Discovered a method of finding truth values in bulk that could potentially speed the process up.

I was wrong. It doesn’t speed up anything.

As mentioned in yesterday’s notes, the majority of the time is being spent getting the cost and finding the truth value. Using a class would reduce this time but require more space, at least for finding the truth value. If I can figure out a way to get the cost of a circuit from its two components, instead of having to go through the process of parentheses matching each time, classes would also speed up this portion. A Hash could potentially work.

I will probably do it with Java because the ability of Java to create multiple constructors will be useful.

I didn’t do extensive testing of each function, but I made the foundation with Java. I made a circuit class that can calculate its cost and truthValue based on the two circuits used to make it. I did a few cursory tests and it seems like it can make about 10^7 circuits in about 12 seconds. This includes the calculation of truth values and cost. The next step is to make the circuit builder method and hope it is an improvement from the python one.

I finally completed the java version with classes and it runs significantly faster than the python version. As a test I did one operator “.” With cost limit 5 and the cost of the operator was 1 with 3 inputs. While the python program took 26 seconds, the Java program took 0.633 seconds. Using NOT, AND, and OR, with the rest of the settings the same, the python program took 165 seconds while the java program took 2.459 seconds. I checked the number of circuits found at each cost and the number of truth values found for each cost to make sure that it was finding all the same circuits and the numbers matched up.

This is a significant improvement to the Python code although it is not as efficiently commented and is very difficult to follow. I think I will call it a day now.

I am running into trouble because java does not let you mutate something as you iterate through it. Which slowed me down a lot.