



Order of operations

In mathematics and computer programming, the **order of operations** is a collection of rules that reflect conventions about which operations to perform first in order to evaluate a given mathematical expression.

These rules are formalized with a ranking of the operations. The rank of an operation is called its **precedence**, and an operation with a *higher* precedence is performed before operations with *lower* precedence. Calculators generally perform operations with the same precedence from left to right,^[1] but some programming languages and calculators adopt different conventions.

For example, multiplication is granted a higher precedence than addition, and it has been this way since the introduction of modern algebraic notation.^{[2][3]} Thus, in the expression $1 + 2 \times 3$, the multiplication is performed before addition, and the expression has the value $1 + (2 \times 3) = 7$, and not $(1 + 2) \times 3 = 9$. When exponents were introduced in the 16th and 17th centuries, they were given precedence over both addition and multiplication and placed as a superscript to the right of their base.^[2] Thus $3 + 5^2 = 28$ and $3 \times 5^2 = 75$.

These conventions exist to avoid notational ambiguity while allowing notation to remain brief.^[4] Where it is desired to override the precedence conventions, or even simply to emphasize them, parentheses () can be used. For example, $(2 + 3) \times 4 = 20$ forces addition to precede multiplication, while $(3 + 5)^2 = 64$ forces addition to precede exponentiation. If multiple pairs of parentheses are required in a mathematical expression (such as in the case of nested parentheses), the parentheses may be replaced by brackets or braces to avoid confusion, as in $[2 \times (3 + 4)] - 5 = 9$.

These rules are meaningful only when the usual notation (called infix notation) is used. When functional or Polish notation are used for all operations, the order of operations results from the notation itself.

Conventional order

The order of operations, that is, the order in which the operations in an expression are usually performed, results from a convention adopted throughout mathematics, science, technology and many computer programming languages. It is summarized as:^{[2][5]}

1. Parentheses
2. Exponentiation
3. Multiplication and division
4. Addition and subtraction

This means that to evaluate an expression, one first evaluates any sub-expression inside parentheses, working inside to outside if there is more than one set. Whether inside parenthesis or not, the operation that is higher in the above list should be applied first. Operations of the same precedence are conventionally evaluated from left to right.

If each division is replaced with multiplication by the reciprocal (multiplicative inverse) then the associative and commutative laws of multiplication allow the factors in each term to be multiplied together in any order. Sometimes multiplication and division are given equal precedence, or sometimes multiplication is given

higher precedence than division; see § Mixed division and multiplication below. If each subtraction is replaced with addition of the opposite (additive inverse), then the associative and commutative laws of addition allow terms to be added in any order.

The root symbol $\sqrt{}$ is traditionally prolonged by a bar (called vinculum) over the radicand (this avoids the need for parentheses around the radicand). Other functions use parentheses around the input to avoid ambiguity.^{[6][7][a]} The parentheses can be omitted if the input is a single numerical variable or constant,^[2] as in the case of $\sin x = \sin(x)$ and $\sin \pi = \sin(\pi)$.^[a] Traditionally this convention extends to monomials; thus, $\sin 3x = \sin(3x)$ and even $\sin \frac{1}{2}xy = \sin(xy/2)$, but $\sin x + y = \sin(x) + y$, because $x + y$ is not a monomial. However, this convention is not universally understood, and some authors prefer explicit parentheses.^[b] Some calculators and programming languages require parentheses around function inputs, some do not.

Symbols of grouping can be used to override the usual order of operations.^[2] Grouped symbols can be treated as a single expression.^[2] Symbols of grouping can be removed using the associative and distributive laws, also they can be removed if the expression inside the symbol of grouping is sufficiently simplified so no ambiguity results from their removal.

Examples

Multiplication before addition:

$$1 + 2 \times 3 = 1 + 6 = 7.$$

Parenthetical subexpressions are evaluated first:

$$(1 + 2) \times 3 = 3 \times 3 = 9.$$

Exponentiation before multiplication, multiplication before subtraction:

$$1 - 2 \times 3^4 = 1 - 2 \times 81 = 1 - 162 = -161.$$

When an expression is written as a superscript, the superscript is considered to be grouped by its position above its base:

$$1 + 2^{3+4} = 1 + 2^7 = 1 + 128 = 129.$$

The operand of a root symbol is determined by the overbar:

