Parser and Tree Builder

Total 100 points.

/accounts/classes/janikowc/submitProject/submit_cs4280_P2 *SubmitFileOrDirectory*

Invocation:
> frontEnd [*file*]
with *file* as before
Wrong invocations may not be graded.

Graded 90% execution 10% structure/standards.

Verify the project grammar is LL(1) or rewrite as needed in an equivalent form.

Use your scanner module and fix if needed. If you fix any errors that you lost points for on P1, ask to have some points returned after P2 works.

Implement the parser in a separate file (`parser.c` and `parser.h`) including the initial auxilary `parser()` function and all nonterminal functions. Call the parser function from main. The parser function generates error or returns the parse tree to main. In `testTree.c` (and `testTree.h`) implement a printing function using preorder traversal with indentations as before for testing purposes (2 spaces per level, print the node's label and any tokens from the node, then children left to right; one node per line). Call the printing function from main immediately after calling the parser and returning the tree. **The printing function call must be later removed for P3 unless debugging.**

The project P2 will be tested assuming white spaces separate all tokens.