```
if (removed == null) {
                        «bind»
                                                      RemoveTermVisitor
                      T -> Object
                                           -removed: LambdaTerm = null
LambdaTermVisitor<T>
                                          +RemoveTermVisitor()
                                          +visit(node: LambdaRoot) (
                                          +visit(node: LambdaApplication)
                                          +visit(node: LambdaAbstraction)
                                          +visit(node: LambdaVariable)
```

```
} else {
  node.setChild(null);
if (removed == null) {
  assert(node.getParent() != null);
  removed = node;
  node.getParent().accept(this);
 else {
  if (node.getFirst() == removed) {
     node.setFirst(null);
   } else if (node.getSecond() == removed) {
     node.setSecond(null);
```

// Cant happen