Thevenin Resistance and Open Circuit Voltage
Braden Huffman and Jacob Feltman

Description:
It is often useful to reduce complex circuitry when modeling electronics. Direct current resistor circuits are frequently reduced down to a thevenin equivalent circuit in modeling. These circuits consist of a single voltage source and a resistor.

For this project, a framework will be built, allowing resistors, voltage sources and current sources to be put in both series and parallel. Mathematica has built in data structures such as linked lists that can be used to do this.

Once we are able to create circuits, through a series of Thevenin/Norton transitions along with resistor combinations, we will simplify more complex circuits to a point where an open circuit voltage and thevenin resistance can be obtained.

Tasks for Braden:
- Design the data structures for nodes in parallel and for nodes in series.
- Design an algorithm for switching between Thevenin and Norton equivalents given resistors, voltage sources and current sources in their decided data structures that calls a resistor simplification algorithm.
- Use the node voltage method to combine Thevenin circuit equivalents that are in parallel.

Tasks for Jacob:
- Design the nodes for resistors, voltage sources and current sources.
- Design an algorithm for resistor simplification given resistors in parallel and/or series in their decided data structures.
- Create normal and edge case circuits to test.

Analysis Requirements and Assumptions:
First, we will assume that all inputted circuits will be DC and can be simplified through resistor simplification along with switching between Thevenin and Norton equivalents. This analysis is primarily reductive in nature. It will be able to take in an arbitrary number of structured parameters and represent the whole using only two.

Possible Pitfalls:
While the process isn't mathematically complex, as all it will require is thoughtful addition, multiplication and division, it is algorithmically complex. Mathematica isn't traditionally an object oriented language, so representing different circuit components may not be trivial like it would in other languages. While it will be possible, it will just be harder to debug.