

COMP 520 Compiler Design Group Milestone #4

Code Generation for GoLite

Due: Wednesday, April 10 11:59 PM

Question 1: *Benchmarks* (5 points)

For each team member, provide 1 benchmark that executes for around 5s (in the Go playground) and computes the solution to a real world problem. It should be valid according to the reference compiler. Use a variety of language features to test performance throughout the generated code.

Question 2: *Final Code Generator* (45 points)

In this last milestone you will complete your code generator. You should now support all constructs in GoLite, and have a fully functional compiler. We do not expect a perfect code generator handling all edge cases, but you should strive to be as complete as possible. To this end, you should be testing on a variety of programs for each construct. The more complete your testing, the more complete your compiler! Please provide a working `execute.sh` script to interact with the provided `verify.sh` script. If you have any questions about this requirement, please come see us!

What to hand in

Create a tag in your Github repository named `milestone4` (lowercase, no extra characters). Information about creating git tags can be found at: <http://git-scm.com/book/en/v2/Git-Basics-Tagging>. Your project should be kept in the following format

```
/
README    (Names, student IDs, any special directions for the TAs)
programs/
  1-scan+parse/
    valid/
    invalid/
  2-typecheck
    invalid/
  3-semantics+codegen/
    valid/
  3-benchmarks+codegen/
    valid/
doc/      (Design documents)
src/      (Source code and build files)
execute.sh (Updated execute script)
build.sh  (Updated build script)
run.sh    (Updated run script)
```