

# Octal

The **octal** numeral system, or **oct** for short, is the base-8 number system, and uses the digits 0 to 7, that is to say 10 represents 8 in decimal and 100 represents 64 in decimal. However, English uses a base-10 number language system and so a true octal system might use different language to avoid confusion with the decimal system.

In the decimal system, each place is a power of ten. For example:

$$74_{10} = 7 \times 10^1 + 4 \times 10^0$$

In the octal system, each place is a power of eight. For example:

$$112_8 = 1 \times 8^2 + 1 \times 8^1 + 2 \times 8^0$$

By performing the calculation above in the familiar decimal system we see why 112 in octal is equal to 64+8+2 = 74 in decimal.

Octal numerals can be easily converted from binary representations (similar to a quaternary numeral system) by grouping consecutive binary digits into groups of three (starting from the right, for integers). For example, the binary representation for decimal 74 is 1001010. Two zeroes can be added at the left: (00)1 001 010, corresponding the octal digits 1 1 2, yielding the octal representation 112.

Numeral systems, bits and Gray code

	hex	dec	oct	3	2	1	0	step
0	0 <sub>hex</sub>	0 <sub>dec</sub>	0 <sub>oct</sub>	0	0	0	0	0
1	1 <sub>hex</sub>	1 <sub>dec</sub>	1 <sub>oct</sub>	0	0	0	1	1
2	2 <sub>hex</sub>	2 <sub>dec</sub>	2 <sub>oct</sub>	0	0	1	0	3
3	3 <sub>hex</sub>	3 <sub>dec</sub>	3 <sub>oct</sub>	0	0	1	1	2
4	4 <sub>hex</sub>	4 <sub>dec</sub>	4 <sub>oct</sub>	0	1	0	0	7
5	5 <sub>hex</sub>	5 <sub>dec</sub>	5 <sub>oct</sub>	0	1	0	1	6
6	6 <sub>hex</sub>	6 <sub>dec</sub>	6 <sub>oct</sub>	0	1	1	0	4
7	7 <sub>hex</sub>	7 <sub>dec</sub>	7 <sub>oct</sub>	0	1	1	1	5
8	8 <sub>hex</sub>	8 <sub>dec</sub>	10 <sub>oct</sub>	1	0	0	0	F
9	9 <sub>hex</sub>	9 <sub>dec</sub>	11 <sub>oct</sub>	1	0	0	1	E
A	10 <sub>hex</sub>	10 <sub>dec</sub>	12 <sub>oct</sub>	1	0	1	0	C
B	11 <sub>hex</sub>	11 <sub>dec</sub>	13 <sub>oct</sub>	1	0	1	1	D
C	12 <sub>hex</sub>	12 <sub>dec</sub>	14 <sub>oct</sub>	1	1	0	0	8
D	13 <sub>hex</sub>	13 <sub>dec</sub>	15 <sub>oct</sub>	1	1	0	1	9
E	14 <sub>hex</sub>	14 <sub>dec</sub>	16 <sub>oct</sub>	1	1	1	0	B
F	15 <sub>hex</sub>	15 <sub>dec</sub>	17 <sub>oct</sub>	1	1	1	1	A

## Contents

### Usage

- [By Native Americans](#)
- [By Europeans](#)
- [In computers](#)
- [In aviation](#)

### Conversion between bases

- [Decimal to octal conversion](#)
  - [Method of successive Euclidean division by 8](#)
  - [Method of successive multiplication by 8](#)
  - [Method of successive duplication](#)
- [Octal to decimal conversion](#)
  - [Method of successive duplication](#)
- [Octal to binary conversion](#)
- [Binary to octal conversion](#)
- [Octal to hexadecimal conversion](#)
- [Hexadecimal to octal conversion](#)

### Real numbers

- [Fractions](#)
- [Irrational numbers](#)

### See also

The octal multiplication table

×	1	2	3	4	5	6	7	10
1	1	2	3	4	5	6	7	10
2	2	4	6	10	12	14	16	20
3	3	6	11	14	17	22	25	30
4	4	10	14	20	24	30	34	40
5	5	12	17	24	31	36	43	50
6	6	14	22	30	36	44	52	60
7	7	16	25	34	43	52	61	70
10	10	20	30	40	50	60	70	100

## Usage

---

### By Native Americans

- The Yuki language in California has an octal system because the speakers count using the spaces between their fingers rather than the fingers themselves.<sup>[1]</sup>
- The Pamean languages in Mexico also have an octal system, because their speakers count on the knuckles of a closed fist.<sup>[2]</sup>

