

#15 Days of Coding Challenge

#Day 8

Basic calculator ii

#Leetcode 227

Solution:

```
class Solution {
public:
    int calculate(string s) {
        stack<string> st;
        vector<string> vec;

        // 1. Tokenize input into numbers and operators
        for (int i = 0; i < s.length(); i++) {
            if (s[i] == ' ') {
                continue;
            }

            string t = "";
            while (i < s.length() && isdigit(s[i])) {
                t += s[i];
                i++;
            }

            if (!t.empty()) {
                st.push(t);
            }

            if (i < s.length() && (s[i] == '+' || s[i] == '-' || s[i] == '*' || s[i] == '/')) {
                st.push(string(1, s[i]));
                i++;
            }
        }

        // 2. Reverse into vector for easy access
        while (!st.empty()) {
            vec.push_back(st.top());
            st.pop();
        }
        reverse(vec.begin(), vec.end());
    }
};
```

```

// 3. First pass: handle * and /
stack<string> temp;
for (int i = 0; i < vec.size(); i++) {
    if (vec[i] == "*" || vec[i] == "/") {
        int num1 = stoi(temp.top());
        temp.pop();
        int num2 = stoi(vec[i + 1]);
        int res = (vec[i] == "*") ? (num1 * num2) : (num1 / num2);
        temp.push(to_string(res));
        i += 2;
    } else {
        temp.push(vec[i]);
        i++;
    }
}

// 4. Transfer back to vec
vector<string> newVec;
while (!temp.empty()) {
    newVec.push_back(temp.top());
    temp.pop();
}
reverse(newVec.begin(), newVec.end());

// 5. Second pass: handle + and -
int result = stoi(newVec[0]);
for (int i = 1; i < newVec.size(); i += 2) {
    string op = newVec[i];
    int num = stoi(newVec[i + 1]);

    if (op == "+") result += num;
    else if (op == "-") result -= num;
}

return result;
};

```

Time Complexity: $O(n)$

Space Complexity: $O(n)$