

COMP 410: Abstract Syntax Trees and Evaluation

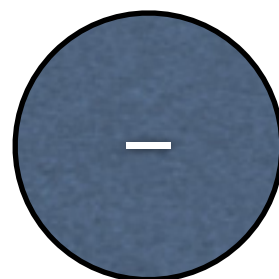
Kyle Dewey

Abstract Syntax Tree

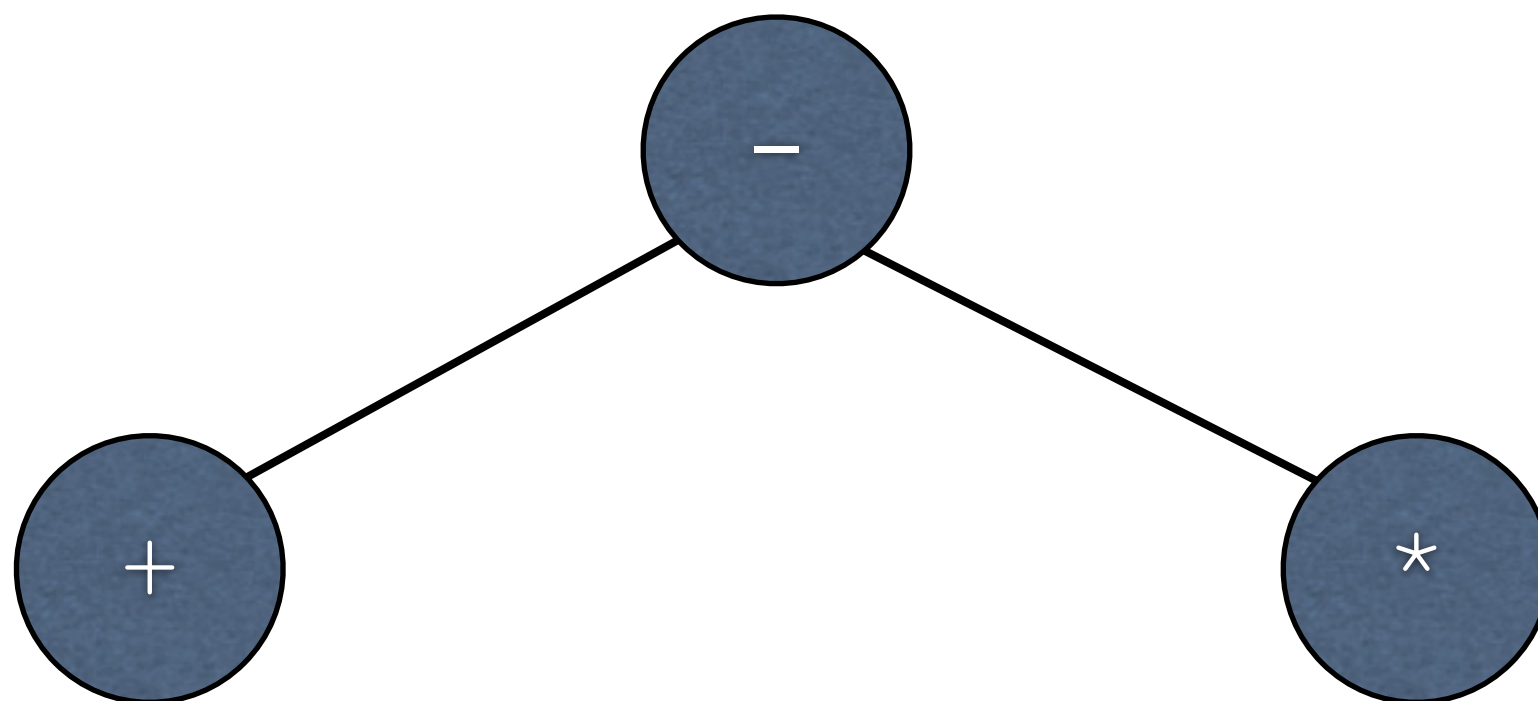
- Abbreviation:AST
- Unambiguous tree-based representation of a sentence in a language
- Very commonly used in compilers, interpreters, and related software