$$\sqrt{1+3} + 5 = \sqrt{4} + 5 = 2 + 5 = 7.$$

A horizontal fractional line also acts as a symbol of grouping:

$$\frac{1+2}{3+4} + 5 = \frac{3}{7} + 5.$$

Parentheses can be nested, and should be evaluated from the inside outward. For legibility, outer parentheses can be made larger than inner parentheses. Alternately, other grouping symbols, such as curly braces $\{ \}$ or square brackets $[]$, are sometimes used along with parentheses $()$. For example:

$$[(1 + 2) \div (3 + 4)] + 5 = (3 \div 7) + 5$$

Special cases

Unary minus sign

There are differing conventions concerning the unary operation '-' (usually pronounced "minus"). In written or printed mathematics, the expression -3^2 is interpreted to mean $-(3^2) = -9$.^{[2][8]}

In some applications and programming languages, notably Microsoft Excel, PlanMaker (and other spreadsheet applications) and the programming language bc, unary operations have a higher priority than binary operations, that is, the unary minus has higher precedence than exponentiation, so in those languages -3^2 will be interpreted as $(-3)^2 = 9$.^[9] This does not apply to the binary minus operation '-'; for example in Microsoft Excel while the formulas $=-2^2$, $=-(2)^2$ and $=0+-2^2$ return 4, the formulas $=0-2^2$ and $=-(2^2)$ return -4.

Mixed division and multiplication

There is no universal convention for interpreting a term containing both division denoted by '÷' and multiplication denoted by '×'. Proposed conventions include assigning the operations equal precedence and evaluating them from left to right, or equivalently treating division as multiplication by the reciprocal and then evaluating in any order;^[10] evaluating all multiplications first followed by divisions from left to right; or eschewing such expressions and instead always disambiguating them by explicit parentheses.^[11]

Beyond primary education, the symbol '÷' for division is seldom used, but is replaced by the use of algebraic fractions,^[12] typically written vertically with the numerator stacked above the denominator – which makes grouping explicit and unambiguous – but sometimes written inline using the slash or solidus symbol, '/'.^[13]

Multiplication denoted by juxtaposition (also known as implied multiplication) creates a visual unit and has higher precedence than most other operations. In academic literature, when inline fractions are combined with implied multiplication without explicit parentheses, the multiplication is conventionally interpreted as having higher precedence than division, so that e.g. $1/2n$ is interpreted to mean $1/(2 \cdot n)$ rather than $(1/2) \cdot n$.^{[2][10][14][15]} For instance, the manuscript submission instructions for the *Physical Review* journals directly state that multiplication has precedence over division,^[16] and this is also the convention observed in physics textbooks such as the *Course of Theoretical Physics* by Landau and Lifshitz^[c] and mathematics textbooks such as *Concrete Mathematics* by Graham, Knuth, and Patashnik.^[17] However, some authors recommend against expressions such as a/bc , preferring the explicit use of parenthesis $a/(bc)$.^[3]

More complicated cases are more ambiguous. For instance, the notation $1/2\pi(a+b)$ could plausibly mean either $1/[2\pi \cdot (a+b)]$ or $[1/(2\pi)] \cdot (a+b)$.^[18] Sometimes interpretation depends on context. The *Physical Review* submission instructions recommend against expressions of the form $a/b/c$; more explicit expressions $(a/b)/c$ or $a/(b/c)$ are unambiguous.^[16]

This ambiguity has been the subject of Internet memes such as " $8 \div 2(2+2)$ ", for which there are two conflicting interpretations: $8 \div [2 \cdot (2+2)] = 1$ and $(8 \div 2) \cdot (2+2) = 16$.^{[15][19]} Mathematics education researcher Hung-Hsi Wu points out that "one never gets a computation of this type in real life", and calls

such contrived examples "a kind of Gotcha! parlor game designed to trap an unsuspecting person by phrasing it in terms of a set of unreasonably convoluted rules."^[12]

Serial exponentiation

If exponentiation is indicated by stacked symbols using superscript notation, the usual rule is to work from the top down:^{[2][7]}

$$a^{b^c} = a^{(b^c)}$$

which typically is not equal to $(a^b)^c$. This convention is useful because there is a property of exponentiation that $(a^b)^c = a^{bc}$, so it's unnecessary to use serial exponentiation for this.

