Language Definition for MiniJava, version 2010.0

Minijava is a greatly simplified version of Java, based on the
following rules. You can modify the language to include extra
features, but please wait until the end of the semester.

--- A MiniJava program must be a legal Java program.

--- A MiniJava program consists of a single file containing a single
    class declaration. The class must have a name that matches
    filename. The main class must public and must have a main method
    that is public, static, and void, and that takes a String[] as a
    parameter. All methods in the main class must be static. Methods
    other than the main method may be marked public, but the privacy
    setting will have no effect.

--- All variables and methods in the main class must be static. There
    are no initializers on declarations.

--- There are no inner classes, generics, enumerated types, or interfaces.

--- Packages and imports are not used.

--- The only primitive types are void, int, boolean, and String. There are no
    doubles or chars.

--- Variables of int, boolean, and String types can be declared.
    Arrays can also be declared and created.

--- Each variable must be declared on a separate line. No
    initializers are permitted. Variables can be declared locally
    within any block.

--- Strings are created only from a string literal or from
    concatenation of two Strings or of a String and an int. They can
    be used only in a concatenation or in a print() statement.

--- The String[] parameter in the main method is always initialized to
    an array of length 0.

--- Printing can be done with System.out.print(arg). The argument
    must be a single int or string value.

--- No do-while loops or for loops are permitted. All ifs must have an else
    clause.

--- There are two kinds of empty statements: either a semicolon on its
    own, or a pair of braces with nothing between them. Either can be
    used, for example, as an empty else clause.

--- An expression followed by a semicolon is a statement. (Note: this
    actually extends the Java rules a bit, which require that the
    expression not be a constant.)

--- Return statements are permitted, and they are required in methods
    that are not declared to be void.

--- Primary (lowest-level) expressions are ids (for variables), method
    calls, parenthesized expressions, iconsts, sconsts, and the
    following keywords: this, null, true, and false.

--- The following operators are permitted:

```
        array creation:     new Whatever[size][][]
                    (where each extra [] implies an extra dimension).
        postfix ops:        [index], .length, .length()
        multiplicative ops: *, /, %
        additive ops:       +, -
        relational ops:     < > <= =>
        equality ops:       == !=
        logical and:        &&
```

```
logical or:         ||
assignment:         =
```

There is no unary minus. The operators on each line above have equal precedence, and the operators appearing higher on the list take precedence over those appearing lower on the list. All binary operators as left associative, except && and ||, which are right associative.