$$(1 + 2) - 3 * 4$$

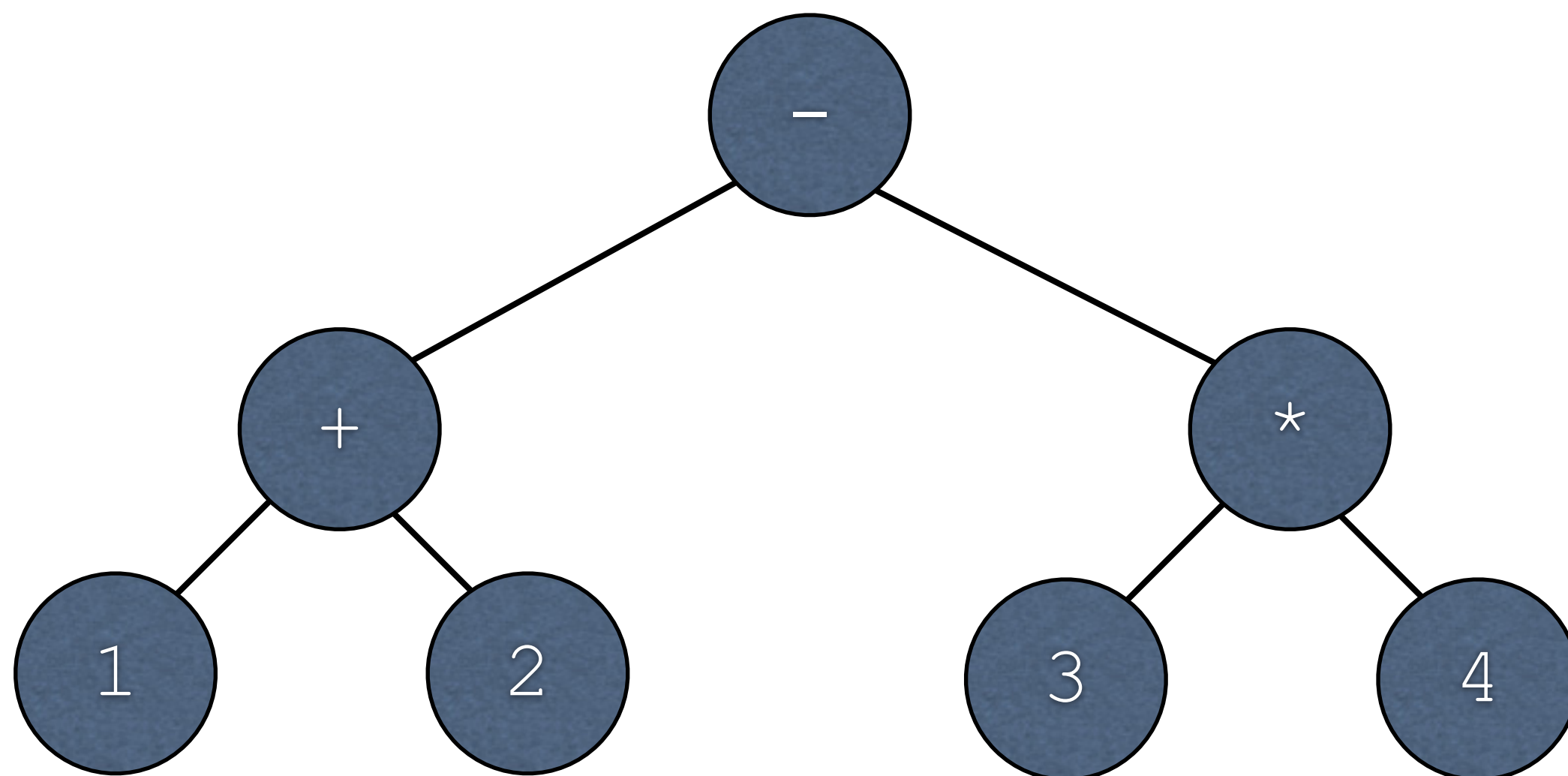
$$(1 + 2) - 3 * 4$$



$(1 + 2) - 3 * 4$

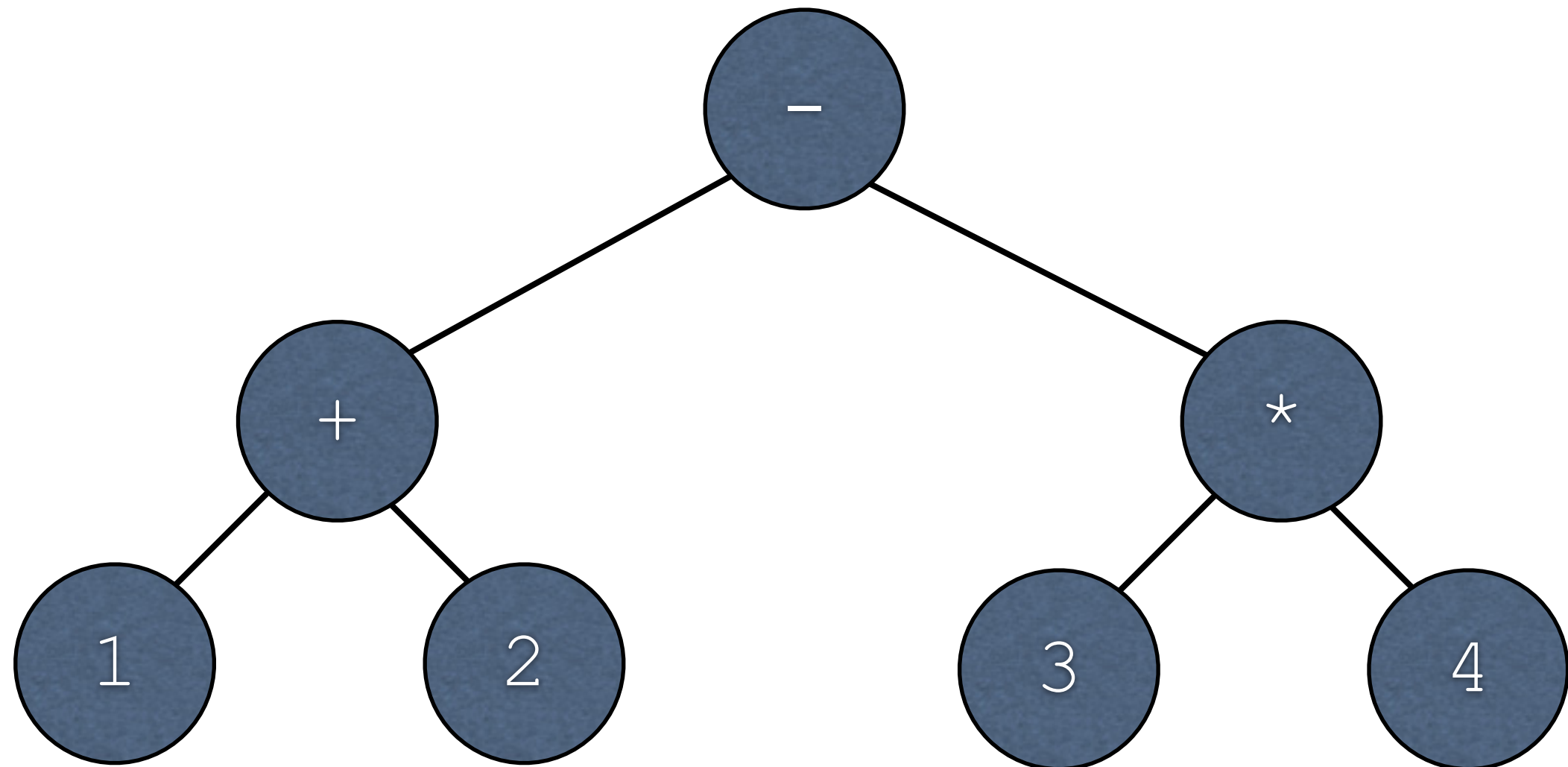


$$(1 + 2) - 3 * 4$$

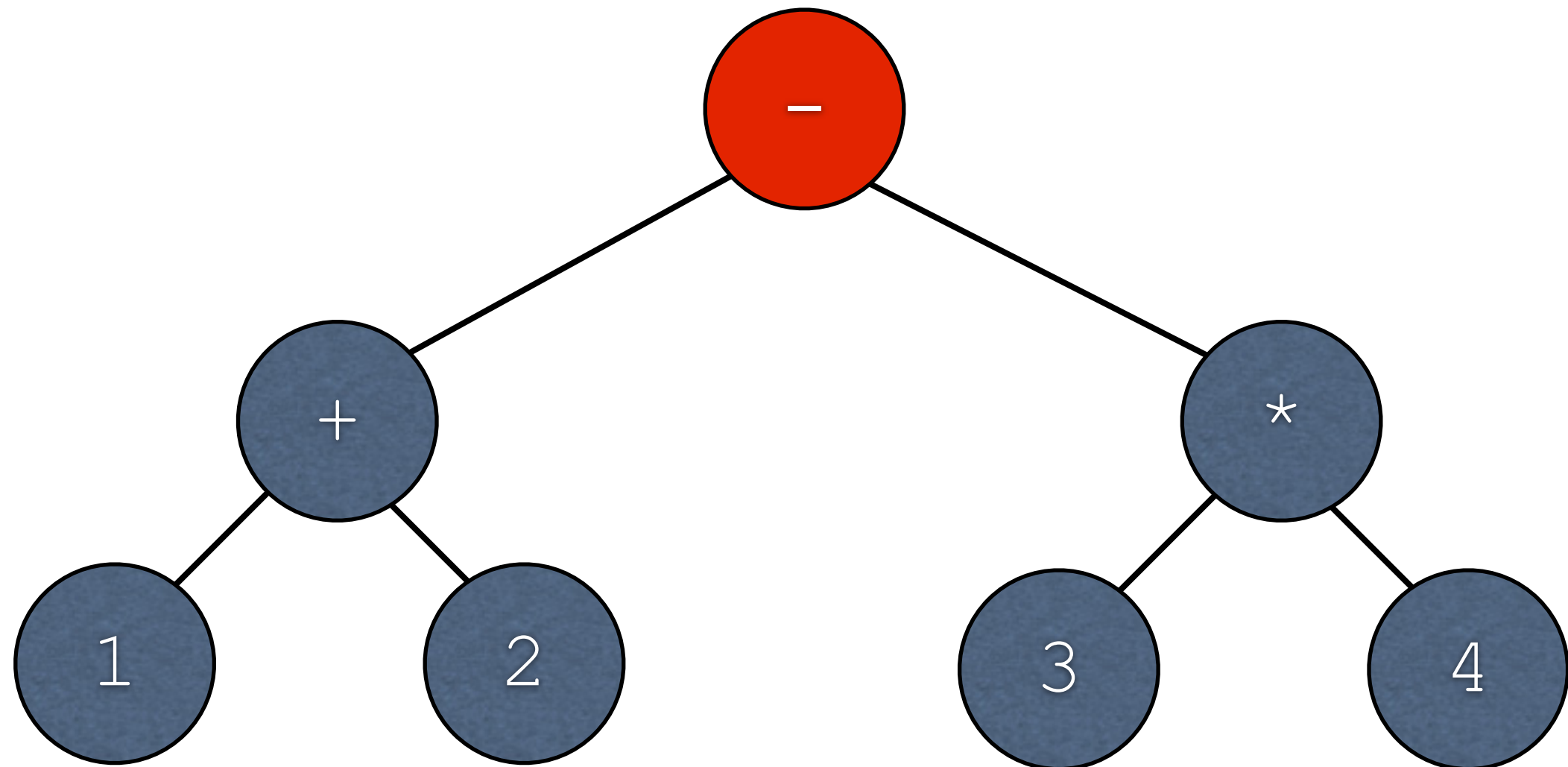