However, when exponentiation is represented by an explicit symbol such as a caret (^) or arrow (↑), there is no common standard. For example, Microsoft Excel and computation programming language MATLAB evaluate $a^b{}^c$ as $(a^b)^c$, but Google Search and Wolfram Alpha as $a^{(b^c)}$. Thus $4^3{}^2$ is evaluated to 4,096 in the first case and to 262,144 in the second case.

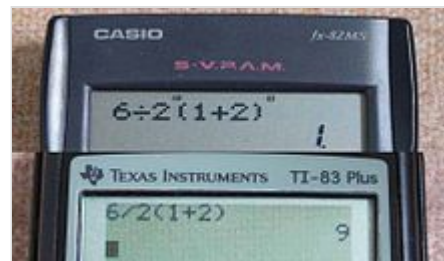
Mnemonics

Mnemonics are often used to help students remember the rules, involving the first letters of words representing various operations.^{[20][21]}

- The acronym **PEMDAS** is common in the United States^[22] and France.^[23] It stands for **P**arentheses, **E**xponents, **M**ultiplication/**D**ivision, **A**ddition/**S**ubtraction.^[24] PEMDAS is sometimes expanded to the mnemonic "Please Excuse My Dear Aunt Sally" in schools.^[25]
- **BEDMAS**, standing for **B**rackets, **E**xponents, **D**ivision/**M**ultiplication, **A**ddition/**S**ubtraction is common in Canada and New Zealand.^[26]
- The United Kingdom and other Commonwealth countries may use **BODMAS** meaning **B**rackets, **O**perations, **D**ivision/**M**ultiplication, **A**ddition/**S**ubtraction.^[26] Sometimes the **O** is expanded as "Of"^[d] or "Order" (i.e. powers/exponents or roots).^[27]
- **BIDMAS** is also used, standing for **B**rackets, **I**ndices, **D**ivision/**M**ultiplication, **A**ddition/**S**ubtraction.^[28]
- In Germany, the convention is simply taught as *Punktrechnung vor Strichrechnung*, dot operations prior line operations referring to the graphical shapes of the taught operator signs U+00B7 · MIDDLE DOT (multiplication), U+2236 ÷ RATIO (division), and U+002B + PLUS SIGN (addition), U+2212 − MINUS SIGN (subtraction).

These mnemonics may be misleading when written this way.^[25] For example, misinterpreting any of the above rules to mean "addition first, subtraction afterward" would incorrectly evaluate the expression^[25] $a - b + c$ as $a - (b + c)$, while the correct evaluation is $(a - b) + c$. These values are different when $c \neq 0$.

Mnemonic acronyms have been criticized for not developing a conceptual understanding of the order of operations, and not addressing student questions about its purpose or flexibility.^{[29][30]} Students learning the order of operations via mnemonic acronyms routinely make mistakes,^[31] as do some pre-service



$6 \div 2(1+2)$ is interpreted as $6 \div (2 \times (1+2))$ by a fx-82MS (upper), and $(6 \div 2) \times (1+2)$ by a TI-83 Plus calculator (lower), respectively.

teachers.^[32] Even when students correctly learn the acronym, a disproportionate focus on memorization of trivia crowds out substantive mathematical content.^[12] The acronym's procedural application does not match experts' intuitive understanding of mathematical notation: mathematical notation indicates groupings in ways other than parentheses or brackets and a mathematical expression is a tree-like hierarchy rather than a linearly "ordered" structure; furthermore, there is no single order by which mathematical expressions must be simplified or evaluated and no universal canonical simplification for any particular expression, and experts fluently apply valid transformations and substitutions in whatever order is convenient, so learning a rigid procedure can lead students to a misleading and limiting understanding of mathematical notation.^[33]

Calculators

Different calculators follow different orders of operations.^[2] Many simple calculators without a stack implement chain input, working in button-press order without any priority given to different operations, give a different result from that given by more sophisticated calculators. For example, on a simple calculator, typing $1 + 2 \times 3 =$ yields 9, while a more sophisticated calculator will use a more standard priority, so typing $1 + 2 \times 3 =$ yields 7.

Calculators may associate exponents to the left or to the right. For example, the expression $a^b{}^c$ is interpreted as $a^{(b^c)}$ on the TI-92 and the TI-30XS MultiView in "Mathprint mode", whereas it is interpreted as $(a^b)^c$ on the TI-30XII and the TI-30XS MultiView in "Classic mode".

