# 作业二：程序分析测试工具实现报告

何欣程 DZ1833010

### 功能描述

实现对于动态语言JavaScript的智能补全工具，主要功能为：当程序员输入表达式”e.”时，该工具能够对”.”后需要补全的属性进行推荐。

对于静态语言，e后面的属性可以较为容易地从e的类型推断出，但由于JavaScript的特殊性，其具有以下四个难点：

1. 由于JavaScript是动态语言，所以较难通过静态分析手段推断e的可能类型，同时，在代码执行的任何位置，e都可以动态地增加或删除一些属性。
2. 很多JavaScript基于框架代码如jQuery实现，因此若要对其做动态分析的话代价十分巨大。
3. 很多JavaScript会调用一些本地库如Document Object Model (DOM)，它由浏览器提供，不能轻易分析。
4. 智能补全工具应该能在代码编写的过程中就能实时提供推荐功能，在这种情况下可能就无法利用调用关系等分析手段。

## 需求分析

1. 工具输入：需要进行代码补全的源码文件，可以是JavaScript文件或含有<script>标签的html文件，将文件中的JavaScript代码提取出来进行分析。在输入代码中，需要进行补全的位置使用类似”e.$$f”的格式进行标记。在后续工作中，成熟的推荐工具可以加入到JavaScript IDE中进行作用，实时对”.”后的内容进行推荐。
2. 工具输出：推荐的属性补全列表，按推荐度从高到低进行排序。
3. 总体方法：使用静态指针分析与动态插桩方法相结合的混合分析方法

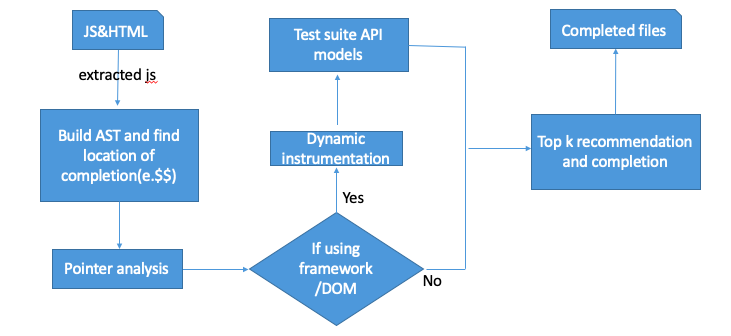
## 概要设计

主要针对第一部分提到的四大难点进行相应解决。

1. 对于C1，我们使用指针分析对JS代码进行分析，找到指向变量e的所有val及其类型，便于进行推荐。
2. 对于C2和C3，使用动态分析方法，记录框架与本地库中各变量及其属性以及关键函数及其调用等。值得注意的是，若是使用动态分析分析所有框架代码，代价十分巨大，因此考虑只动态运行其test suite。原则上，框架代码的test suite应该包含对该框架所有重要变量及函数的测试，因此通过运行test case得到这些变量和函数的属性，便于进行补全。
3. 对于C4 ，由于当前在开发过程中的需要补全的代码可能没有被其他程序调用过，无法使用指针分析，那么可以使用以下三点启发式的方法：记录当前程序中所有被读写的对象；记录局部变量和参数；默认认为作为一个对象属性的函数在完成该对象内属性补全的时候一定会被推荐。

## 详细设计

该工具流程图如下：



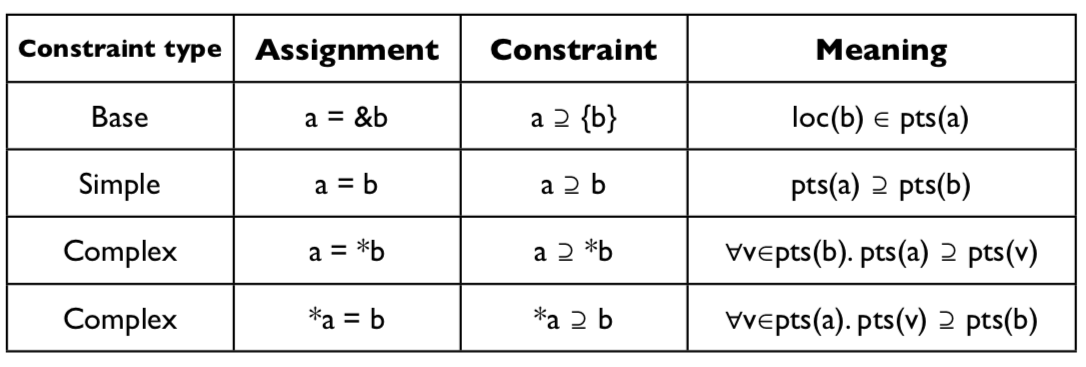
首先，从输入的js/html中提取js代码，使用esprima构建抽象语法树，找到具有形如”e.$$f”的表达式，这种表达式表示需要补全对象e的属性。接着，对代码进行指针分析，记录指向该对象e的所有val及其prototype。若是使用了框架代码或本地库，需要增加动态分析。使用js\_wala插桩框架test suite并动态运行，记录所有对象的属性及方法及函数调用等信息。最后，根据动静态分析相结合的结果进行代码推荐。

