Lab 4
Expressions and design
22 September 2015

The drill for this lab is given below. Come to lab on Tuesday either with it completed or with a specific written question in your notebook identifying which drill step you got to and what about it you're stuck on.

Tinkerblock drill

This part of the lab is, again, philosophically similar to an end-of-chapter drill: it's somewhat contrived but lets you write a short program to practice the basics.

In your directory for this lab, you'll encode the work we already did in class, and then build a working program.

1. Add the description of the tinkerblock problem to the readme as a description of the program.

2. Don't worry about getting any drawn diagrams in there, but do encode the worked-out examples as test cases, with `.in` and `.expect` files.

3. Write a `.cpp` file with the general program stuff (`#include`, `main`, etc) and check that it compiles before you start adding more.

4. Steps 3 and 4 of the design process involve writing pseudocode and identifying nameable values (variables, constants) in the process. Do so.

5. Add them to your `.cpp` file piece by piece, writing code to read in data, compute the required values according to your algorithm, and print a result. Try to compile and test your code after every meaningful chunk that you add.