CMPUT 275 Final Project Proposal (Sample)

Project Tile: Simple Arduino Calculator
Group Members: *names omitted*

Description:

In this project, we are planning to implement a simple calculator on Arduino, using the ILI9341 touchscreen as the I/O device. A GUI will be implemented on the touchscreen and the users can type expressions by touching the screen.

This calculator will support evaluating simple expressions like this:

$$1.1 + 2 \cdot 3 + (4 + 5/6 \cdot (7 - 8 \cdot 9))$$

We also hope to add support for some unary and trigonometric operators, such as sin(x), cos(x) and so forth. For simplicity, all digits are floating point numbers (but this is not yet settled for certain). We might also support integers if we have enough time.

Although this would make the parser much more complicated, we would like to add support for functions and variables as well.

Milestones:

1. **By March 15:** Build a simple GUI – a keyboard, where the upper space is reserved for displaying expressions and results.
2. **By March 22:** Build the lexer, which evaluates the whole expression the user typed and splits it into tokens. Tokens are obtained by calling a function called getNextToken(), which will evaluate and split the next token.
3. **By April 4:** Build the parser, this is a non-trivial part. We will evaluate a variety of algorithms and choose the best one. We are planning to look into the following algorithms: Recursive Descent (LL(1)), Shunting Yard Algorithm (converting the expressing to RPN) or other LR algorithms. Only one of the algorithms will be used.

Bonus Milestones (Time Permitting):

- Add support for variables and unary operators. Unary operators will include some common functions such as sin(x), cos(x), tan(x) and log10(x). In this part, we will implement a symbol table.
- Add support for user-defined functions with multiple input parameters. This is another non-trivial part. Additional modules might be added to support this feature.
- Support integers, rather than only floating-point numbers.

A note on this proposal: *The use of the touchscreen is rather minimal compared to what was done in the major assignments. However, the remaining elements of the project are non-trivial, in particular the algorithms used for parsing expressions. A project like this could be easily adapted to a desktop C++ program with a terminal or file-based user interface. It could alternately interface the Arduino display with a C++ server that does the more difficult computations.*