Exercise: First Side of AST/Evaluation Sheet

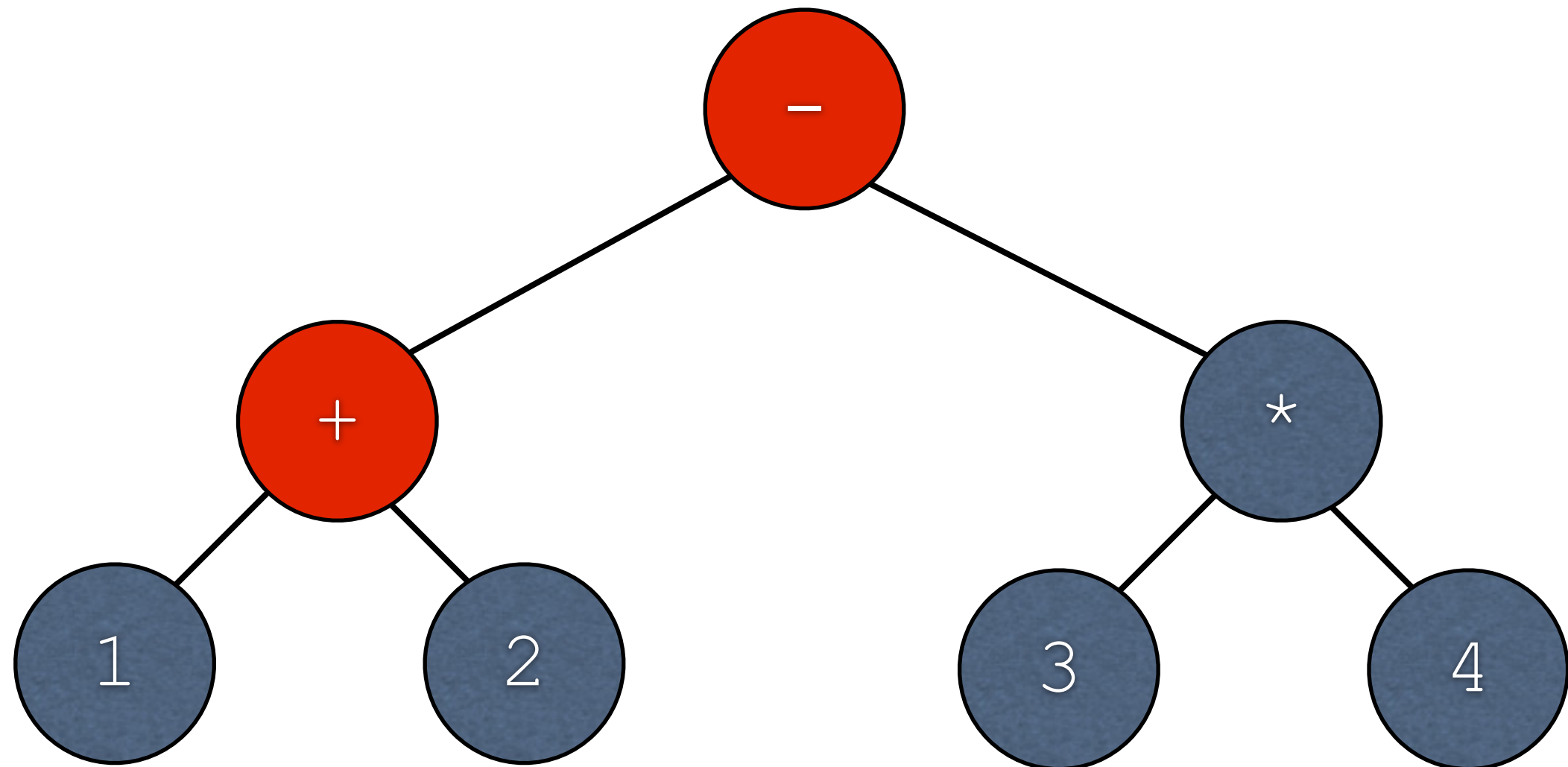
Evaluation



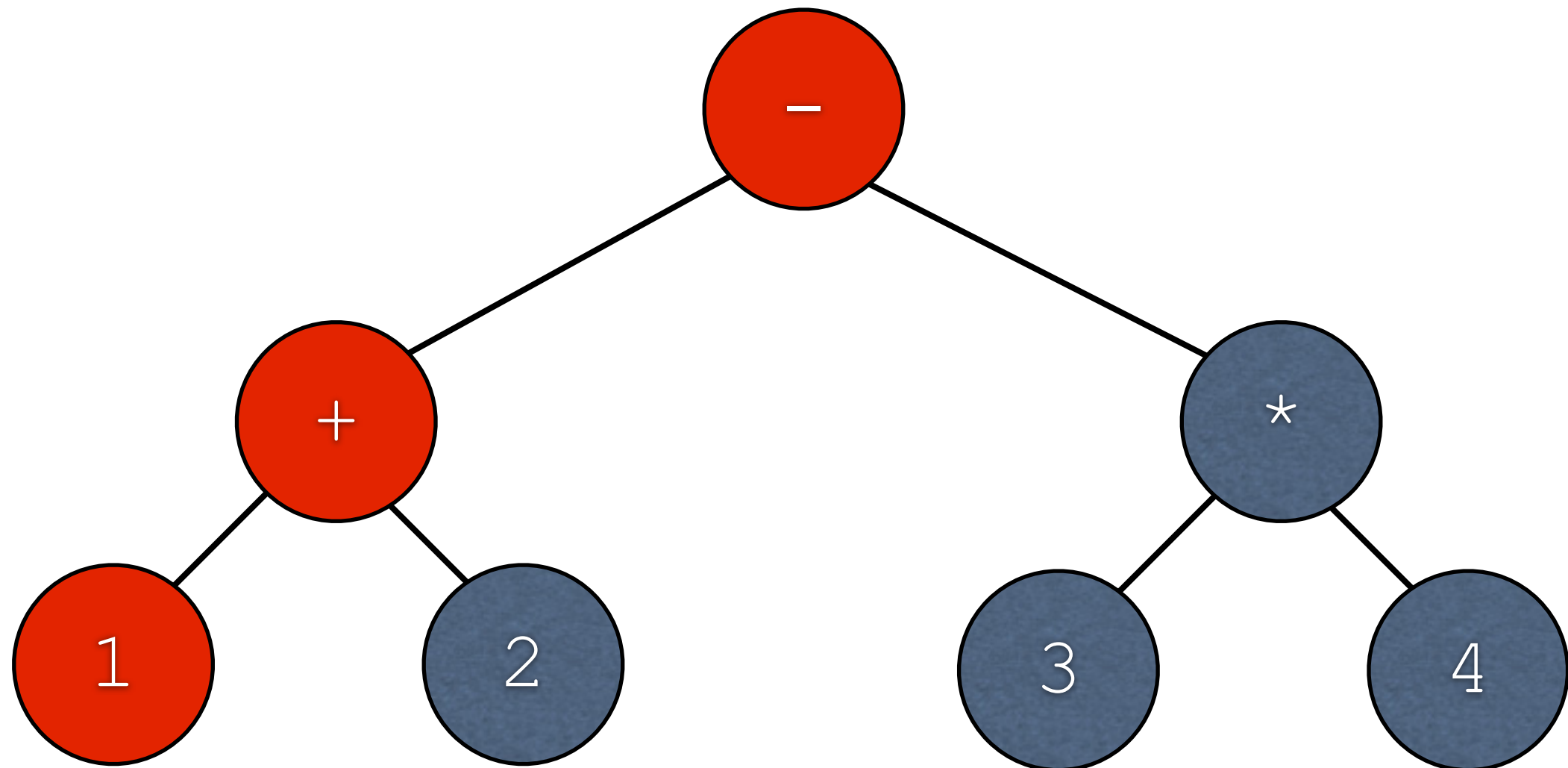
Evaluation



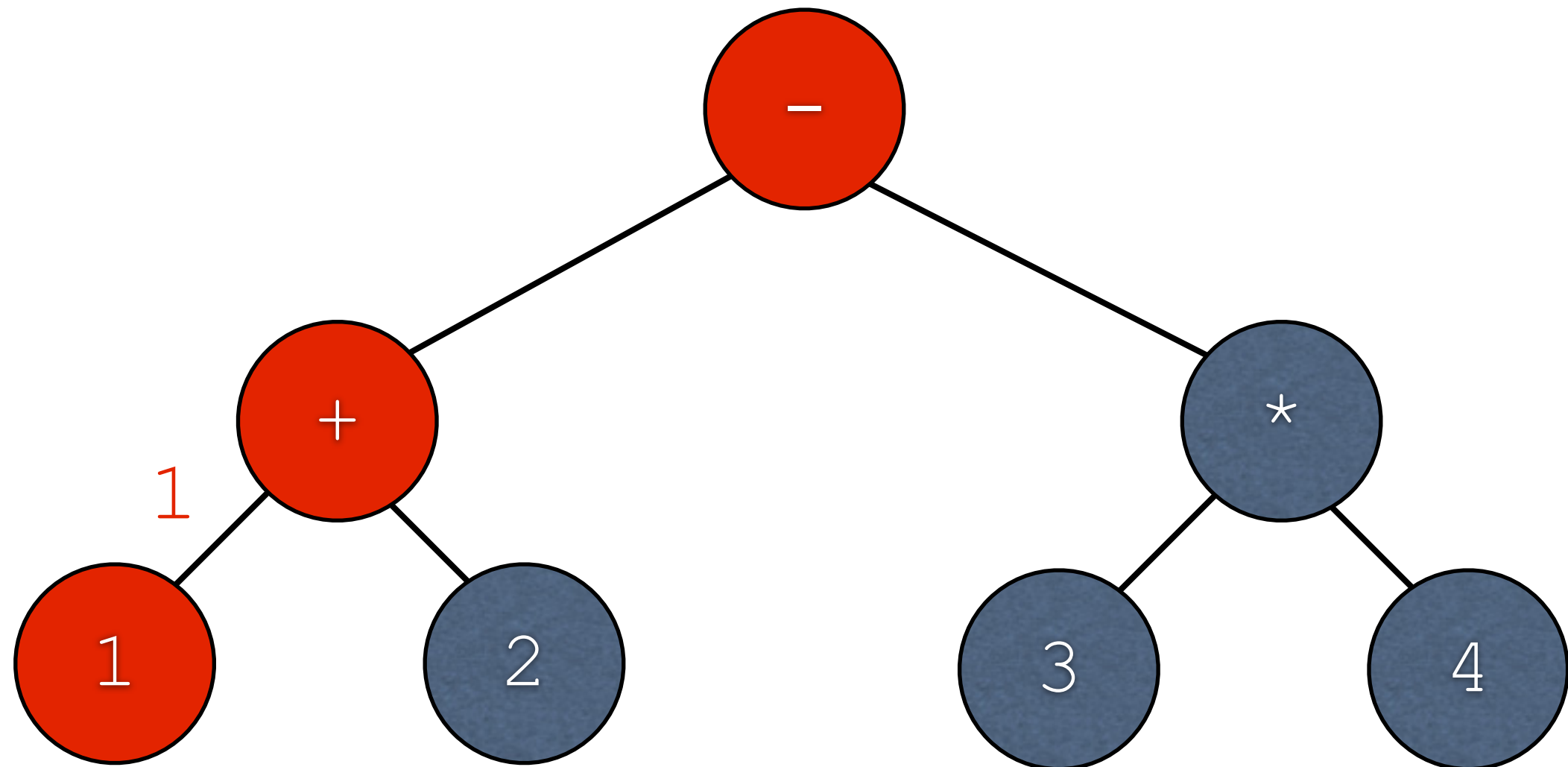
