**Homework 4 - Generating Lower-Level Intermediate Code**

The next step of your project will translate the Piglet intermediate representation to an even lower-level IR, which we'll call Spiglet. Spiglet is a proper subset of Piglet. The [Spiglet grammar](http://cgi.di.uoa.gr/%7Ethp06/project_files/spiglet.html) ([JavaCC version](http://cgi.di.uoa.gr/%7Ethp06/project_files/spiglet.jj)) has two differences from that of Piglet:

* A StmtExp is not an Exp in Spiglet.
* In many places, a SimpleExp or Temp is used instead of an Exp.

Practically this means that many expressions need to be broken down into simpler ones. The usual example programs in Spiglet are [here](http://cgi.di.uoa.gr/%7Ethp06/project_files/spiglet-examples) (corresponding to the [MiniJava examples](http://cgi.di.uoa.gr/%7Ethp06/project_files/minijava-examples)). Since Spiglet is a subset of Piglet, you can use the same [formatter](http://cgi.di.uoa.gr/%7Ethp06/project_files/pretty-printer.jar) and [interpreter](http://cgi.di.uoa.gr/%7Ethp06/project_files/pgi.jar) as in the previous homework. If you want to ensure that a Spiglet program is legal Spiglet (and not just Piglet) you can use [this parser](http://cgi.di.uoa.gr/%7Ethp06/project_files/spp.jar) by typing "java -jar spp.jar < [Input-program]". If the program is syntactically legal Spiglet, you will receive the message "Program parsed successfully".

Your program should run as follows:  
  
java [MainClassName] [file1.pg] [file2.pg] ... [fileN.pg]  
  
That is, your program must compile to Spiglet all **.pg** files given as arguments. Moreover, the outputs must be stored in files named file1.spg, file2.spg, ... fileN.spg respectively.