An expression like $1/2x$ is interpreted as $1/(2x)$ by TI-82,^[3] as well as many modern Casio calculators^[34] (configurable on some like the fx-9750GIII), but as $(1/2)x$ by TI-83 and every other TI calculator released since 1996,^{[35][3]} as well as by all Hewlett-Packard calculators with algebraic notation. While the first interpretation may be expected by some users due to the nature of implied multiplication,^[36] the latter is more in line with the rule that multiplication and division are of equal precedence.^[3]

When the user is unsure how a calculator will interpret an expression, parentheses can be used to remove the ambiguity.^[3]

Order of operations arose due to the adaptation of infix notation in standard mathematical notation, which can be notationally ambiguous without such conventions, as opposed to postfix notation or prefix notation, which do not need orders of operations.^{[37][38]} Hence, calculators utilizing Reverse Polish notation (RPN) using a stack to enter expressions in the correct order of precedence do not need parentheses or any possibly model-specific order of execution.^{[25][24]}

Programming languages

Most programming languages use precedence levels that conform to the order commonly used in mathematics,^[39] though others, such as APL, Smalltalk, Occam and Mary, have no operator precedence rules (in APL, evaluation is strictly right to left; in Smalltalk, it is strictly left to right).

Furthermore, because many operators are not associative, the order within any single level is usually defined by grouping left to right so that $16/4/4$ is interpreted as $(16/4)/4 = 1$ rather than $16/(4/4) = 16$; such operators are referred to as "left associative". Exceptions exist; for example, languages with operators corresponding to the cons operation on lists usually make them group right to left ("right associative"), e.g. in Haskell, $1:2:3:4:[] == 1:(2:(3:(4:[]))) == [1,2,3,4]$.

Dennis Ritchie, creator of the C language, said of the precedence in C (shared by programming languages that borrow those rules from C, for example, C++, Perl and PHP) that it would have been preferable to move the bitwise operators above the comparison operators.^[40] Many programmers have become accustomed to this order, but more recent popular languages like Python^[41] and Ruby^[42] do have this order reversed. The relative precedence levels of operators found in many C-style languages are as follows:

1	() [] -> . ::	Function call, scope, array/member access
2	! ~ - + * & <u>sizeof</u> <u>type cast</u> ++ - -	(most) unary operators, <u>sizeof</u> and <u>type casts</u> (right to left)
3	* / % MOD	Multiplication, division, <u>modulo</u>
4	+ -	Addition and subtraction
5	<< >>	Bitwise shift left and right
6	< <= > >=	Comparisons: less-than and greater-than
7	== !=	Comparisons: equal and not equal
8	&	Bitwise AND
9	^	Bitwise exclusive OR (XOR)
10		Bitwise inclusive (normal) OR
11	&&	Logical AND
12		Logical OR
13	? :	Conditional expression (ternary)
14	= += -= *= /= %= &= = ^= <<= >>=	Assignment operators (right to left)
15	,	<u>Comma operator</u>

Examples:

- !A + !B is interpreted as (!A) + (!B)
- ++A + !B is interpreted as (++A) + (!B)
- A + B * C is interpreted as A + (B * C)
- A || B && C is interpreted as A || (B && C)
- A && B == C is interpreted as A && (B == C)
- A & B == C is interpreted as A & (B == C)

(In Python, Ruby, PARI/GP and other popular languages, A & B == C is interpreted as (A & B) == C.)

Source-to-source compilers that compile to multiple languages need to explicitly deal with the issue of different order of operations across languages. Haxe for example standardizes the order and enforces it by inserting brackets where it is appropriate.

See also

- connective#Order of precedence

- ## Notes

Simplified formal grammar for arithmetical expressions in a programming language (*left*),^[43] and derivation of the example expression $(a+b)^2/2$ (*right*). The latter corresponds to a hierarchical structure ("syntax tree") which is unique for the given expression. The compiler generates machine code from the tree in such a way that operations originating at the lowest hierarchy level are executed first.

- ## References

1. "Calculation operators and precedence: Excel" (<https://support.microsoft.com/en-us/office/calculation-operators-and-precedence-36de9366-46fe-43a3-bfa8-cf6d8068eacc>). *Microsoft*

Support. Microsoft. 2023. Retrieved 2023-09-17.