Evaluation



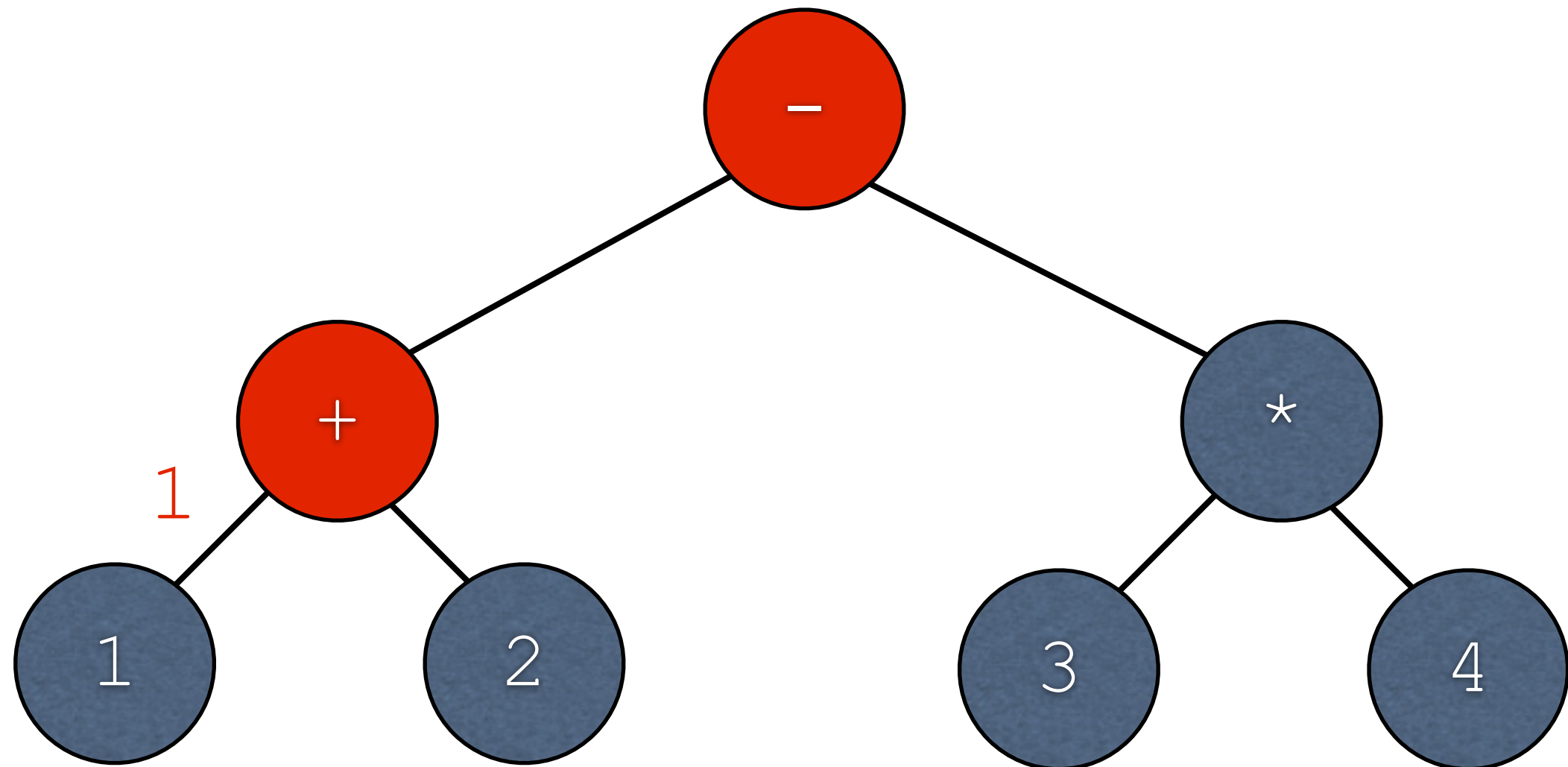
Evaluation



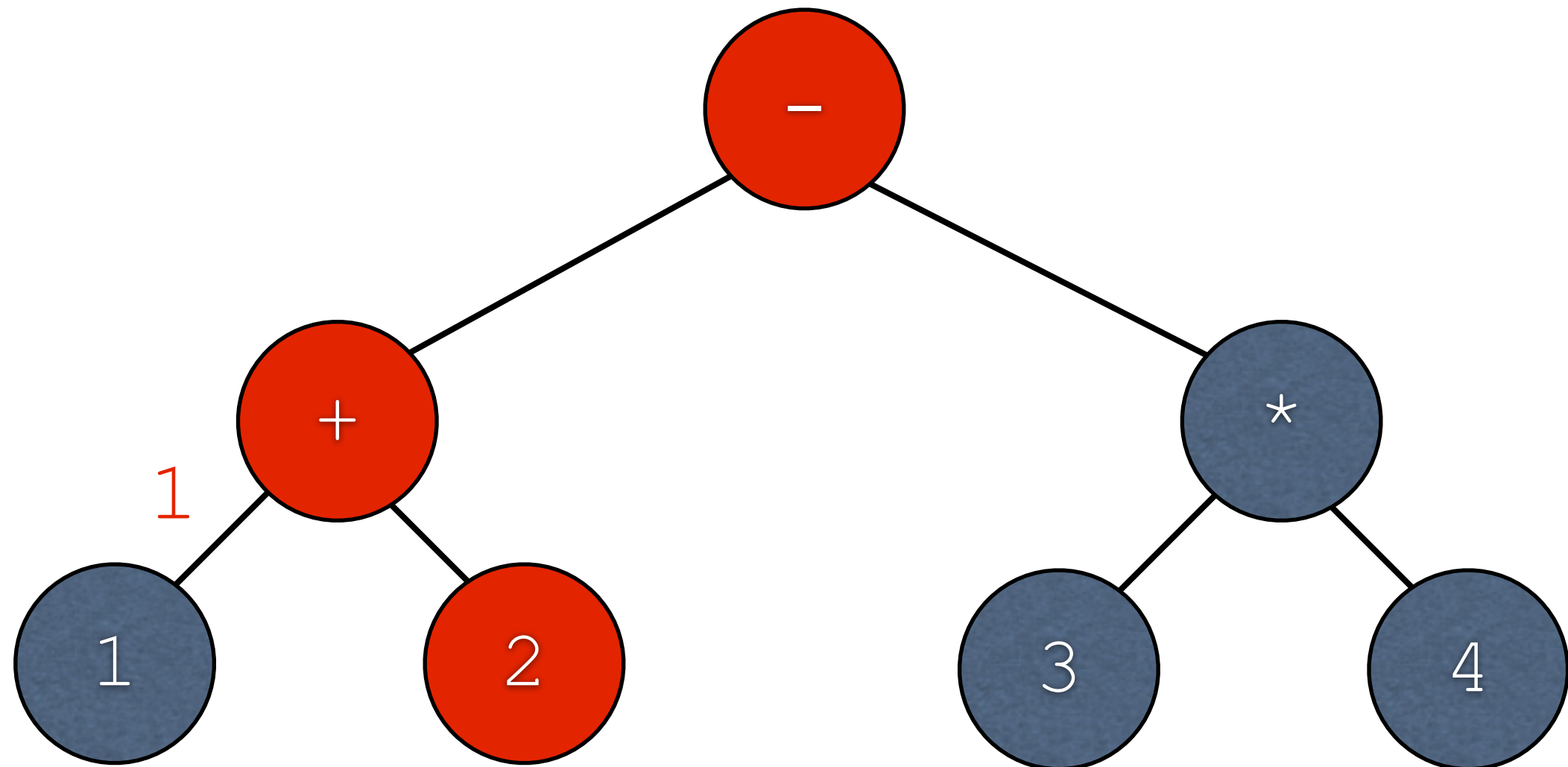
Evaluation



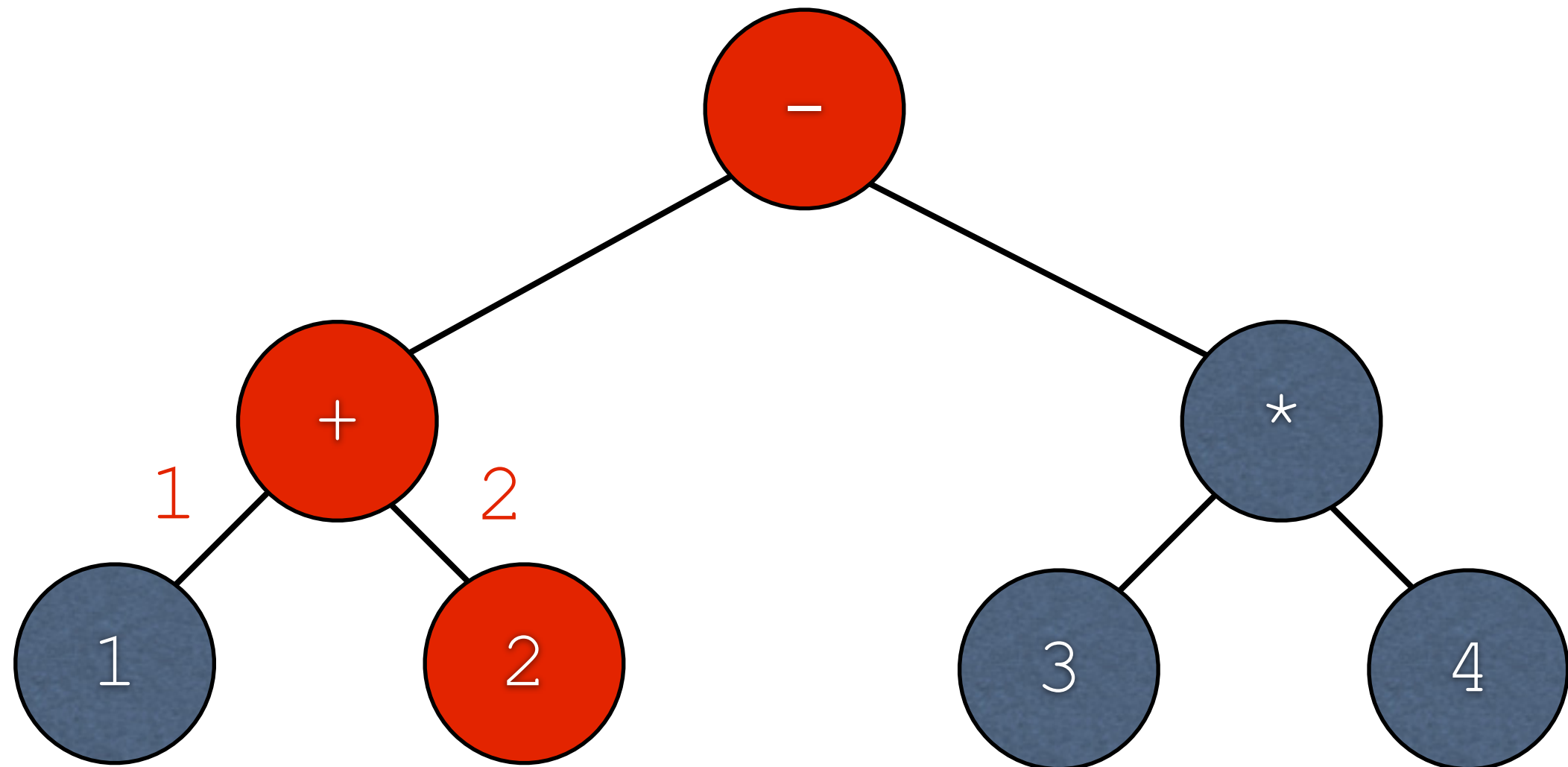
Evaluation



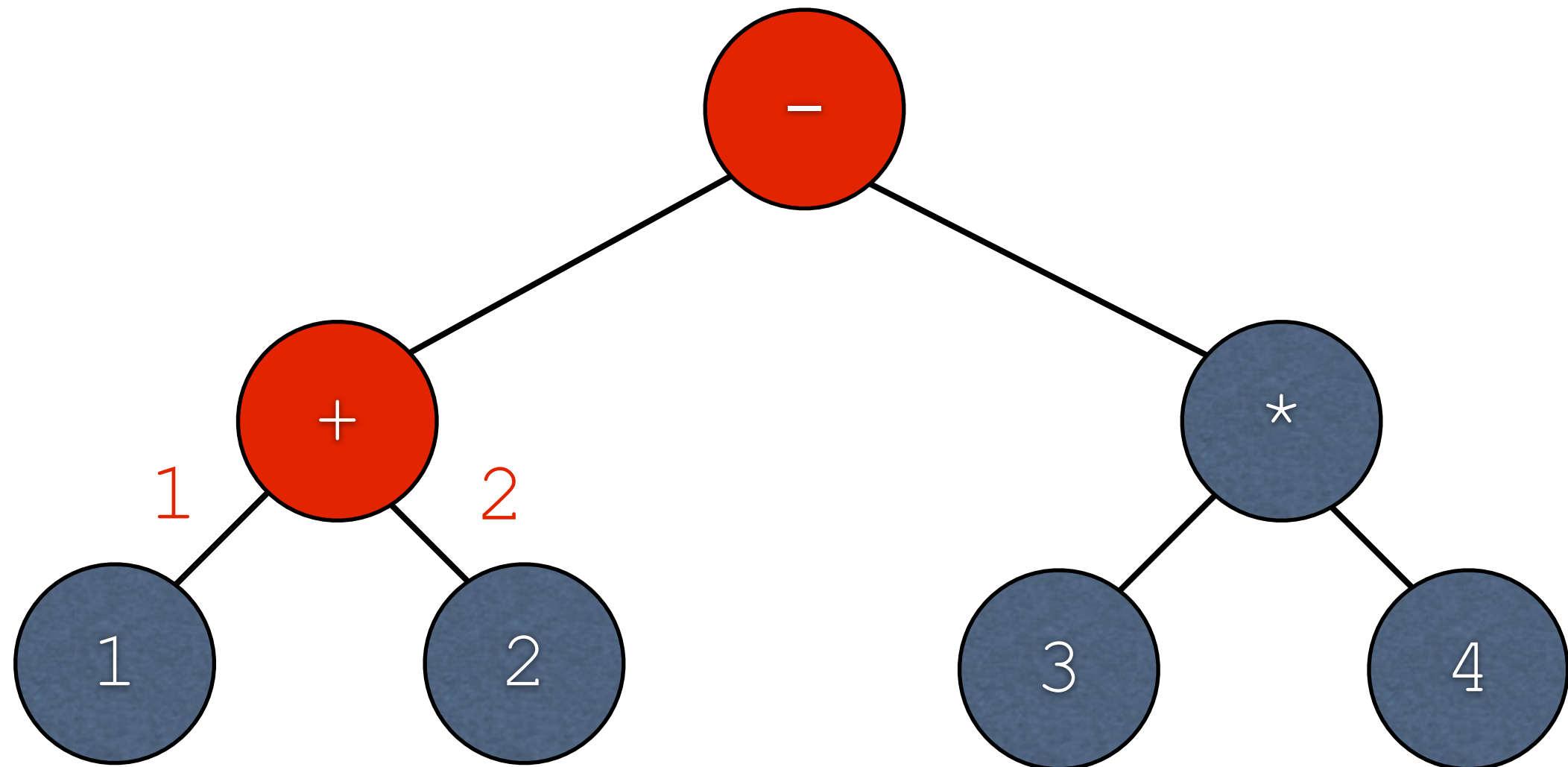