### By Europeans

- It has been suggested that the reconstructed Proto-Indo-European (PIE) word for "nine" might be related to the PIE word for "new". Based on this, some have speculated that proto-Indo-Europeans used an octal number system, though the evidence supporting this is slim.<sup>[3]</sup>
- In 1668, John Wilkins in *An Essay towards a Real Character, and a Philosophical Language* proposed use of base 8 instead of 10 "because the way of Dichotomy or Bipartition being the most natural and easie kind of Division, that Number is capable of this down to an Unite".<sup>[4]</sup>
- In 1716, King Charles XII of Sweden asked Emanuel Swedenborg to elaborate a number system based on 64 instead of 10. Swedenborg however argued that for people with less intelligence than the king such a big base would be too difficult and instead proposed 8 as the base. In 1718 Swedenborg wrote (but did not publish) a manuscript: "En ny rekenkonst som om vexlas wid Thetalet 8 i stelle then wanliga wid Thetalet 10" ("A new arithmetic (or art of counting) which changes at the Number 8 instead of the usual at the Number 10"). The numbers 1-7 are there denoted by the consonants l, s, n, m, t, f, u (v) and zero by the vowel o. Thus 8 = "lo", 16 = "so", 24 = "no", 64 = "loo", 512 = "looo" etc. Numbers with consecutive consonants are pronounced with vowel sounds between in accordance with a special rule.<sup>[5]</sup>
- Writing under the pseudonym "Hirossa Ap-Iccim" in *The Gentleman's Magazine*, (London) July 1745, Hugh Jones proposed an octal system for British coins, weights and measures. "Whereas reason and convenience indicate to us an uniform standard for all quantities; which I shall call the *Georgian standard*; and that is only to divide every integer in each *species* into eight equal parts, and every part again into 8 real or imaginary particles, as far as is necessary. For tho' all nations count universally by *tens* (originally occasioned by the number of digits on both hands) yet 8 is a far more complete and commodious number; since it is divisible into halves, quarters, and half quarters (or units) without a fraction, of which subdivision *ten* is incapable...." In a later treatise on Octave computation (1753) Jones concluded: "Arithmetic by *Octaves* seems most agreeable to the Nature of Things, and therefore may be called Natural Arithmetic in Opposition to that now in Use, by *Decades*; which may be esteemed Artificial Arithmetic."<sup>[6]</sup>
- In 1801, James Anderson criticized the French for basing the metric system on decimal arithmetic. He suggested base 8, for which he coined the term *octal*. His work was intended as recreational mathematics, but he suggested a purely octal system of weights and measures and observed that the existing system of English units was already, to a remarkable extent, an octal system.<sup>[7]</sup>
- In the mid-19th century, Alfred B. Taylor concluded that "Our octonary [base 8] radix is, therefore, beyond all comparison the "*best possible one*" for an arithmetical system." The proposal included a graphical notation for the digits and new names for the numbers, suggesting that we should count "*un, du, the, fo, pa, se, ki, unty, unty-un, unty-du*" and so on, with successive multiples of eight named "*unty, duty, thety, foty, paty, sety, kity* and *under*." So, for example, the number 65 (101 in octal) would be spoken in octonary as *under-un*.<sup>[8][9]</sup> Taylor also republished some of Swedenborg's work on octal as an appendix to the above-cited publications.

## In computers

Octal became widely used in computing when systems such as the UNIVAC 1050, PDP-8, ICL 1900 and IBM mainframes employed 6-bit, 12-bit, 24-bit or 36-bit words. Octal was an ideal abbreviation of binary for these machines because their word size is divisible by three (each octal digit represents three binary digits). So two, four, eight or twelve digits could concisely display an entire machine word. It also cut costs by allowing Nixie tubes, seven-segment displays, and calculators to be used for the operator consoles, where binary displays were too complex to use, decimal displays needed complex hardware to convert radices, and hexadecimal displays needed to display more numerals.

All modern computing platforms, however, use 16-, 32-, or 64-bit words, further divided into eight-bit bytes. On such systems three octal digits per byte would be required, with the most significant octal digit representing two binary digits (plus one bit of the next significant byte, if any). Octal representation of a 16-bit word requires 6 digits, but the most significant octal digit represents (quite inelegantly) only one bit (0 or 1). This representation offers no way to easily read the most significant byte, because it's smeared over four octal digits. Therefore, hexadecimal is more commonly used in programming languages today, since two hexadecimal digits exactly specify one byte. Some platforms with a power-of-two word size still have instruction subwords that are more easily understood if displayed in octal; this includes the PDP-11 and Motorola 68000 family. The modern-day ubiquitous x86 architecture belongs to this category as well, but octal is rarely used on this platform, although certain properties of the binary encoding of opcodes become more readily apparent when displayed in octal, e.g. the ModRM byte, which is divided into fields of 2, 3, and 3 bits, so octal can be useful in describing these encodings. Before the availability of assemblers, some programmers would handcode programs in octal; for instance, Dick Whipple and John Arnold wrote Tiny BASIC Extended directly in machine code, using octal.<sup>[10]</sup>

Octal is sometimes used in computing instead of hexadecimal, perhaps most often in modern times in conjunction with file permissions under Unix systems (see chmod). It has the advantage of not requiring any extra symbols as digits (the hexadecimal system is base-16 and therefore needs six additional symbols beyond 0–9). It is also used for digital displays.

