

COMP 520 Compiler Design

Group Milestone #4

Code Generation for GoLite

Due: Tuesday, April 4

Final Code Generator (50 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.

Your compiler should support a set of flags, including the normal `-h` and `-v` flags to give the help and version information.

What to hand in

For this last milestone you should create a tag called *milestone4*. 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          Your group names, student IDs, relevant info and
                   instructions for each milestone (just add information
                   as you finish each milestone. Make it easy for the TAs to
                   grade your milestone!. Your README file should also
                   include references to any code that you have read to get
                   ideas, or code that you have used in your compiler.
                   For code that you have used, you must ensure that:
                       (1) you have permission to use it, and
                       (2) you have clearly indicated in your compiler
                           the source of the code.

  programs/
    valid/         (your valid programs from M1)
    invalid/       (your invalid programs from M1)
      types/       (your invalid types programs corresponding to M2)
    benchmark/     (your two benchmarks programs from M3)
    code/          (your valid programs corresponding to M3)
  src/            (the source code and build files. You must use some sort of
```

	automatic build system like Makefile or ant)
doc/	(design documents)
build.sh	(a build script to compile your compiler)
run.sh	(a run script that when invoked as "run derp.go" will run your compiler)