Opening Discussion

- Uses of recursion:
  - Graph traversals.
  - Searching solution spaces.
  - Divide and conquer.
  - So much more.
- IcP solutions.
Another one of my favorite recursive algorithms is formula parsing. This allows us to have the user type in a function and our code can evaluate it.

We do this through “divide and conquer”. We split the formula in two across the lowest precedence operator then recursively evaluate the two halves.

We can use this to put function plotting into our program if we give it the ability to handle a variable.
Some of the more efficient sorting algorithms are divide and conquer algorithms that are implemented with recursion.

- They divide on the way down, then may or may not do work on the way back up.
- Have $O(n \log n)$ average performance.
Merge Sort

- Break in half repeatedly on the way down. Recursively sort on each half.
- Merge sorted parts on the way back up.
- Can't happen in place because merge operation can't be done in one array.
QuickSort

- Can be done in place.
  - Pick a pivot.
  - Move all other elements either before or after the pivot as needed.
  - Recurse on the stuff before and after the pivot.
- Does all work on the way down, nothing on the way up.
- Inefficient List/Vector version is really short.
Minute Essay

- How are your projects coming?