Programming Project 9
Final Project Phase C
EE312 Summer 2014

Due August 15th, 2014 before 10:00AM CDT (classtime)
Projects received after 10:00AM on August 15th will receive 0.

FIFTEEN POINTS

General: For our final project we will write our own little toy programming language. The language will have functions, loops, conditional statements, and arithmetic. For Phase C, we're asking you to polish up your projects, complete any missing functionality that you think you can still knock out, and pretty-up the code. There is no new functionality required for Phase C beyond what was already specified for Phase B. However, the design and coding quality of your implementation will be a major part of your Phase C grade (Probably about half).

All the code you produce in this project must be original. You may not seek outside help (other than the assistance of the instructor, TA and authorized tutors). Please keep in mind that we will be using a plagiarism checker with this project.

Using BSTMap
I've coded up a binary search tree implementation and organized it into a map. You are not obligated to use this data structure in your solution to the project. It is provided for your reference and/or convenience. The advantage of the BSTMap is that it is a full-fledged map, with insert/remove/find and scan functionality, as well as copy constructors, destructors and assignment operators. I used the BSTMap as the basis for my symbol table. Actually, I used multiple instances of the BSTMap (one for each stack frame). Anyway, if you're using an array and want to upgrade to a tree for your symbol table, you're welcome to use the BSTMap. Please only use the BSTMap if it makes your life easier. You won't get bonus points for using my code!

Memory Leaks
An interpreter that does not leak memory is a better solution than an interpreter that leaks memory. As such we will begin testing for memory leaks. The penalty for having memory leaks will be small. In fact, the penalty might be zero, depending on how the project is going. However, we reserve the right to give a higher score to a project without memory leaks than to an equivalent project with leaks. The solution does not appear to have any memory leaks at present, so I'm pretty sure it can be done.

That's it!
That's all there is. Please refer to the Phase A and Phase B instructions for specifications regarding what a Blip program is supposed to do when it is run.

Good luck and have fun!