Evaluation



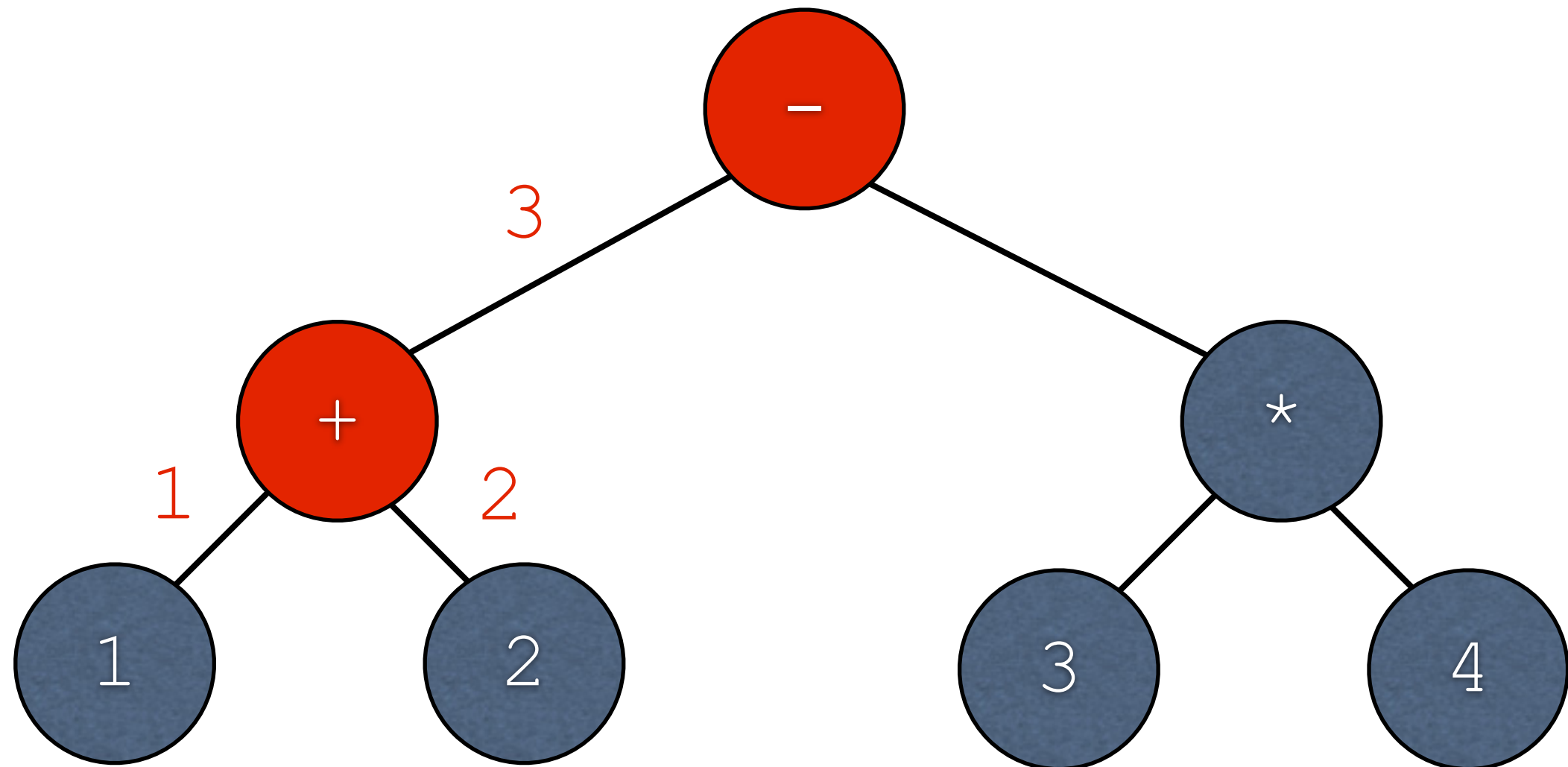
Evaluation



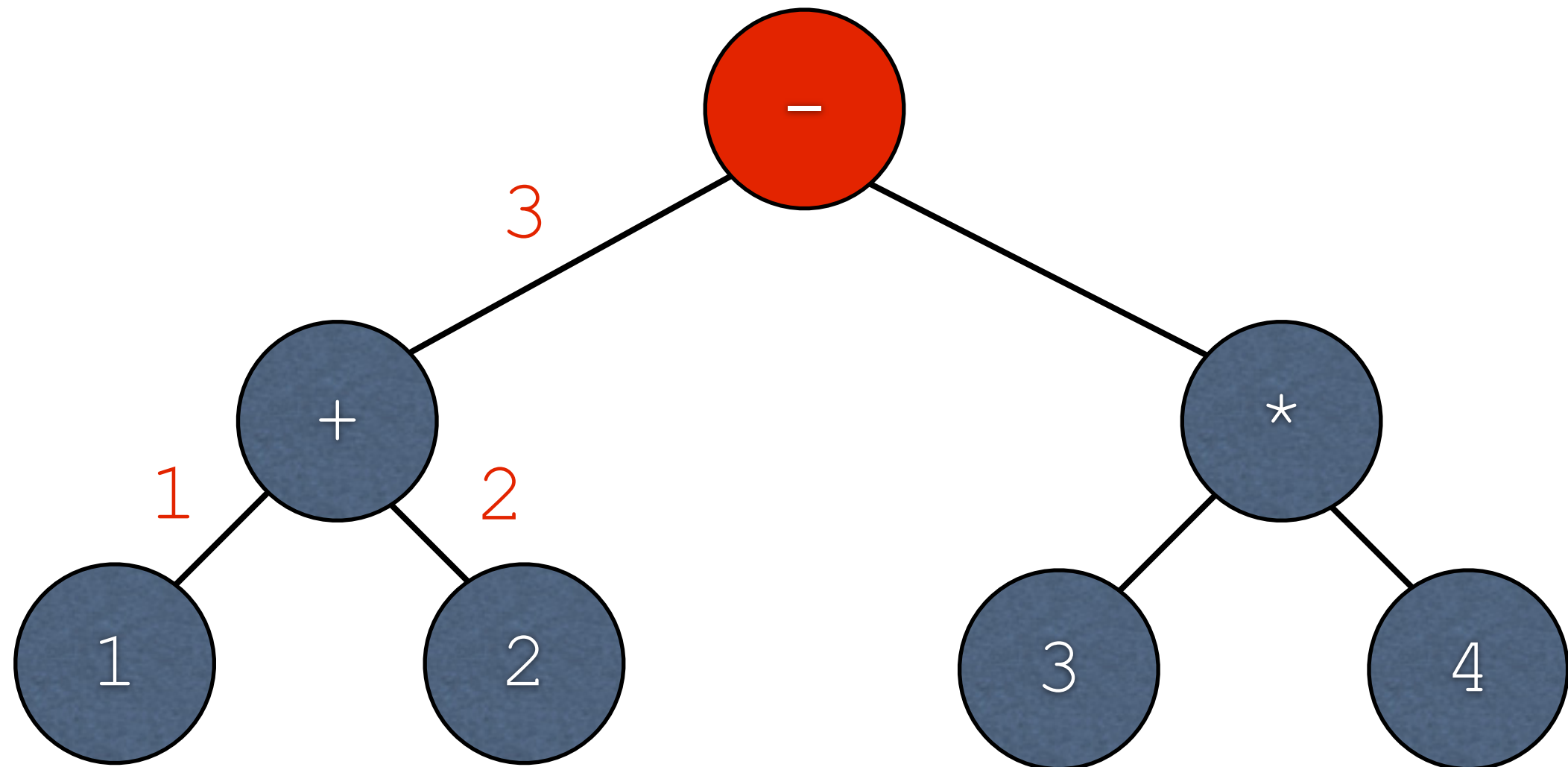
Evaluation



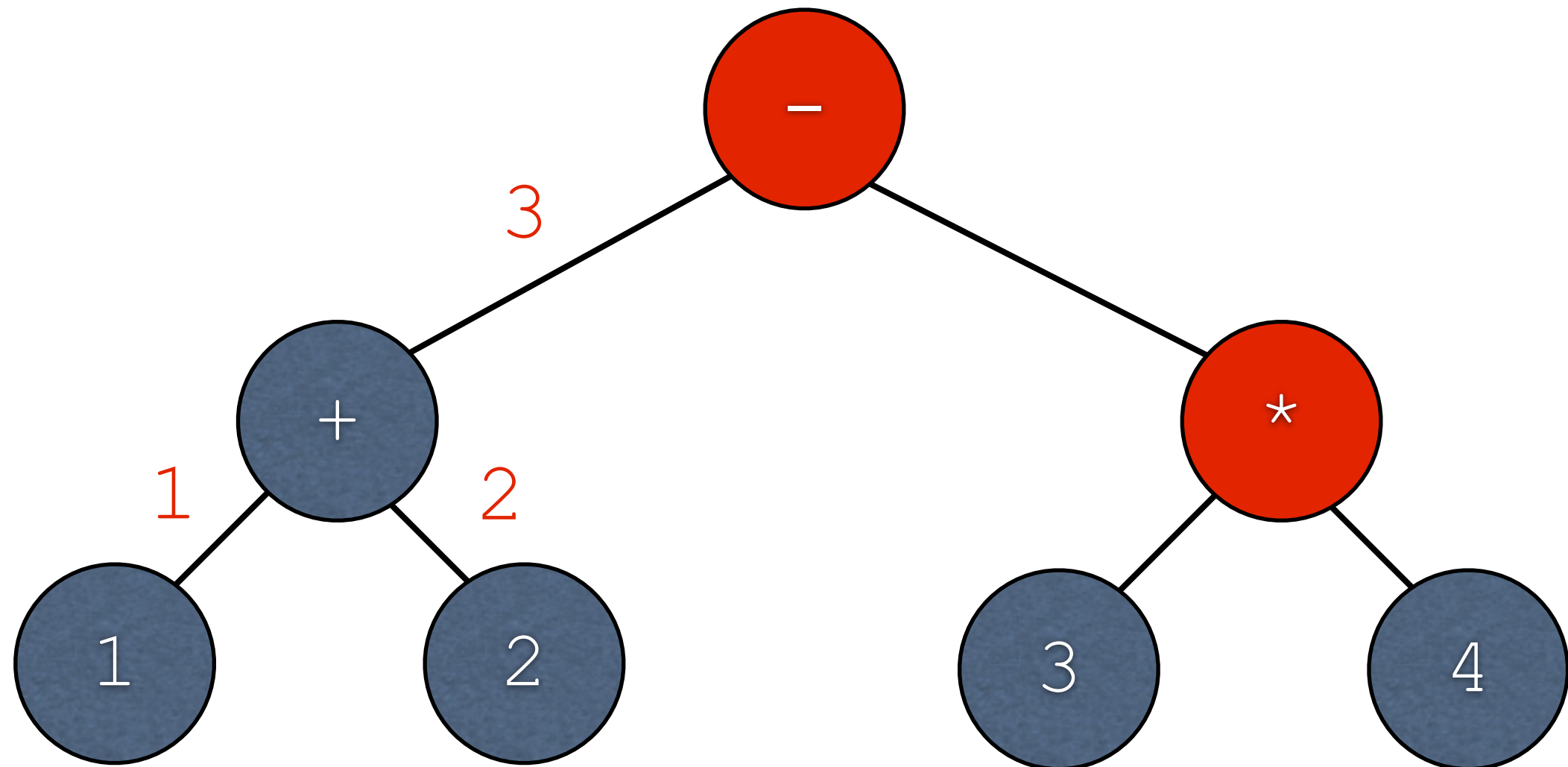
Evaluation



