**CSE 504: Assignment 2**

**Loop Identification**

**Due date:  Tuesday, November 6**

In this assignment, you will implement an LLVM pass to identify loops and do a simple loop nesting analysis. This assignment is intended to give you more insight into the LLVM compiler analysis and help you better understand the LLVM IR.

To complete this assignment, you will perform the work described below. The report and the files are to be made available to the TA for evaluation by midnight on the due date. Do **not** delete interim versions of your files or anything else that documents how you have performed your assignment tasks.

**Here is what you should do:**

For this work you will use your account on Seawulf. You should already be familiar with the basics of LLVM/Clang. You will need the LLVM IR and basic pass knowledge gained in the first assignment. Assume no optimization is enabled for this assignment.

The things you need to accomplish are listed as follows.

1) Write a simple program that contains several loops (including nested loops), and observe the LLVM IR code generated for loops. Document your observations about the IR code for loops, including 1) how the loop functionality is implemented, and 2) the control flow pattern at the level of basic blocks.

2) Write a pass to identify and number all loops in a function. The loop identification method should be derived from your observation above. The desired pass should print out how many basic blocks and instructions are contained in each loop discovered. For example:  
Loop 0: 3 basic blocks; 11 instructions  
Loop 1: 5 basic blocks; 30 instructions

3) Analyze the nesting relationship of loops in your pass, i.e., determines whether a loop is contained by another one. As a result, the pass should also print out which loops are nested within which other loops. For example:  
Loop 1 is nested within loop 2  
Loop 4 is nested within loop 2

4) Write a test program, which includes various non-trivial loops (multiple basic blocks!) and nested loops. The remainder of the test program should be relatively simple. Test the implemented pass with this program.

You will only need to compile your test program to the IR (i.e., using -c -S -emit-llvm), and thus a complete program is not required (i.e., the main function is not necessary).

As always, do **not wait** until the due date to do your work: otherwise the machine may become very full around the due date.

You will need to submit your pass code as a zip file, as well as an assignment report. The zip should include all the code changes that you have done along with all intermediate .ll files and output files generated. Please name the zip file according to your SBU\_ID. Please also leave appropriately named files in an appropriately named directory in your course account in case we need to inspect them. Your documentation should provide any explanations needed.

Your report should describe your observations on the loop IR code, the design and implementation of your pass, and the test code and results. It should be no more than 4 letter-size pages in length.

**Grading:**

**Grades from A through F** will be assigned. Grades will be based upon the correctness of your work, completeness of your responses, and the overall quality of your work. Please do your own work: do NOT copy code or text from other class participants or from any other source.