CS502: Compiler Design

Fall 2021 (Due Sep 4th, 2021)

Assignment A1: Helping Pooh

1 Assignment Objective

Use JavaCC and JTB to extend a programming language and traverse over parse trees.

2 Helping Pooh

Pooh bear (yes, the one from your favorite childhood cartoon "Winnie the Pooh") is so lazy that once he saw honey he left his grammar partial. So help Pooh finish his task by completing the grammar for the language "PoohJ". Also teach him how to count things by traversing parse trees.

3 Detailed Specification

You are provided with a grammar file `poohj.jj`, which models a Java-like object-oriented programming language, consisting of classes, objects, integer variables and arrays, while loops, etc. Your first task is to expand the jj file to include few more language features, as listed below.

- Types: A floating-point type `float`
- Algebraic operations: subtraction (`-`), multiplication (`*`) and division (`/`)
- Logical operations: or (`||`), comparison (`<=`)
- Object-oriented: inheritance (`extends`)
- Array-related: array-lookup (`Identifier[Identifier]`), array-length (`Identifier.length`)
- Loops: Standard for loop with single index-variable declaration, condition check, and increment by 1

Once you have completed the language `PoohJ`, you need to generate a parser, and write a visitor to print the following: For each class, for each method, print (in lexicographic order):

- Classname : ParentClassname
- Classname.methodname
- No. of local variables (includes parameters)
- No. of unreferenced variables
- No. of control-flow altering statements (excluding system calls and constructors)

3.1 Example Input

```
class Test {
    public static void main(String[] args) {
        System.out.println(new A().foo(10));
    }
}
class A {
    public int foo(int p) {
        int x;
        boolean y;
        boolean b;
        int z;
        int w;
        int q;
        x = 10;
        y = true;
        z = 0;
        while (y) {
            z = z + p;
            if (z != 10) {
                y = false;
            }
        }
        q = this.bar();
        return z;
    }
    public int bar() {
        int a;
        int b;
        a = 10;
        return a;
    }
}
class B extends A {}
```

3.2 Example Output

```
Class A :
Method A.bar 2 1 1
Method A.foo 7 2 4
Class B : A
Class Test :
Method Test.main 1 1 1
```

3.3 Evaluation

Your submission must be named as `rollnum-a1.zip`, where `rollnum` is your roll-number in small letters. Upon unzipping the submission, we should get a directory named `rollnum-a1`. The main class inside this directory should be named `Main.java`. Your program should read from the standard input and print to the standard output. You can leave all the visitors and syntax-tree nodes as it is, but remember to remove all the `.class` files.

We would run the following commands in the evaluation script:

- `javac Main.java`
- `java Main < test > out`

If the contents of `out` match with the expected output for the testcase, you would get marks for the corresponding testcase.

4 Plagiarism Warning

You are allowed to discuss publicly on the Slack channel, but are supposed to do the assignment completely individually. We would be using sophisticated plagiarism checkers, and if similarity is found, the penalty used in the course would be as follows:

- First instance: 0 marks in the assignment
- Second instance: F grade in the course

-*-*-*- Do the assignment honestly, enjoy learning the course. -*-*-*-