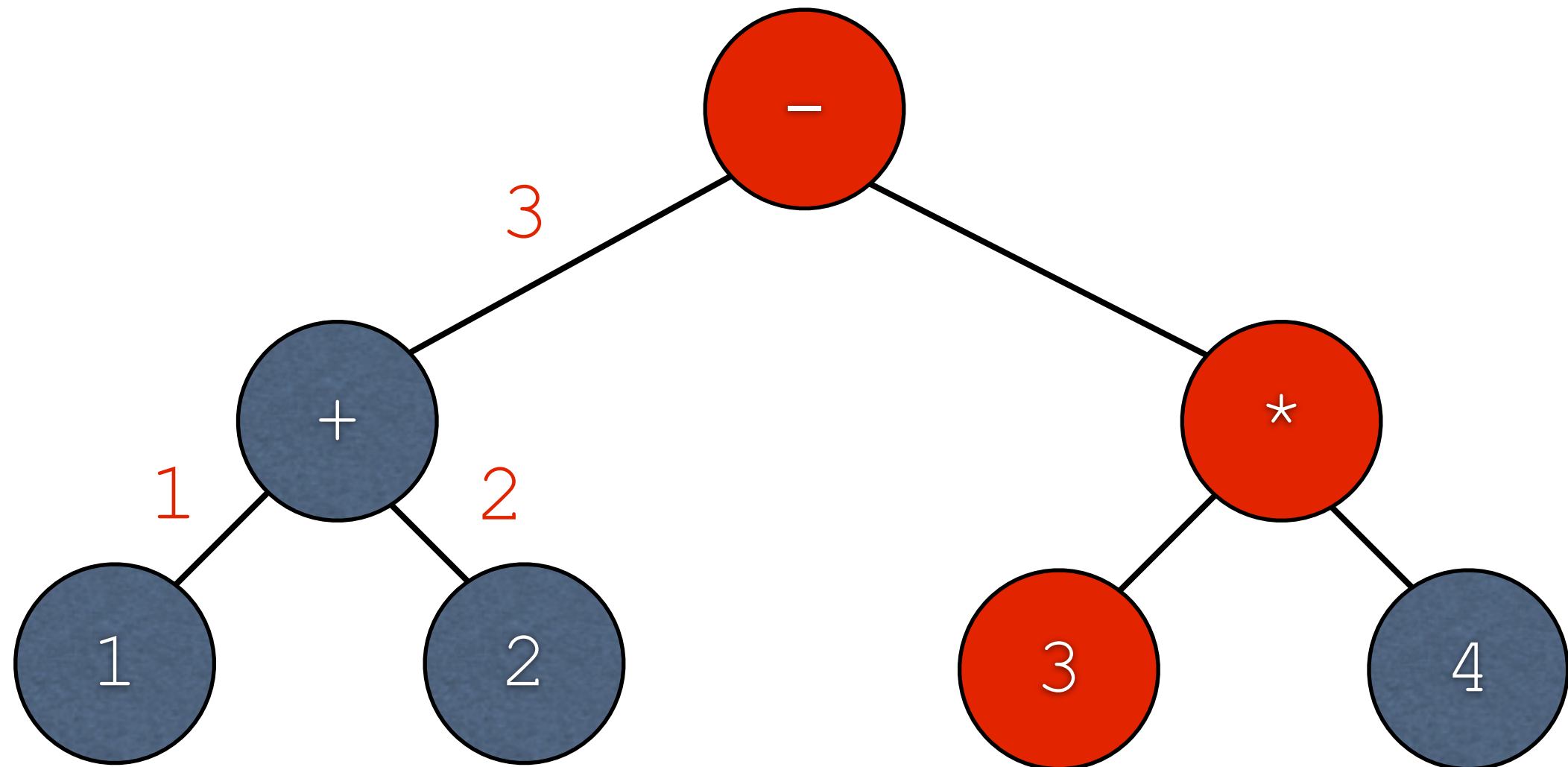
Evaluation



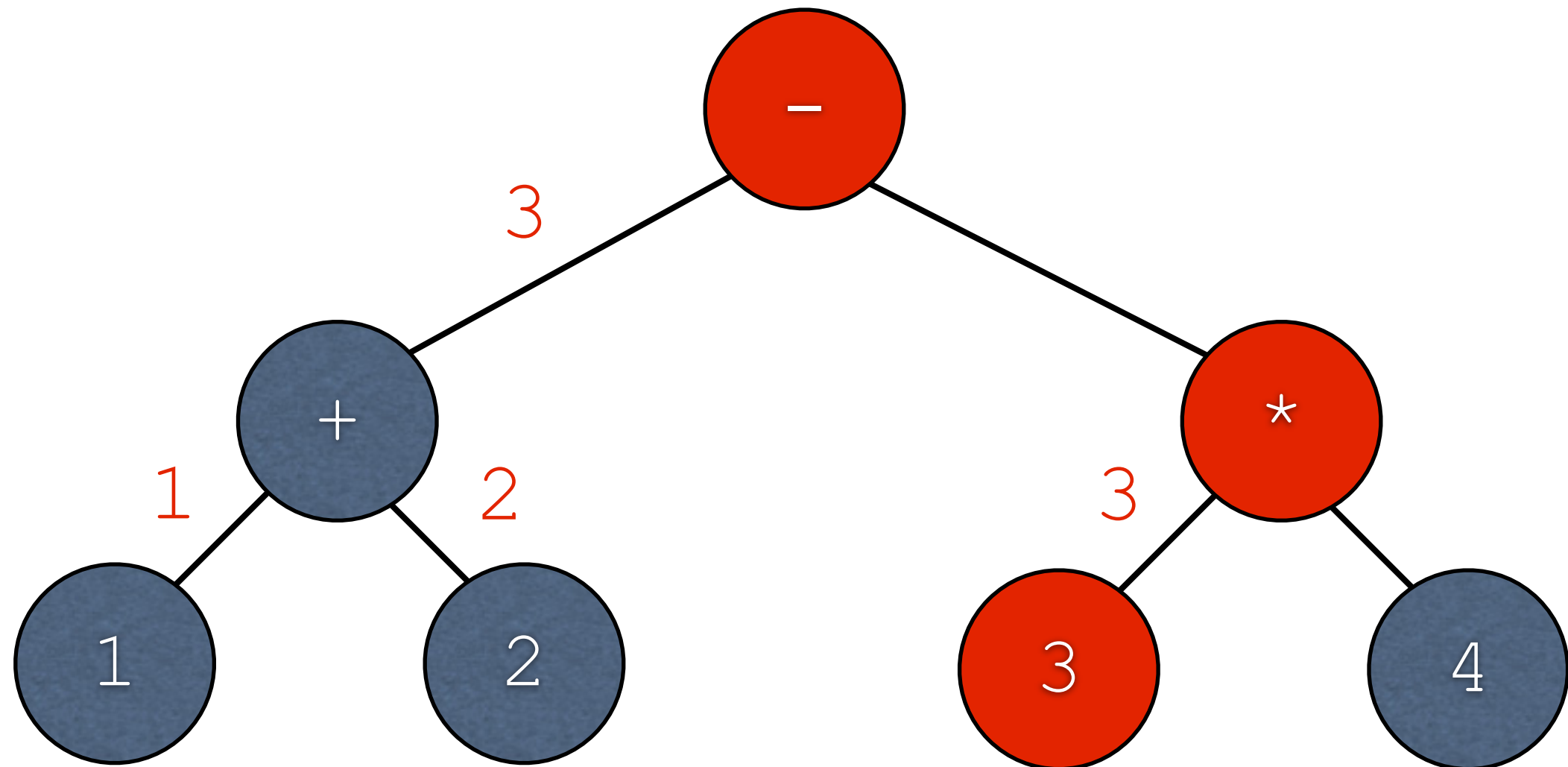
Evaluation



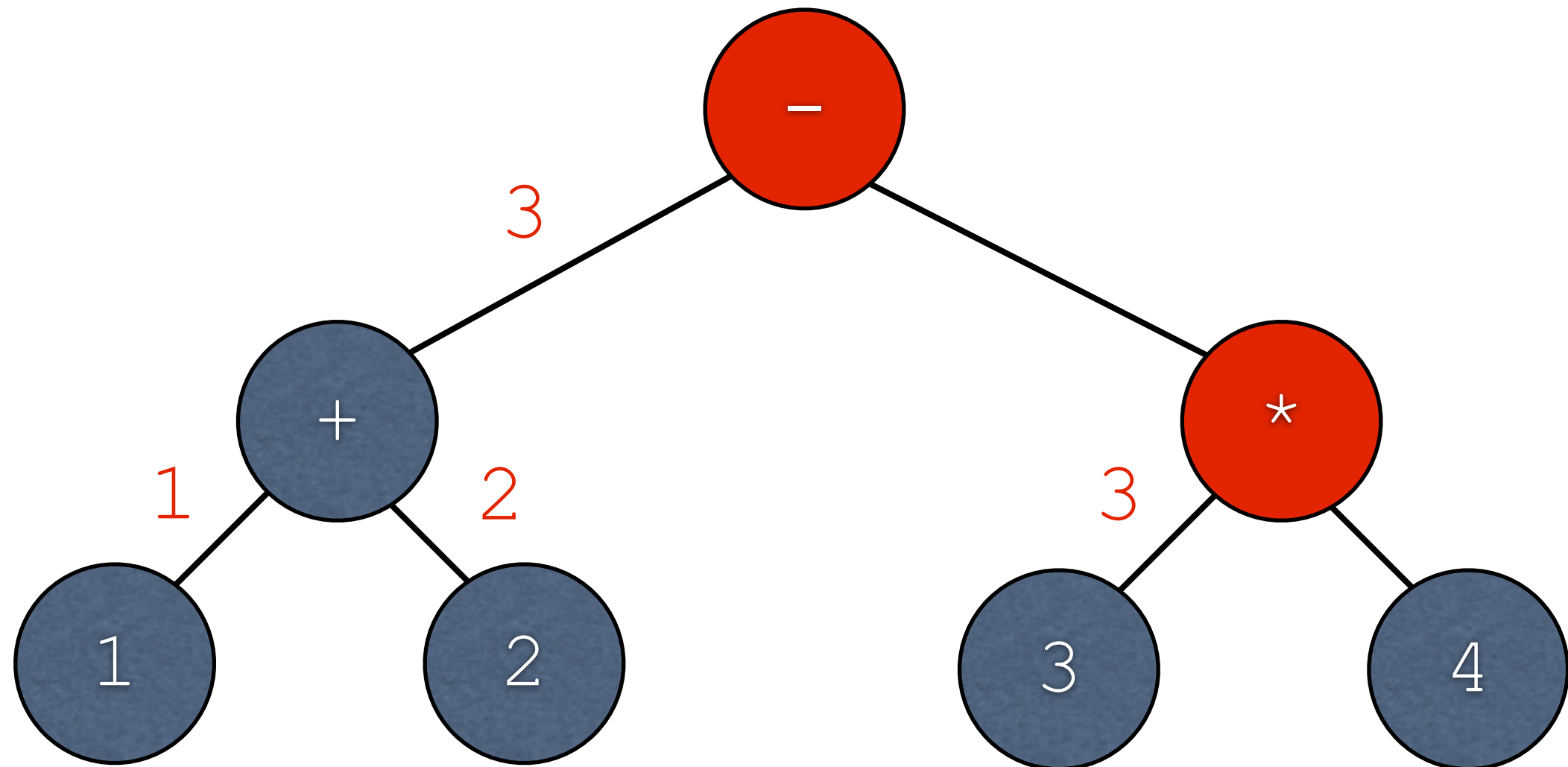
Evaluation



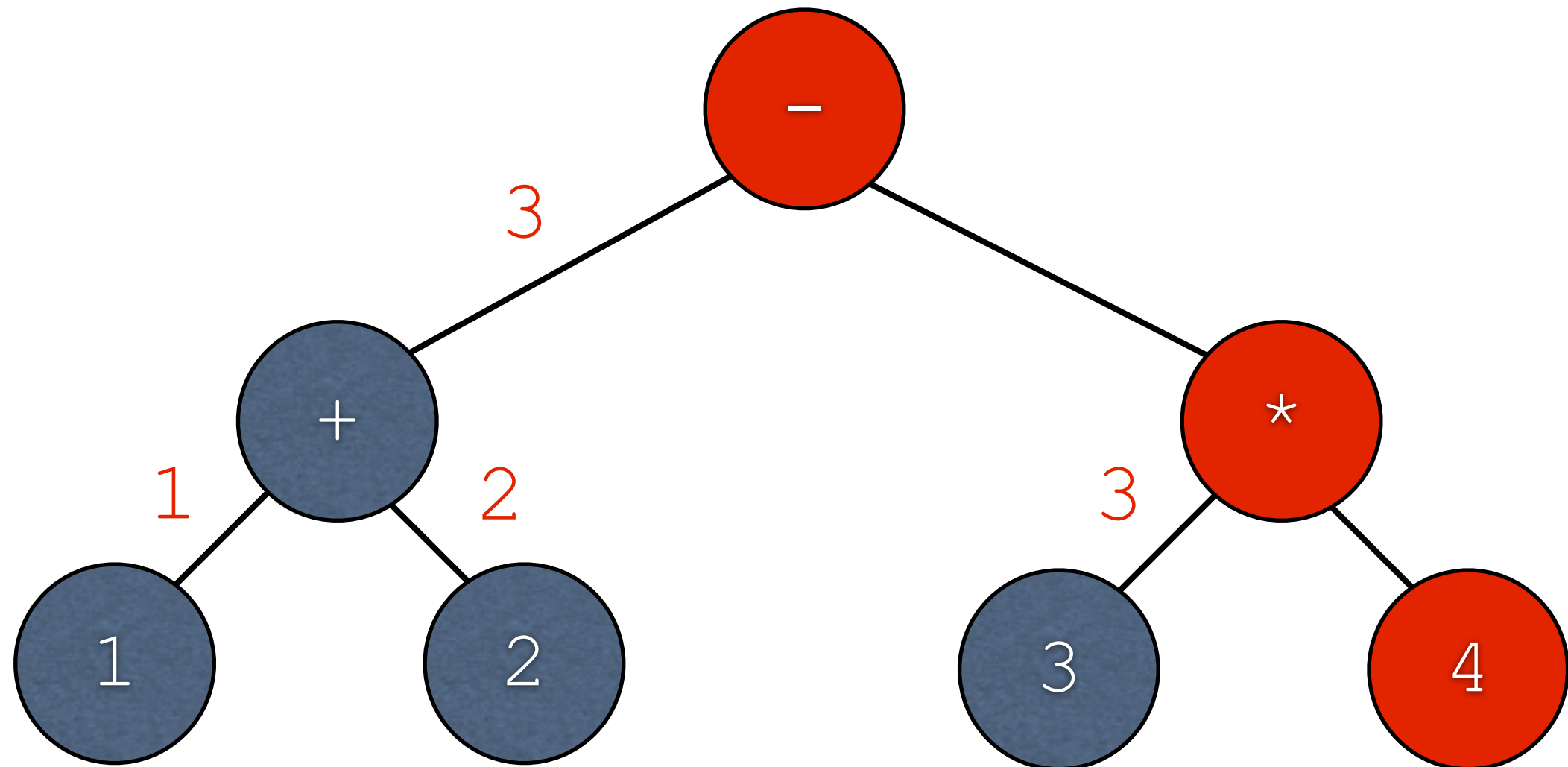
