Let there be n symbols, and $n-1$ operations between them. We solve the following two subproblems: How many ways are there to place parentheses to make the expression starting from at the l^{th} symbol and ending at r^{th} symbol evaluate to true (T), and many ways are there to place parentheses to make the expression starting from at the l^{th} symbol and ending at r^{th} symbol evaluate to false (F)? For example in "$true$ AND $false$ NOR $true$", $T(1,2)$ would be the number of ways of making "$true$ AND $false$" evaluate to true with correct bracketing (in this case, $T(1,2)=1$). The base case is that $T(i,i)$ is 1 if symbol i is true, and 0 if symbol i is false. The reverse applies to $F(i,i)$. Otherwise, for each subproblem, we 'split' the expression around an operator m so that everything to the left of the operator is in its own parentheses, and everything to the right of the operator is in its own parentheses to form two smaller expressions. For example, splitting the sample expression around "NOR" would give "$(true \text{ AND } false) \text{ NOR } (true)$". We then evaluate the subproblems on each of the two sides, and combine the results together depending on the type of operator we are splitting by, and whether we want the result to evaluate to true or false. We solve both subproblems in parallel:

$$T(l,r) = \sum_{m=l}^{r-1} \text{TSplit}(l,m,r)$$

$$F(l,r) = \sum_{m=l}^{r-1} \text{FSplit}(l,m,r)$$

$$TSplit(l,m,r) = \begin{cases} T(l,m) \times T(m+1,r), \\ \text{if operator } m \text{ is "AND"} \\ T(l,m) \times F(m+1,r) + T(l,m) \times T(m+1,r) + F(l,m) \times T(m+1,r), \\ \text{if operator } m \text{ is "OR"} \\ T(l,m) \times F(m+1,r) + F(l,m) \times F(m+1,r) + F(l,m) \times T(m+1,r), \\ \text{if operator } m \text{ is "NAND"} \\ F(l,m) \times F(m+1,r), \\ \text{if operator } m \text{ is "NOR"} \end{cases}$$

$$FSplit(l,m,r) = \begin{cases} T(l,m) \times F(m+1,r) + F(l,m) \times F(m+1,r) + F(l,m) \times T(m+1,r), \\ \text{if operator } m \text{ is "AND"} \\ F(l,m) \times F(m+1,r), \\ \text{if operator } m \text{ is "OR"} \\ T(l,m) \times T(m+1,r), \\ \text{if operator } m \text{ is "NAND"} \\ T(l,m) \times F(m+1,r) + T(l,m) \times T(m+1,r) + F(l,m) \times T(m+1,r), \\ \text{if operator } m \text{ is "NOR"} \end{cases}$$

Assume the expression has x symbols. The final solution is $T(1,x)$.