

# CS3423: Mini-Assignment #1

## Due Sunday September 13<sup>th</sup>, 2020 at 11:59 PM

1. List out the differences between the working of compilers and interpreters. Analyse the differences by taking into consideration some simple programming examples and running them using an interpreter and compiler (ex: Python and C/C++ programs). Also, briefly compare them on Error and Exception handling, Memory management, Intermediate representation, and how they handle type information.

**Note:** Take this exercise as a hands-on. Meaning, do not simply list out the differences found on the web. Back it up with experimental evidence

2. Investigate what lexical analysers and parsers are used in GCC and Clang/LLVM infrastructures? Give a brief description about each of them.

3. Write a note on the various standard flags used in compilers i.e, -S -E -g -c, etc. Also, write a note on the various optimization passes of compilers: O0, O1, O2, O3, -Os -Oz, levels of Compilers.

4. **[Bonus]** Analyse the time taken by each module: Lexical analysis and parsing, Code Optimization and Code generation in GCC and LLVM. List out your findings for some examples. [For this it is recommended that you run a sufficiently large program and do precise measurements.]

**Submission** Your submission should be a gzip archive with name Mini-Asgn-1\_ROLLNO.tar.gz containing all code samples used and a PDF, preferably generated from a lyx/latex file, containing the answers to the above questions.

**Evaluation** All the questions (1-3) in this assignment carry an equal weightage and you will be evaluated on the basis of reportings in the hands-on exercise and answers of remaining questions. Bonus question will be considered for bonus points.