2. Bronstein, Ilja Nikolaevič; Semendjajew, Konstantin Adolfovič (1987) [1945]. "2.4.1.1. Definition arithmetischer Ausdrücke" [Definition of arithmetic expressions]. In Grosche, Günter; Ziegler, Viktor; Ziegler, Dorothea (eds.). *Taschenbuch der Mathematik* [Pocketbook of mathematics] (in German). Vol. 1. Translated by Ziegler, Viktor (23rd ed.). Thun, Switzerland: Harri Deutsch. pp. 115–120, 802. ISBN 3-87144-492-8. "Regel 7: Ist $F(A)$ Teilzeichenreihe eines arithmetischen Ausdrucks oder einer seiner Abkürzungen und F eine Funktionenkonstante und A eine Zahlenvariable oder Zahlenkonstante, so darf $F A$ dafür geschrieben werden. [Darüber hinaus ist noch die Abkürzung $F^n(A)$ für $(F(A))^n$ üblich. Dabei kann F sowohl Funktionenkonstante als auch Funktionsvariable sein.]"
3. Peterson, Dave (September–October 2019). *The Math Doctors* (blog). Order of Operations: "Why?" (<https://www.themathdoctors.org/order-of-operations-why/>); "Why These Rules?" (<https://www.themathdoctors.org/order-of-operations-why-these-rules/>); "Subtle Distinctions" (<https://www.themathdoctors.org/order-of-operations-subtle-distinctions/>); "Fractions, Evaluating, and Simplifying" (<https://www.themathdoctors.org/order-of-operations-fractions-evaluating-and-simplifying/>); "Implicit Multiplication?" (<https://www.themathdoctors.org/order-of-operations-implicit-multiplication/>); "Historical Caveats" (<https://www.themathdoctors.org/order-of-operations-historical-caveats/>). Retrieved 2024-02-11.
Peterson, Dave (August–September 2023). *The Math Doctors* (blog). Implied Multiplication: "Not as Bad as You Think" (<https://www.themathdoctors.org/implied-multiplication-1-not-as-bad-as-you-think/>); "Is There a Standard?" (<https://www.themathdoctors.org/implied-multiplication-2-is-there-a-standard/>); "You Can't Prove It" (<https://www.themathdoctors.org/implied-multiplication-3-you-cant-prove-it/>). Retrieved 2024-02-11.
4. Swokowski, Earl William (1978). *Fundamentals of Algebra and Trigonometry* (<https://books.google.com/books?id=bZdfLRHFGw8C>) (4 ed.). Boston: Prindle, Weber & Schmidt. ISBN 0-87150-252-6. p. 1: "The *language of algebra* [...] may be used as shorthand, to abbreviate and simplify long or complicated statements."
5. Weisstein, Eric Wolfgang. "Precedence" (<https://mathworld.wolfram.com/Precedence.html>). *MathWorld*. Retrieved 2020-08-22.
6. Oldham, Keith B.; Myland, Jan C.; Spanier, Jerome (2009) [1987]. *An Atlas of Functions: with Equator, the Atlas Function Calculator* (2nd ed.). Springer. doi:10.1007/978-0-387-48807-3 (<https://doi.org/10.1007%2F978-0-387-48807-3>). ISBN 978-0-387-48806-6.
7. Olver, Frank W. J.; Lozier, Daniel W.; Boisvert, Ronald F.; Clark, Charles W., eds. (2010). *NIST Handbook of Mathematical Functions*. National Institute of Standards and Technology. ISBN 978-0-521-19225-5. MR 2723248 (<https://mathscinet.ams.org/mathscinet-getitem?mr=2723248>).
8. Angel, Allen R.; Runde, Dennis C.; Gilligan, Lawrence; Semmler, Richard (2010). *Elementary Algebra for College Students* (8th ed.). Prentice Hall. Ch. 1, §9, Objective 3. ISBN 978-0-321-62093-4.
9. "Formula Returns Unexpected Positive Value" (<https://web.archive.org/web/20150419091629/https://support.microsoft.com/en-gb/kb/kbview/132686>). Microsoft. 2005-08-15. Archived from the original (<https://support.microsoft.com/en-gb/kb/kbview/132686>) on 2015-04-19. Retrieved 2012-03-05.

10. Chrystal, George (1904) [1886]. *Algebra*. Vol. 1 (5th ed.). "Division", Ch. 1 §§19–26 (<https://archive.org/details/algebraelementar01chryuoft/page/14/>), pp. 14–20.

