

Parser

For this milestone you will extend your software from the previous milestone to parse MiniJava programs.

Your parser will take the name of a text file on disk as an argument. The program will parse the file according to the syntax given in the Term Project Description. Your program will output a parse-tree.

Additionally, augment your software so that it performs panic mode error recovery and some phrase-level error recovery.

DELIVERABLES

Sample Input

You must submit 4 sample input files, **by midnight at least three days before** the milestone deadline. *Three of the sample inputs should be **valid** MiniJava programs. The fourth should include at least one syntax error.* Please carefully consider the grammar in the Term Project Description when creating your samples.

Please note: The following properties of MiniJava are the most common sources of problems:

- Class declarations and field declaration in MiniJava must **not** have a privileged access modifier; they cannot be declared public, private, or protected.
- MiniJava's main class must have just a main method; it must not have any other methods. All methods in MiniJava must be declared public. The main method must not have a return statement; all other methods must end with a return statement.
- The set of valid statement forms in MiniJava is much smaller than Java's. Verify that each statement in your samples matches one of the six statement forms:
 1. variable declaration with initialization,
 2. block,
 3. if-then-else,
 4. while,
 5. print, or
 6. assignment.

Your sample input files should simply be placed in your Subversion repository. **Before the deadline, please email me with the location of your sample input files within your repository.** Please limit your sample inputs to no more than 250 lines and no more than 80 characters per line. You do not have to submit sample output at this time.

I will give a set of sample inputs to every team at least two days before the milestone deadline. I will give a full set of sample inputs to every team at class time on the milestone deadline.

Sample Output and Code

By midnight on the milestone deadline you must commit to your Subversion repository:

- the output of your program for each sample input (well labeled so I can match input to output)
- the source code for your program
- a plain text file named Parser.txt in the MilestonePointers subdirectory of your repository.
This file should contain a pointer to the source code and program output for your parser.

Grading

Your grade will be based on whether you submit the required sample inputs and the percentage of the sample input files that your program processes correctly.