Objectives.

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.

To compile the *j--* compiler, run the following command:

```
$ ant clean compile jar
```

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

```
$ $j/j--/bin/j-- P.java
```

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

```
$ java P
```

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

```
$ $j/j--/bin/j-- BasicTypes.java
$ java BasicTypes 1.0 42
3.14159
145236
```

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

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

```
$ $j/j--/bin/j-- Operators.java
$ java Operators
false
true
2
2
true
false
```

Problem 3. (*Conditional Expression*) Add support for conditional expression (`e1 ? e2 : e3`).

```
$ $j/j--/bin/j-- 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); // words
long table_time_cost = 3;                          // comparisons
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:
}
```

```
$ $j/j--/bin/j-- 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.

```
$ $j/j--/bin/j-- DoWhileStatement.java
$ java DoWhileStatement
55
```

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

```
$ $j/j--/bin/j-- ForStatement.java
$ java ForStatement
55
```

Problem 7. (*Exception Handlers*) Add support for exception handling, which involves supporting the `try`, `catch`, `finally`, `throw`, and `throws` clauses.

```
$ $j/j--/bin/j-- ExceptionHandlers.java
$ java ExceptionHandlers 42
42: The answer to life, the universe and everything!
Done!
$ java ExceptionHandlers 43
43
Done!
```

Problem 8. (*Interface Type Declaration*) Implement support for interface declaration.

```
$ $j/j--/bin/j-- Interface.java
$ java Interface 5
25
```

Files to Submit

1. `j--.tar.gz` (*j--* source tree as a single gzip file)

2. `report.txt` (project report)

Before you submit:

- Make sure you create the gzip file `j--.tar.gz` such that it only includes the source files and not the binaries, which can be done on the terminal as follows:

```
$ cd $j/j--
$ ant clean
$ cd ..
$ tar -czvf j--.tar.gz j--/*
```

- 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

---