Chrystal's book was the canonical source in English about secondary school algebra of the turn of the 20th century, and plausibly the source for many later descriptions of the order of operations. However, while Chrystal's book initially establishes a rigid rule for evaluating expressions involving ' \div ' and ' \times ' symbols, it later consistently gives implicit multiplication higher precedence than division when writing inline fractions, without ever explicitly discussing the discrepancy between formal rule and common practice.

11. Cajori, Florian (1928). *A History of Mathematical Notations*. Vol. 1. La Salle, Illinois: Open Court. §242. "Order of operations in terms containing both \div and \times " (https://archive.org/details/s/b29980343_0001/page/274/), p. 274.
12. Wu, Hung-Hsi (2007) [2004]. "'Order of operations' and other oddities in school mathematics" (<https://math.berkeley.edu/~wu/order5.pdf>) (PDF). Dept. of Mathematics, University of California. Retrieved 2007-07-03.
13. In the [ISO 80000](#) standard, the division symbol ' \div ' is entirely disallowed in favor of a slash symbol:

ISO 80000-2:2019, "Quantities and units – Part 2: Mathematics" (<https://www.iso.org/obp/ui/#iso:std:iso:80000:-2:ed-2:v2:en>). International Standards Organization.

14. Lennes, N. J. (1917). "Discussions: Relating to the Order of Operations in Algebra". *The American Mathematical Monthly*. **24** (2): 93–95. doi:10.2307/2972726 (<https://doi.org/10.2307/2972726>). JSTOR 2972726 (<https://www.jstor.org/stable/2972726>).
15. Strogatz, Steven (2019-08-02). "The Math Equation That Tried to Stump the Internet" (<https://www.nytimes.com/2019/08/02/science/math-equation-pedmas-bemdas-bedmas.html>). *The New York Times*. Retrieved 2024-02-12. In this article, Strogatz describes the order of operations as taught in middle school. However, in a comment (<https://nyti.ms/32mhext#permid=101802829>), he points out,

"Several commenters appear to be using a different (and more sophisticated) convention than the elementary PEMDAS convention I described in the article. In this more sophisticated convention, which is often used in algebra, implicit multiplication (also known as multiplication by juxtaposition) is given higher priority than explicit multiplication or explicit division (in which one explicitly writes operators like \times * / or \div). Under this more sophisticated convention, the implicit multiplication in $2(2 + 2)$ is given higher priority than the explicit division implied by the use of \div . That's a very reasonable convention, and I agree that the answer is 1 if we are using this sophisticated convention.

"But that convention is not universal. For example, the calculators built into Google and WolframAlpha use the less sophisticated convention that I described in the article; they make no distinction between implicit and explicit multiplication when they are asked to evaluate simple arithmetic expressions. [...]"

16. "Physical Review Style and Notation Guide" (<https://publish.aps.org/files/styleguide-pr.pdf>) (PDF). American Physical Society. 2012. § IV.E.2.e. Retrieved 2012-08-05.
17. Graham, Ronald L.; Knuth, Donald E.; Patashnik, Oren (1994). *Concrete Mathematics* (2nd ed.). Reading, Mass: Addison-Wesley. "A Note on Notation", p. xi. ISBN 0-201-55802-5. MR 1397498 (<https://mathscinet.ams.org/mathscinet-getitem?mr=1397498>). "An expression of the form a/bc means the same as $a/(bc)$. Moreover, $\log x/\log y = (\log x)/(\log y)$ and $2n! = 2(n!)$."
18. Fateman, R. J.; Caspi, E. (1999). *Parsing TEX into mathematics* (https://people.eecs.berkeley.edu/~fateman/papers/parsing_tex.pdf) (PDF). International Symposium on Symbolic and Algebraic Computation, Vancouver, 28–31 July 1999.

