Adding an element

Adding an element post-constructor is easier than writing a new netlist file.

1. Adding to existing nodes

Say, we want to add an additional resistor to a simple voltage divider.

```
circuit = Circuit('circuits/voltage_divider.txt');
circuit.list
```

```
ans =
    'Vin 1 0 DC 5
     R1 1 2 1000
     R2 2 0 3000
     '
```

If we want to add an element between any existing nodes, the process is very simple. We create the element and add it.

```
R = Resistor('Rx',2,0,2000);
```

Be careful not to run the `add` function more than once, if this is not intended.

```
circuit.add(R)
circuit.list
```

```
ans =
    'Vin 1 0 DC 5
     R1 1 2 1000
     R2 2 0 3000
     Rx 2 0 2000
     '
```

2. Adding onto a wire

```
circuit = Circuit('circuits/voltage_divider.txt');
circuit.list
```

```
ans =
    'Vin 1 0 DC 5
     R1 1 2 1000
     R2 2 0 3000
     '
```

If instead we want to add an element onto a single node, the process is slightly more complicated. When -1 is used as a nodal connection, it means that the node has yet to be created, so in this case, the resistor R_x will go onto the wire between R_1 and R_2.

```
R = Resistor('Rx',2,-1,2000);
```

We can then add it to the circuit object using the `add` function. This will prompt the user to resolve any connection conflicts by choosing where to reconnect surrounding element.

```
circuit.add(R)
```

```
circuit.list
```

```
ans =
    'Vin 1 0 DC 5
     R1 1 2 1000
     R2 3 0 3000
     Rx 2 3 2000
     '
```

The circuit can now be manipulated as if it was created with a netlist.

```
ELAB.simplify(circuit);
circuit.list
```

```
ans =
    'Vin -1 0 DC 5
     Req1 -1 0 6000
     '
```