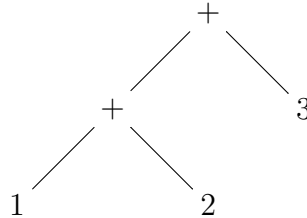# Determining the Order of Precedence

1. Addition and addition, e.g `1 + 2 + 3`

   When the same (binary) operator is used it works from left to right. So this can be interpreted as `(1 + 2) + 3`. The evaluation tree for this expression is

   It is worth noting that addition and subtraction have the same order of precedence and are evaluated from left to right. You cannot develop simple experiments to determine the true order of evaluation.

2. Determine the order of precedence for `+` and `*`.

   - Consider the expression `3 + 4 * 5`. There are two ways to add parentheses: `(3 + 4) * 5` and `3 + (4 * 5)`. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

   - Consider the expression `3 * 4 + 5`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

   - Which gets calculated first addition or multiplication?

3. Determine the order of precedence for `*` and `//`

   - Consider the expression `5 // 4 * 3`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

   - Consider the expression `5 * 4 // 3`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

   - Which gets calculated first multiplication or exponentiation?

4. Determine the order of precedence for `*` and `**`

   - Consider the expression `2 * 3 ** 4`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

- Consider the expression `2 ** 3 * 4`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

- Which gets calculated first multiplication or exponentiation?

5. Determine the order of precedence for `-` (negation) and `**`

- Consider the expression `- 2 ** 3`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

- Consider the expression `2 ** - 3`. There are two ways to add parentheses. Use Python to evaluate all three expressions to determine which way Python calculates the original expression. Draw the evaluation tree for the expression.

- Which gets calculated first negation or exponentiation?

6. (Challenge problem) Develop an experiment to determine the order of precedence of `and` and `or`.