19. Haelle, Tara (2013-03-12). "What Is the Answer to That Stupid Math Problem on Facebook? And why are people so riled up about it?" (<https://slate.com/technology/2013/03/facebook-math-problem-why-pemdas-doesnt-always-give-a-clear-answer.html>). *Slate*. Retrieved 2023-09-17.
20. "Rules of arithmetic" (<http://www.mathcentre.ac.uk/resources/uploaded/mc-ty-rules-2009-1.pdf>) (PDF). *Mathcentre.ac.uk*. 2009. Retrieved 2019-08-02.
21. Ginsburg, David (2011-01-01). "Please Excuse My Dear Aunt Sally (PEMDAS)--Forever!" (http://blogs.edweek.org/teachers/coach_gs_teaching_tips/2011/01/math_teachers_please_excuse_dear_aunt_sally--forever.html). *Education Week - Coach G's Teaching Tips*. Retrieved 2023-09-17.
22. Ali Rahman, Erna Sukinnah; Shahrill, Masitah; Abbas, Nor Arifahwati; Tan, Abby (2017). "Developing Students' Mathematical Skills Involving Order of Operations" (<https://files.eric.ed.gov/fulltext/EJ1148460.pdf>) (PDF). *International Journal of Research in Education and Science*. **3** (2): 373–382. doi:10.21890/ijres.327896 (<https://doi.org/10.21890%2Fijres.327896>). p. 373: "The PEMDAS is an acronym or mnemonic for the order of operations that stands for Parenthesis, Exponents, Multiplication, Division, Addition and Subtraction. This acronym is widely used in the United States of America. Meanwhile, in other countries such as United Kingdom and Canada, the acronyms used are BODMAS (Brackets, Order, Division, Multiplication, Addition and Subtraction) and BIDMAS (Brackets, Indices, Division, Multiplication, Addition and Subtraction)."
23. "Le calcul qui divise : $6 \div 2(1+2)$ " (<https://www.youtube.com/watch?v=tYf3CpbqAVo>). *Micmaths* (Video) (in French).
24. Vanderbeek, Greg (2007). *Order of Operations and RPN* (<https://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1045&context=mathmidexppap>) (Expository paper). Master of Arts in Teaching (MAT) Exam Expository Papers. Lincoln: University of Nebraska. Paper 46. Retrieved 2020-06-14.
25. Ball, John A. (1978). *Algorithms for RPN calculators* (<https://archive.org/details/algorithmsforrpn0000ball/page/31>) (1st ed.). Cambridge, Mass: Wiley. p. 31. ISBN 0-471-03070-8.
26. Naddor, Josh (2020). *Order of Operations: Please Excuse My Dear Aunt Sally as her rule is deceiving* (<https://esploro.libs.uga.edu/esploro/outputs/9949365543302959>) (MA thesis). University of Georgia.
27. "Order of operations" (http://syllabus.bos.nsw.edu.au/assets/global/files/maths_s3_sampleu1.doc) (DOC). *Syllabus.bos.nsw.edu.au*. Retrieved 2019-08-02.
28. Foster, Colin (2008). "Higher Priorities" (<https://www.researchgate.net/publication/261812399>). *Mathematics in School*. **37** (3): 17. JSTOR 30216129 (<https://www.jstor.org/stable/30216129>).
29. Ameis, Jerry A. (2011). "The Truth About PEDMAS". *Mathematics Teaching in the Middle School*. **16** (7): 414–420. doi:10.5951/MTMS.16.7.0414 (<https://doi.org/10.5951%2FMTMS.16.7.0414>). JSTOR 41183631 (<https://www.jstor.org/stable/41183631>).
30. Cheng, Eugenia (2023). *Is Math Real? How Simple Questions Lead Us to Mathematics' Deepest Truths*. Basic Books. pp. 235–238. ISBN 978-1-541-60182-6.
31. Lee, Jae Ki; Licwinko, Susan; Taylor-Buckner, Nicole (2013). "Exploring Mathematical Reasoning of the Order of Operations: Rearranging the Procedural Component PEMDAS" (<https://journals.library.columbia.edu/index.php/jmetc/article/download/633/79>). *Journal of Mathematics Education at Teachers College*. **4** (2): 73–78. doi:10.7916/jmetc.v4i2.633 (<https://doi.org/10.7916%2Fjmetc.v4i2.633>). p. 73: "[...] students frequently make calculation errors with expressions which have either multiplication and division or addition and subtraction next to each other. [...]"

