研讨会

关于自动化匹配注释与代码的学习

周桂安MF1832267

程序切片工具——WALA

√ 构建调用图和系统依赖图然后实现程序切片

wala切片的关键点在找到seed statement，官网例子的seed语句是在public static void main（String args []）函数中调用的println（）函数，类似的下面第一个seed statement是在contains中调用的indexOf方法；第二个例子的statement是return index

- statement: NORMAL getMax:6 = binaryop(add) 3 , 5 Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere----------80statement: NORMAL getMax:conditional branch(ge, to iindex=14) 1,2 Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere----------83

- statement: NORMAL getMax:9 = binaryop(add) 1 , 5 Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere----------84statement: NORMAL getMax:goto (from iindex= 13 to iindex = 19) Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere----------86

- statement: NORMAL getMax:conditional branch(eq, to iindex=19) 2,7 Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere--------86

- statement: NORMAL getMax:11 = binaryop(add) 2 , 5 Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere----------91statement: NORMAL getMax:return Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere----------93

- statement: PARAM_CALLEE:Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere v1

- statement: PARAM_CALLEE:Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere v2

- statement: PARAM_CALLEE:Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere v3

- statement: EXC_RET_CALLEE:Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere

- statement: METHOD_ENTRY:Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere

- statement: METHOD_EXIT:Node: < Application, Lpaixu/w, getMax(III)V > Context: Everywhere

```java
 319        * @return {@code true} if this list contains the specified element
 320        */
 321⊖    public boolean contains(Object o) {
 322        return indexOf(o) != -1;
 323    }
 324

 600⊖    public int indexOf(Object o) {
 601        int index = 0;
 602        if (o == null) {
 603            for (Node<E> x = first; x != null; x = x.next) {
 604                if (x.item == null)
 605                    return index;
 606                index++;
 607            }
 608        } else {
 609            for (Node<E> x = first; x != null; x = x.next) {
 610                if (o.equals(x.item))
 611                    return index;
 612                index++;
 613            }
 614        }
 615        return -1;
 616    }
```

ForWardSilece

```
method:contains, class:Llist/LinkedList
Source line number = 322
method:contains, class:Llist/LinkedList
Source line number = 322
method:contains, class:Llist/LinkedList
Source line number = 322
method:contains, class:Llist/LinkedList
Source line number = 322
method:indexOf, class:Llist/LinkedList
Source line number = 602
method:indexOf, class:Llist/LinkedList
Source line number = 603
method:indexOf, class:Llist/LinkedList
Source line number = 604
method:indexOf, class:Llist/LinkedList
Source line number = 604
method:indexOf, class:Llist/LinkedList
Source line number = 605
method:indexOf, class:Llist/LinkedList
Source line number = 606
method:indexOf, class:Llist/LinkedList
Source line number = 603
method:indexOf, class:Llist/LinkedList
Source line number = 603
method:indexOf, class:Llist/LinkedList
Source line number = 608
method:indexOf, class:Llist/LinkedList
Source line number = 609
method:indexOf, class:Llist/LinkedList
Source line number = 609
method:indexOf, class:Llist/LinkedList
Source line number = 610
method:indexOf, class:Llist/LinkedList
Source line number = 610
method:indexOf, class:Llist/LinkedList
Source line number = 610
method:indexOf, class:Llist/LinkedList
Source line number = 611
method:indexOf, class:Llist/LinkedList
Source line number = 612
method:indexOf, class:Llist/LinkedList
Source line number = 609
method:indexOf, class:Llist/LinkedList
Source line number = 609
method:indexOf, class:Llist/LinkedList
Source line number = 615
method:equals, class:Ljava/lang/Object
Source line number = 149
method:equals, class:Ljava/lang/Object
```

BackwardSlice

```java
600⊖    public int indexOf(Object o) {
601         int index = 0;
602         if (o == null) {
603             for (Node<E> x = first; x != null; x = x.next) {
604                 if (x.item == null)
605                     return index;
606                 index++;
607             }
608         } else {
609             for (Node<E> x = first; x != null; x = x.next) {
610                 if (o.equals(x.item))
611                     return index;
612                 index++;
613             }
614         }
615         return -1;
616     }
617
```

```
method:indexOf, class:Llist/LinkedList
Source line number = 610
method:indexOf, class:Llist/LinkedList
Source line number = 611
method:indexOf, class:Llist/LinkedList
Source line number = 612
method:indexOf, class:Llist/LinkedList
Source line number = 612
method:indexOf, class:Llist/LinkedList
Source line number = 613
method:indexOf, class:Llist/LinkedList
Source line number = 614
method:indexOf, class:Llist/LinkedList
Source line number = 611
method:indexOf, class:Llist/LinkedList
Source line number = 611
method:main, class:Llist/LinkedList
Source line number = 89
method:main, class:Llist/LinkedList
Source line number = 89
method:main, class:Llist/LinkedList
Source line number = 96
```

下周计划

- 继续阅读学习机器学习和自然语言处理相关

- 完成某些类的slice任务，并和老师汇报讨论下一步task