To slay a dragon
The quest which seemed outlandish
Is now completed

Working:

1.0

- Line numbering
- Two Styles of Comments

2.0

- Unlimited nesting of subprograms
- Array access on both sides of assignment
- Allow **for** statements
- Another loop construct (while)

2.5

- Visual Print
- (Not sure what memory leak handler is, so unsure if I have it or not)

3.0

- All Items in Checklist
- Error Reporting

4.0

- Input/Output statements
- Simple expressions (arithmetic and relational): gencode
- Statements (assignment, conditional, loop)
- Nonlocal names: base frame pointer (static scope parent)
- Recursive routines (example: GCD program)
- Complex expressions (register spilling)
- (optional) Arrays (L-value, R-value, parameters, nonlocal)

The gencode issue with dangling else (where no code ran in the then statements with no else) has been fixed.

Left to do:

- Floating point
- SLASH, AND, OR operators

Free-Will:

- Array access (L-value, R-value, parameters, nonlocal)
- Optional inclusion of result keyword in function declarations
- While loops

My compiler is working on all test cases in the Semantic Folder, and all test cases in the CodeGeneration folder up to t12.

How to use:

- Make (default) compiles the compiler into an executable called mypc
- All input to mypc is done through input redirection
- The assembly output is placed in out.s
- Make assemble assembles out.s into an executable called COMPILED