32. Dupree, Kami M. (2016). "Questioning the Order of Operations". *Mathematics Teaching in the Middle School*. **22** (3): 152–159. doi:10.5951/mathteacmiddscho.22.3.0152 (<https://doi.org/10.5951%2Fmathteachmiddscho.22.3.0152>).
33. Taff, Jason (2017). "Rethinking the Order of Operations (or What Is the Matter with Dear Aunt Sally?)". *The Mathematics Teacher*. **111** (2): 126–132. doi:10.5951/mathteacher.111.2.0126 (<https://doi.org/10.5951%2Fmathteacher.111.2.0126>).
34. "Calculation Priority Sequence" (https://support.casio.com/global/en/calc/manual/fx-82MS_85MS_220PLUS_300MS_350MS_en/technical_informatoin/sequence.html). support.casio.com. Casio. Retrieved 2019-08-01.
35. "Implied Multiplication Versus Explicit Multiplication on TI Graphing Calculators" (http://epsstore.ti.com/OA_HTML/csksxvm.jsp?nSetId=103110). Texas Instruments. 2011. Retrieved 2015-08-24.
36. *Announcing the TI Programmable 88!* (http://www.datamath.net/Leaflets/TI-88_Announcemnt.pdf) (PDF). Texas Instruments. 1982. Retrieved 2017-08-03. "Now, implied multiplication is recognized by the AOS and the square root, logarithmic, and trigonometric functions can be followed by their arguments as when working with pencil and paper." (NB. The TI-88 only existed as a prototype and was never released to the public.)
37. Simons, Peter Murray (2021). "Łukasiewicz's Parenthesis-Free or Polish Notation" (<https://plato.stanford.edu/entries/lukasiewicz/polish-notation.html>). *Stanford Encyclopedia of Philosophy*. Dept. of Philosophy, Stanford University. Retrieved 2022-03-26.
38. Krtolica, Predrag V.; Stanimirović, Predrag S. (1999). "On some properties of reverse Polish Notation". *Filomat*. **13**: 157–172. JSTOR 43998756 (<https://www.jstor.org/stable/43998756>).
39. Henderson, Harry (2009) [2003]. "Operator Precedence" (<https://books.google.com/books?id=3Tla6d153uwC&pg=PA355>). *Henderson's Encyclopedia of Computer Science and Technology* (Rev. ed.). New York: Facts on File. p. 355. ISBN 978-0-8160-6382-6. Retrieved 2023-09-17.
40. Ritchie, Dennis M. (1996). "The Development of the C Language". *History of Programming Languages* (<https://www.bell-labs.com/usr/dmr/www/chist.html>) (2 ed.). ACM Press.
41. "6. Expressions" (<https://docs.python.org/3/reference/expressions.html>). *Python documentation*. Retrieved 2023-12-31.
42. "precedence - RDoc Documentation" (https://ruby-doc.org/3.3.0/syntax/precedence_rdoc.html). *ruby-doc.org*. Retrieved 2023-12-31.
43. Backus, John Warner; et al. (1963). "§ 3.3.1: Arithmetic expressions". In Naur, Peter (ed.). *Revised Report on the Algorithmic Language Algol 60* (<https://www.masswerk.at/algol60/report.htm>) (Report). Retrieved 2023-09-17. (CACM Vol. 6 pp. 1–17; The Computer Journal, Vol. 9, p. 349; Numerische Mathematik, Vol. 4, p. 420.)
44. Jones, Derek M. (2008) [2006]. "Developer beliefs about binary operator precedence" (<http://www.knosof.co.uk/cbook/accu06.html>). *CVu*. **18** (4): 14–21. Retrieved 2023-09-17.

Further reading

- Fothe, Michael; Wilke, Thomas, eds. (2015). *Keller, Stack und automatisches Gedächtnis – eine Struktur mit Potenzial* (<https://dl.gi.de/bitstream/handle/20.500.12116/4381/lni-t-7.pdf?sequence=1&isAllowed=y>) [*Cellar, stack and automatic memory – a structure with potential*]

External links

- Bergman, George Mark (2013). "Order of arithmetic operations; in particular, the $48/2(9+3)$ question" (https://math.berkeley.edu/~gbergman/misc/numbers/ord_ops.html). Dept. of Mathematics, University of California. Retrieved 2020-07-22.
 - Zachary, Joseph L. (1997) "Operator Precedence", supplement to *Introduction to Scientific Programming*. University of Utah. Maple worksheet (<http://www.cs.utah.edu/~zachary/isp/worksheets/operprec/operprec.html>), Mathematica notebook (<http://www.cs.utah.edu/~zachary/ispmma/notebooks/operprec/operprec.html>).
-

Retrieved from "https://en.wikipedia.org/w/index.php?title=Order_of_operations&oldid=1226349884"

■