

Assignment 6: Object-oriented Interpreter

1 Implementation

You are to implement an interpreter for an extended version of the imperative language featuring objects. The object-oriented extension works exactly as in Java language.

You will implement a Java class `Interpreter` with a main method that reads a file in the given language from standard input (`System.in`) and executes the static `main` method (which takes no arguments) of the first, non-optional class defined in the input. The input file may contain other classes which maybe used in by the main class. Execute the statements, functions and methods invoked by the main method until the program returns or fails.

If the execution fails, the `Interpreter` should print the line and column number of the failing operator (`beginLine` and `beginColumn` fields of `NodeToken`) with a descriptive error message to standard error. The format of the message should be “LINE MESSAGE”.

If the file does not parse, the program should print “Parse error.” to standard error (`System.err`).

2 Submission

You must submit the implementations to your subversion repository to the directory `comp3351/f2007/$USER/P6/`. Include only the provided grammar, the `Interpreter` implementation and the provided build script. The files must be called

- `objects.jj`
- `Makefile`
- `src/edu/du/cs/comp3351/p6/Interpreter.java`

You must check that the submitted code compiles by invoking `make`. Verify that the output of your program matches the expected output using your own testcases.

You will not get any points if your submission does not compile without modifications, fails to run with the provided testing script or if your submission is in the wrong directory. Make sure to use “P6” for the directory name and **not** “p6”. Furthermore, make absolutely sure to use the correct package (“edu.du.cs.comp3351.p6”) for your **Interpreter** class. Also make sure that your **main** method is in **Interpreter** and not in **Parser**.