CPSC326 | Homework 3: ASTParser + Pretty Print | 2/25/2025 | Arjuna Herbst

I created these tests to put stress on certain areas of the parser that I struggled with. Test 1 involves a while loop containing an if else block. Test 2 used mixed arithmetic operators nested in parentheses to ensure my parser properly prioritized the operators. The other tests involve nested function calls and recursive field accesses, since I had trouble telling the difference between function calls and variables references. I also struggled with keeping the parser in sync when encountering multiple elements that started with an ID, which led to errors like expecting a closing paren.

Test Case Inputs:

1.
```
void main() {
  while true {
    if false { x = 1 }
    else if true { x = 2 }
    else { x = 3 }
  }
}
```
2.
```
void main() {
  x = ((1 + 2) * (3 - 4)) / 5
}
```
3.
```
void main() {
  f(g(1, 2), h())
}
```
4.
```
void main() {
    ptr1.next[f(x) + g(y)].data[(a and b) or not (c)].value = new
     node(
         x.left[i + j].data,
         ptr2.values[k * 2].next[(x >= y) and (z <= w)],
         f(g(1, h(2, k())), m(n[0].x))
    )
    while ((x or y) and z) or (not (v) and not (((v == 0)
    or not (x)))) {
         arr[i + j].field.data[k - 1].value = (40 / (10 / 2))
    }
}
```

5.

```
        void main(){
            tree.left.right.data = new Node(
                left.value + right.value,
                null
            )
            if tree.height > maxHeight {
                tree.parent.balance = (left.depth - right.depth)
             }
        }
```