Evaluation



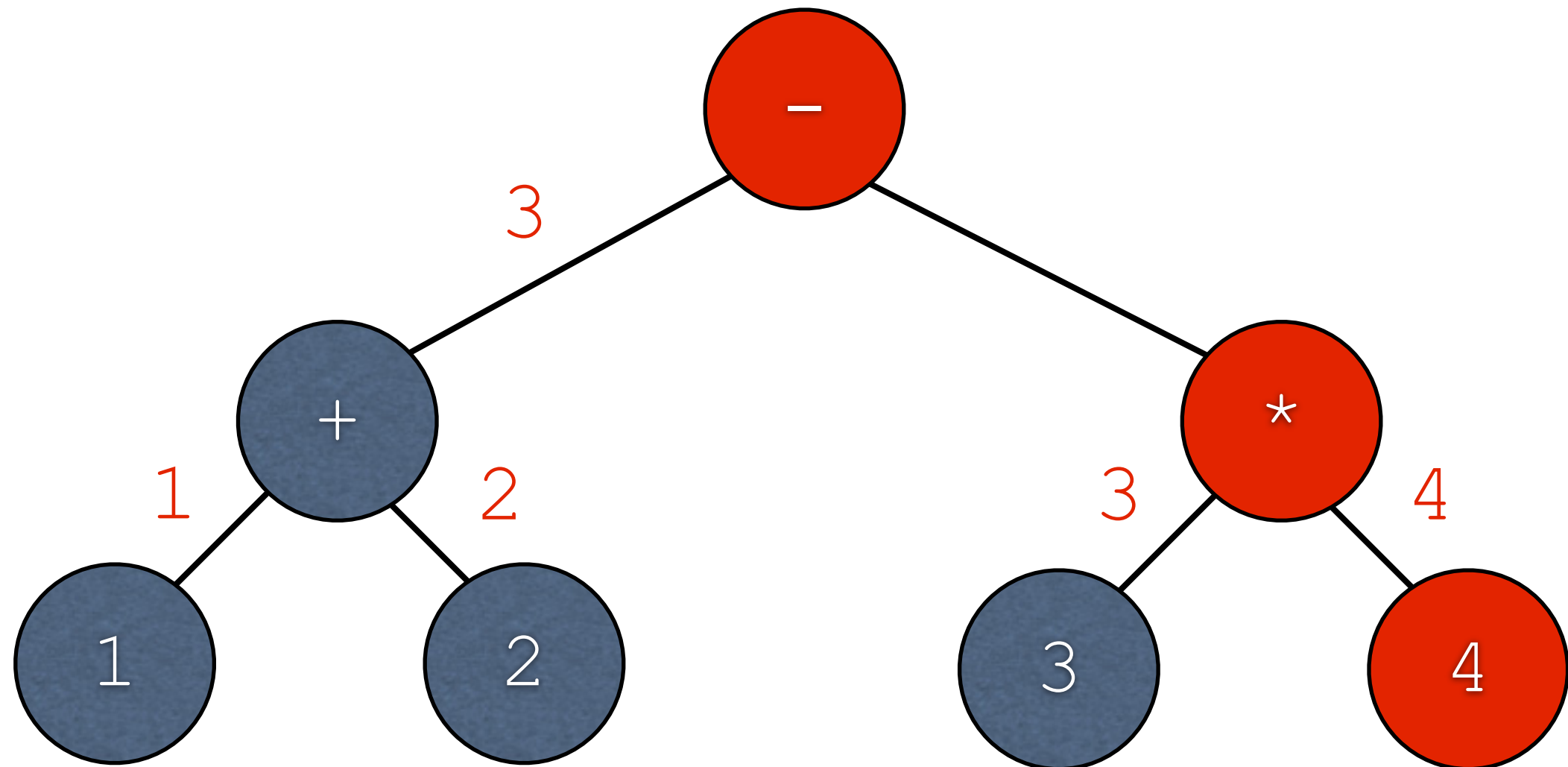
Evaluation



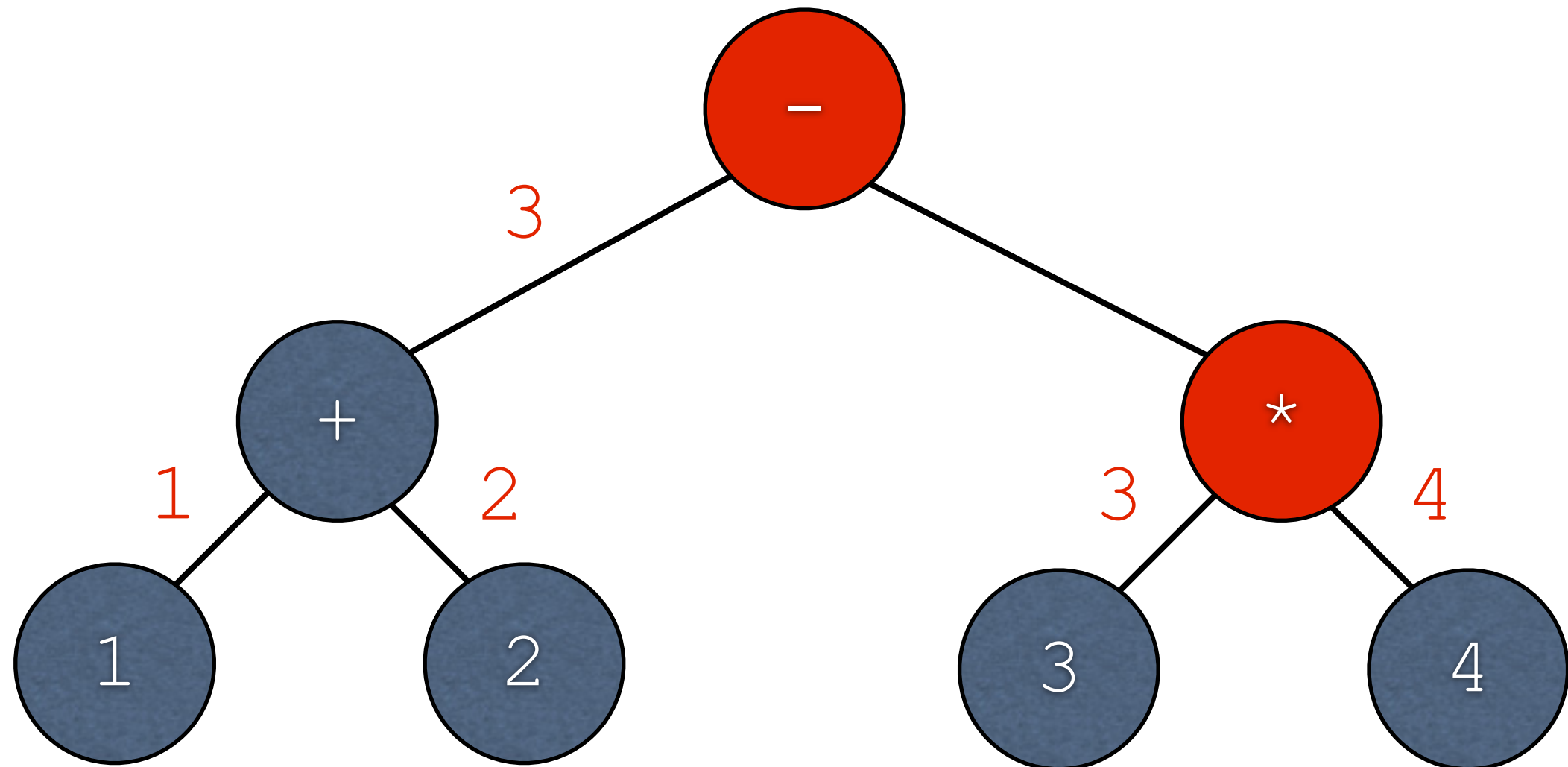
Evaluation



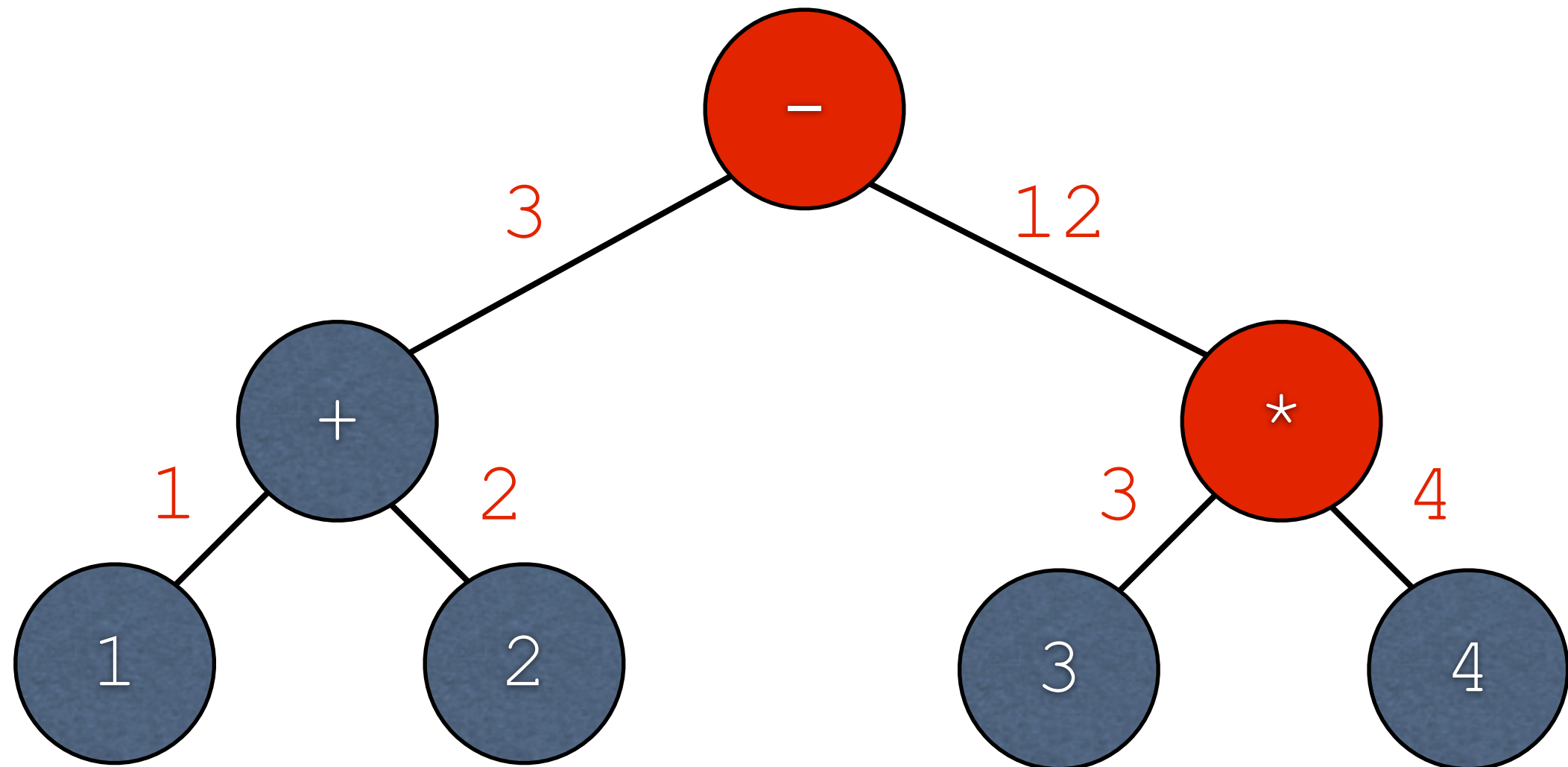
Evaluation



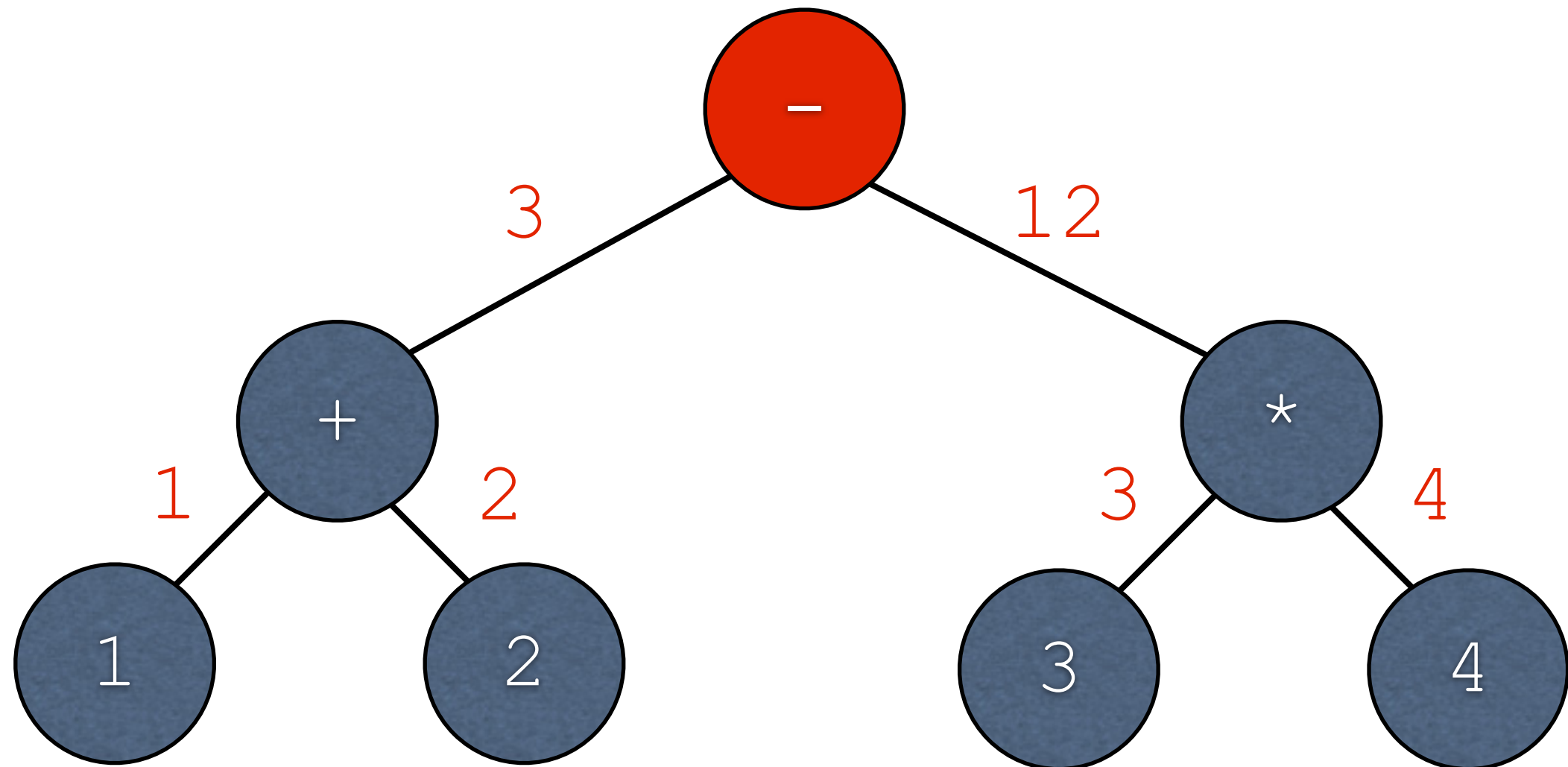