核心方法：

1. Pointer Analysis 指针分析

指针分析的目的是对于任何一个指针／引用，在编译阶段就知道它会指向内存那块位置(位置在这里并不是0xFFFF这样的具体位置，而是说指向哪个stack/heap上local/object)。这样我们把问题转化为: 对于任何一个任何指针，能否在编译阶段知道它所有可能(may)指向的位置。因为我们关心的是指针所指向的位置，指针是一个变量，能改变变量值的命令只有赋值语句。

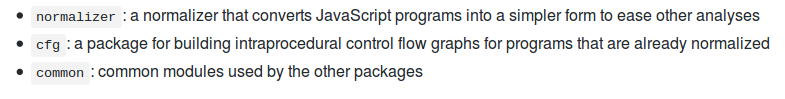
和C类似，我们定义有关指针的赋值语句见下表第一列:



指针p实际是一个集合(pts)，它的元素是所有它可能指向的变量。所以不难理解表中前两行的约束和意义了: 当把变量b所在的地址赋值给a时，a可能指向的变量自然就包含了b的地址啦; 当b是指针时，把b的值赋值给a相当于把b所有可能的元素添加到a中。对于’\*’我们要多分析一层，\*b表示的是b中的每个元素。举个例子,　假设b = {p, q}，那么\*b指的就是p和q。so对于第三行相当于把p的值赋值给a且把q的值赋值给a, 也就是把集合p和q的所有元素添加到a里。第四行则是把(假设a = {i, j}) b赋值给i且也赋值给j, 也就是i和j都讲拥有b所有的元素。同其它数据流分析一样，在分析时我们先把赋值语句转化成约束，在对约束求解不动点从而得出每个指针变量的值。[1]

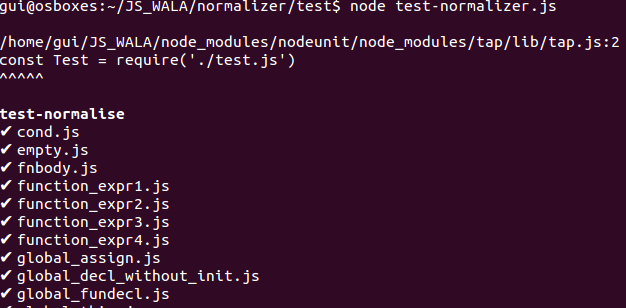
2.JS\_WALA[2]

是一个js写成的wala工具，可以对java及js代码进行动态插桩分析，共有以下三个部分：

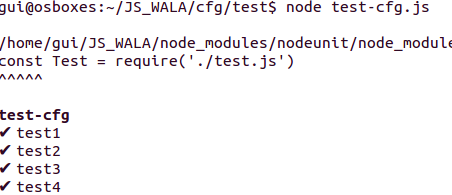


## 运行结果

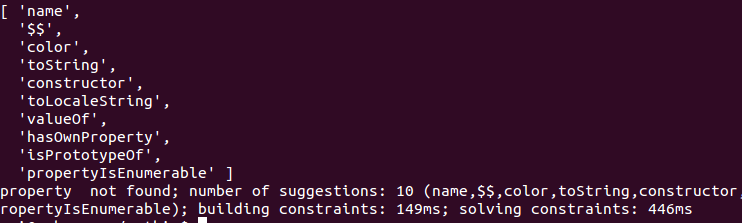
使用JS\_WALA对程序进行标准化：



生成控制流程图:



对测试js/html文件进行属性补全:



## 参考资料

[1] <https://blog.csdn.net/majestyhao/article/details/49961585>

[2] <https://github.com/wala/JS_WALA>