

z5242692

Chenqu Zhao

COMP9101 (T2-2020)

Homework 4 - Q1

Let there be n symbols and $n - 1$ operators between them. Now solve the following two subproblems: compute the number of ways to parenthesize the expression between the a th and the b th symbol to evaluate its value to true(T), compute the number of ways to the same but evaluate its value to False(F).

The base cases are that $T(i, i) = 1$ if symbol i is true, and $T(i, i) = 0$ if symbol i is false. In contrast, $F(i, i) = 1$ if symbol i is false, and $F(i, i) = 0$ if symbol i is true.

For each subproblem, we split the expression around an operator m so that everything to the left of m is in a bracket, and everything to the right of m is in another bracket. This way, we form two smaller expressions. Then, we evaluate the subproblems on each of the two brackets and combine the results together according to the type of operator we are splitting by, and whether we want the result to be true or false. Solve both subproblems in parallel:

$$\begin{aligned} T(a, b) &= \sum_{m=a}^{b-1} TSplit(a, m, b) \\ F(a, b) &= \sum_{m=a}^{b-1} FSplit(a, m, b) \\ TSplit(a, m, b) &= \begin{cases} T(a, m) * T(m + 1, b) & \text{if } m \text{ is AND} \\ T(a, m) * T(m + 1, b) + T(a, m) * F(m + 1, b) \\ \quad + F(a, m) * T(m + 1, b) & \text{if } m \text{ is OR} \\ T(a, m) * F(m + 1, b) + F(a, m) * T(m + 1, b) \\ \quad + F(a, m) * F(m + 1, b) & \text{if } m \text{ is NAND} \\ F(a, m) * F(m + 1, b) & \text{if } m \text{ is NOR} \end{cases} \end{aligned}$$

$$FSplit(a, m, b) = \begin{cases} T(a, m) * F(m + 1, b) + F(a, m) * T(m + 1, b) \\ \quad + F(a, m) * F(m + 1, b) & \text{if } m \text{ is AND} \\ F(a, m) * F(m + 1, b) & \text{if } m \text{ is OR} \\ T(a, m) * T(m + 1, b) & \text{if } m \text{ is NAND} \\ T(a, m) * T(m + 1, b) + T(a, m) * F(m + 1, b) \\ \quad + F(a, m) * T(m + 1, b) & \text{if } m \text{ is NOR} \end{cases}$$

The result $T(1, n)$ is the number of ways we can put parentheses in the expression such that it will evaluate to true.