COMP 520 Compiler Design
Group Milestones #3 and #4

Code Generation for GoLite or OncoTime ....

Due: Milestone #3: Monday, April 4
Due: Milestone #4 and Final Report: Friday, April 15

Overview:

The purpose of this milestone is to develop the code generator for your compiler. You will do this in two phases as described below in Milestone #3 and Milestone #4.

Milestone #3

Question 1: *Computation Intensive Programs* (5 points)

Develop two example programs which perform some reasonably intense computation, something that you think will run for between 10 and 60 seconds when compiled (GoLite), or which does some non-trival database query (OncoTime).

We will use these benchmarks to evaluate the efficiency of the code produced by the code generators.

Question 2: *Code Generation Status Report* (10 points)

For Milestone #3 you should:

- Decide upon what sort of code you will generate. You can choose between low-level code or high-level code such as C, JavaScript, Python or Java.
- Design the code generation patterns for the key constructs of your language. Ensure that you are not choosing a strategy that will generate excessively slow code. For example, if you are generating C code, then attempt to generate code that the C compiler will be able to effectively optimize.
- Implement the code generation for a subset of your language. Choose some key language features. For OncoTime a could goal would be able to do the filtering and generate code for the `foreach patient` construct. For GoLite, you should choose some simple expressions, statements and control constructs to start with.

What to hand in

You will be developing your project in your team's github repository. At each milestone you will create a tag before the due date, and the TAs will review the code associated with that tag. For this milestone you should create a tag called *milestone3*. Information about creating git tags can be found at: `http://git-scm.com/book/en/v2/Git-Basics-Tagging`.

Please ensure the following documents are in your git project.

```
programs/
  benchmark/  (your two benchmarks programs)

doc/
  milestone3.pdf  (a summary (less than one page) of what you have
                   implemented)
```

Milestone #4: ***Final Code Generator*** (50 points)

You should complete your compiler, generating correct code. I expect that the GoLite teams should be able to generate code for all the constructs in GoLite. For the OncoTime teams, you should try to complete as many constructs as is reasonably possible. For events, you only have to handle five events, although supporting more events would be great.

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,  should have at least two for the
             first milestone)
   invalid/ (your invalid programs for testing)
      types/ (your invalid types programs corresponding to Q1)
   benchmark/  (your two benchmarks programs)
src/ (the source code and build files.  You must use some sort of
        automatic build system like Makefile or ant)
doc/ (design documents)
```

Final Report

The structure for the final report will be announced soon. In the interim, you should continue to keep your notes on design decisions.