue

DD路径：



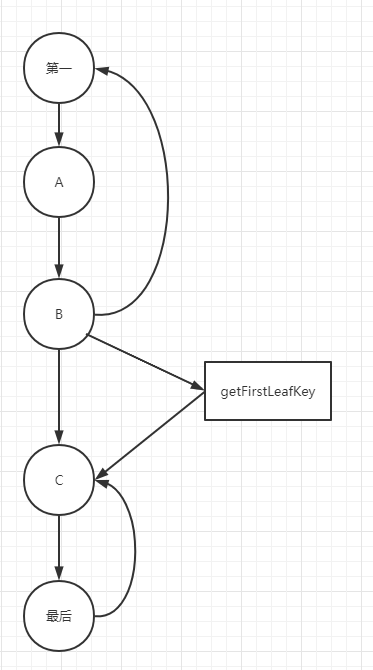
1. getFirstLeafKey

DD路径：



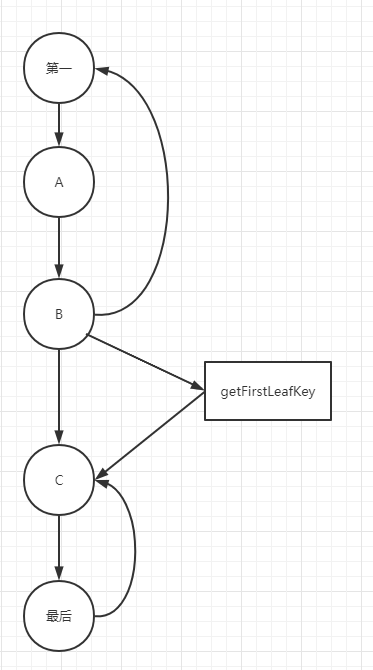
1. getRange

DD路径：



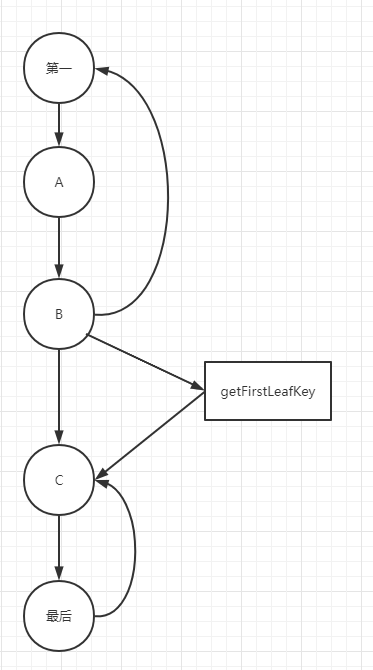
1. merge

DD路径：



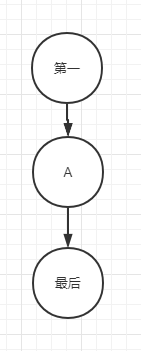
1. split

DD路径：



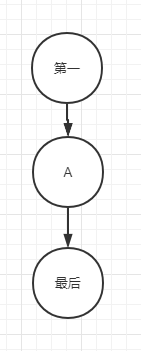
1. isOverflow

DD路径：



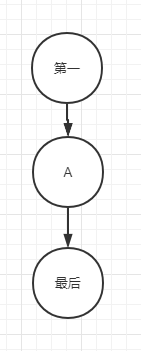
1. isUnderflow

DD路径：



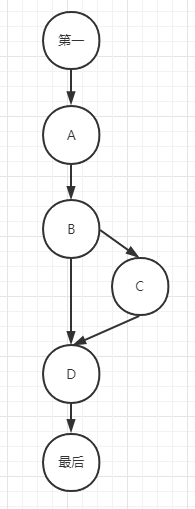
1. getChild

DD路径：



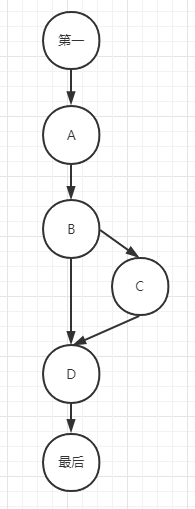
1. deleteChild

DD路径：



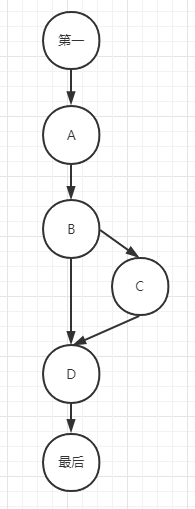
1. insertChild

DD路径：



1. getChildLeftSibling

DD路径：



1. getChildRightSibling

DD路径：