Evaluation



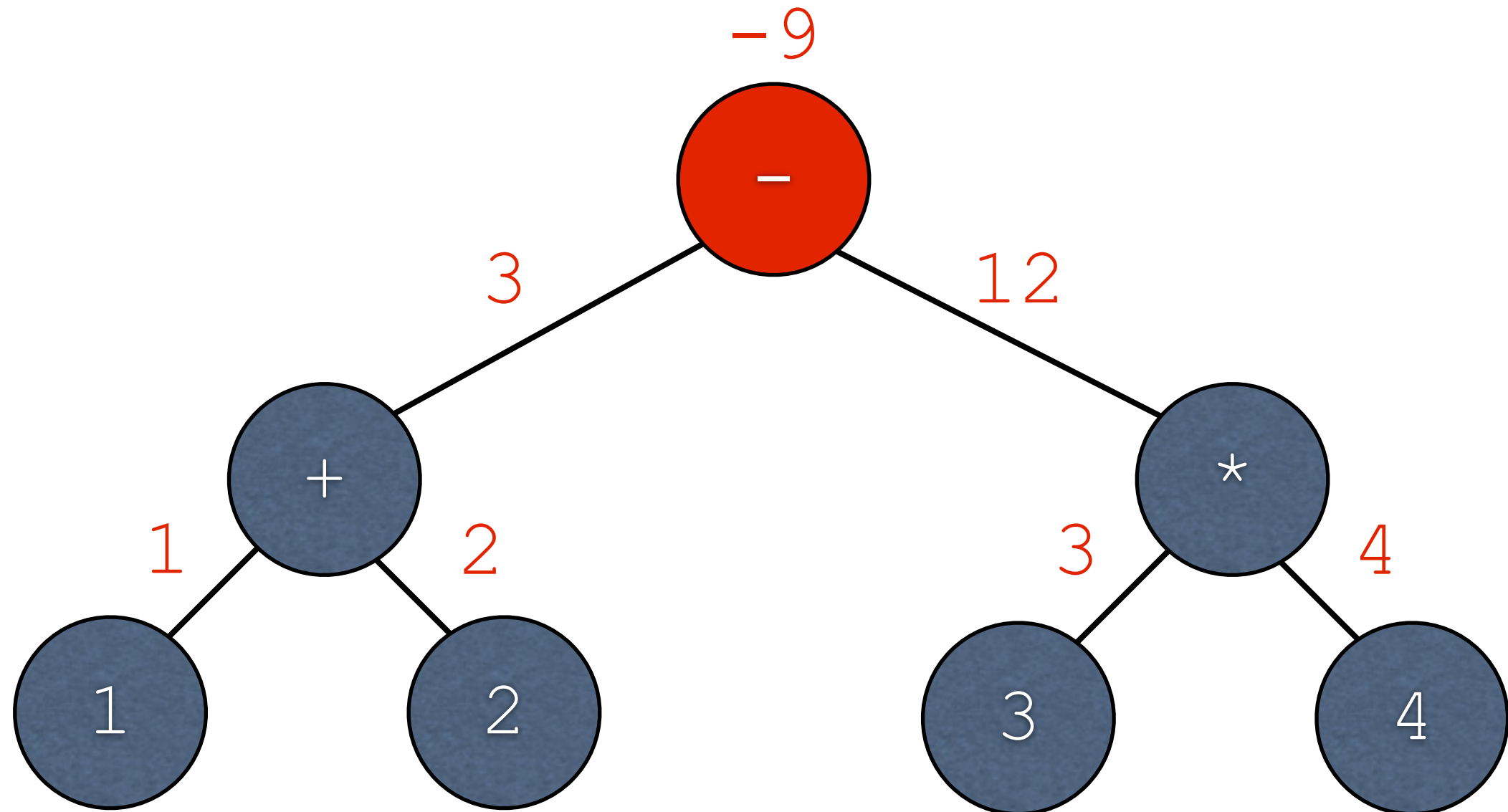
Evaluation



Evaluation



Evaluation



Exercise: Second Side of AST/Evaluation Sheet

Evaluator Example:

`arithmetic_evaluator.py`