

Goal

1. Implement type checking for the Java programming constructs that were introduced in *j--* as part of Project 3 (Parsing).
2. Implement JVM code generation for those Java programming constructs.

Download the Project Tests

Download and unzip the tests [↗](#) for this project under `$j/j--`.

Run the following command inside the `$j/j--` directory to compile the *j--* compiler with your changes:

```
>_ ~/workspace/j--
$ ant
```

To compile a *j--* program `P.java`, run the following command:

```
>_ ~/workspace/j--
$ bash ./bin/j-- P.java
```

Run the following command to run the *j--* program `P.class`:

```
>_ ~/workspace/j--
$ java P
```

Problem 1. (*Long and Double Basic Types*) Add support for `long` and `double` basic types.

```
>_ ~/workspace/j--
$ bash ./bin/j-- project5/tests/BasicTypes.java
$ java BasicTypes 1.0 42
3.14159
145236
```

Problem 2. (*Operators*) Add support for the following operators.

```
~      !=      /      /=      -=
++      --      *=      %      %=
>>      >>=     >>>     >>>=    >=
<<      <<=     <      ^      ^=
|      |=      ||     &      &=
```

```
>_ ~/workspace/j--
$ bash ./bin/j-- project5/tests/Operators.java
$ java Operators
false
true
2
2
true
false
```

Problem 3. (*Conditional Expression*) Add support for conditional expression $(e_1 ? e_2 : e_3)$.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/ConditionalExpression.java  
$ java ConditionalExpression 31  
odd  
$ java ConditionalExpression 42  
even
```

Problem 4. (Switch Statement) Add support for a switch statement. Here's some code you may want to use to decide which instruction (TABLESWITCH or LOOKUPSWITCH) to emit:

```
long table_space_cost = 4 + ((long) hi - lo + 1);  
long table_time_cost = 3;  
long lookup_space_cost = 3 + 2 * (long) nlabels;  
long lookup_time_cost = nlabels;  
int opcode = (nlabels > 0 && table_space_cost + 3 * table_time_cost <= lookup_space_cost  
             + 3 * lookup_time_cost) ? TABLESWITCH : LOOKUPSWITCH;
```

Where hi is the highest case label value, lo is the lowest case label value, and $nlabels$ are the total real case labels in the switch statement. For example, in the following code, 1, 3, and 5 are the real labels, whereas 2 and 4 are the fake labels to be generated.

```
switch (a) {  
    case 1:  
    case 3:  
    case 5:  
}
```

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/SwitchStatement.java  
$ java SwitchStatement 4  
Spring  
$ java SwitchStatement 7  
Summer  
$ java SwitchStatement 10  
Fall
```

Problem 5. (Do-while Statement) Add support for a do-while statement.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/DoWhileStatement.java  
$ java DoWhileStatement  
55
```

Problem 6. (For Statement) Add support for a for statement.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/ForStatement.java  
$ java ForStatement  
55
```

Problem 7. (Break Statement) Add support for a break statement.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/BreakStatement.java  
$ java BreakStatement 5  
0  
1  
2  
3  
4
```

5

Problem 8. (*Continue Statement*) Add support for a continue statement.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/ContinueStatement.java  
$ java ContinueStatement 10  
1  
3  
5  
7  
9
```

Problem 9. (*Exception Handlers*) Add support for exception handling, which involves supporting the try, catch, finally, throw, and throws clauses.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/tests/ExceptionHandlers.java  
$ java ExceptionHandlers 42  
42: The answer to life, the universe and everything!  
Done!  
$ java ExceptionHandlers 43  
43  
Done!
```

Problem 10. (*Interface Type Declaration*) Implement support for interface declaration.

```
>_ ~/workspace/j--  
$ bash ./bin/j-- project5/testes/Interface.java  
$ java Interface 5  
25
```

Files to Submit

1. \$j/j--/project3/report.txt
2. \$j/j--/src/jminusminus/j--.jj
3. \$j/j--/src/jminusminus/JAssignment.java
4. \$j/j--/src/jminusminus/JBinaryExpression.java
5. \$j/j--/src/jminusminus/JBooleanBinaryExpression.java
6. \$j/j--/src/jminusminus/JComparison.java
7. \$j/j--/src/jminusminus/JBreakStatement.java
8. \$j/j--/src/jminusminus/JClassDeclaration.java
9. \$j/j--/src/jminusminus/JConditionalExpression.java
10. \$j/j--/src/jminusminus/JConstructorDeclaration.java
11. \$j/j--/src/jminusminus/JContinueStatement.java
12. \$j/j--/src/jminusminus/JDoWhileStatement.java
13. \$j/j--/src/jminusminus/JForStatement.java
14. \$j/j--/src/jminusminus/JInterfaceDeclaration.java
15. \$j/j--/src/jminusminus/JLiteralDouble.java
16. \$j/j--/src/jminusminus/JLiteralLong.java

```
17. $j/j--/src/jminusminus/JMethodDeclaration.java
18. $j/j--/src/jminusminus/JSwitchStatement.java
19. $j/j--/src/jminusminus/JThrowStatement.java
20. $j/j--/src/jminusminus/JTryCatchFinallyStatement.java
21. $j/j--/src/jminusminus/JUnaryExpression.java
22. $j/j--/src/jminusminus/Parser.java
23. $j/j--/src/jminusminus/Scanner.java
24. $j/j--/src/jminusminus/TokenInfo.java
25. $j/j--/src/jminusminus/Type.java
```



Before You Submit

- Make sure you name the classes and files you create exactly as suggested in this writeup. Remember, names are case-sensitive.
- Make sure your report uses the given template, isn't too verbose, doesn't contain lines that exceed 80 characters, and doesn't contain spelling mistakes.