In programming languages, octal literals are typically identified with a variety of prefixes, including the digit 0, the letters o or q, the digit–letter combination 0o, or the symbol &<sup>[11]</sup> or \$. In *Motorola convention*, octal numbers are prefixed with @, whereas a small (or capital<sup>[12]</sup>) letter o<sup>[12]</sup> or q<sup>[12]</sup> is added as a postfix following the *Intel convention*.<sup>[13][14]</sup> In Concurrent DOS, Multiuser DOS and REAL/32 as well as in DOS Plus and DR-DOS various environment variables like \$CLS, \$ON, \$OFF, \$HEADER or \$FOOTER support an \nnn octal number notation,<sup>[15][16][17]</sup> and DR-DOS DEBUG utilizes \ to prefix octal numbers as well.

For example, the literal 73 (base 8) might be represented as 073, o73, q73, 0o73, \73, @73, &73, \$73 or 73o in various languages.

Newer languages have been abandoning the prefix 0, as decimal numbers are often represented with leading zeroes. The prefix q was introduced to avoid the prefix o being mistaken for a zero, while the prefix 0o was introduced to avoid starting a numerical literal with an alphabetic character (like o or q), since these might cause the literal to be confused with a variable name. The prefix 0o also follows the model set by the prefix 0x used for hexadecimal literals in the C language; it is supported by Haskell,<sup>[18]</sup> OCaml,<sup>[19]</sup> Python as of version 3.0,<sup>[20]</sup> Raku,<sup>[21]</sup> Ruby,<sup>[22]</sup> Tcl as of version 9,<sup>[23]</sup> PHP as of version 8.1<sup>[24]</sup> and it is intended to be supported by ECMAScript 6<sup>[25]</sup> (the prefix 0 originally stood for base 8 in JavaScript but could cause confusion,<sup>[26]</sup> therefore it has been discouraged in ECMAScript 3 and dropped in ECMAScript 5<sup>[27]</sup>).

Octal numbers that are used in some programming languages (C, Perl, PostScript...) for textual/graphical representations of byte strings when some byte values (unrepresented in a code page, non-graphical, having special meaning in current context or otherwise undesired) have to be escaped as \nnn. Octal representation may be particularly handy with non-ASCII bytes of UTF-8, which encodes groups of 6 bits, and where any start byte has octal value \3nn and any continuation byte has octal value \2nn.

Octal was also used for floating point in the Ferranti Atlas (1962), Burroughs B5500 (1964), Burroughs B5700 (1971), Burroughs B6700 (1971) and Burroughs B7700 (1972) computers.

## In aviation

Transponders in aircraft transmit a code, expressed as a four-octal-digit number, when interrogated by ground radar. This code is used to distinguish different aircraft on the radar screen.

## Conversion between bases

---

### Decimal to octal conversion

#### Method of successive Euclidean division by 8

To convert integer decimals to octal, divide the original number by the largest possible power of 8 and divide the remainders by successively smaller powers of 8 until the power is 1. The octal representation is formed by the quotients, written in the order generated by the algorithm. For example, to convert  $125_{10}$  to octal:

$$\begin{aligned}125 &= 8^2 \times 1 + 61 \\61 &= 8^1 \times 7 + 5 \\5 &= 8^0 \times 5 + 0\end{aligned}$$

Therefore,  $125_{10} = 175_8$ .

Another example:

$$\begin{aligned}900 &= 8^3 \times 1 + 388 \\388 &= 8^2 \times 6 + 4 \\4 &= 8^1 \times 0 + 4 \\4 &= 8^0 \times 4 + 0\end{aligned}$$

Therefore,  $900_{10} = 1604_8$ .

#### Method of successive multiplication by 8

To convert a decimal fraction to octal, multiply by 8; the integer part of the result is the first digit of the octal fraction. Repeat the process with the fractional part of the result, until it is null or within acceptable error bounds.

Example: Convert  $0.1640625$  to octal:

$$\begin{aligned}0.1640625 \times 8 &= 1.3125 = \mathbf{1} + 0.3125 \\0.3125 \times 8 &= 2.5 = \mathbf{2} + 0.5 \\0.5 \times 8 &= 4.0 = \mathbf{4} + 0\end{aligned}$$

Therefore,  $0.1640625_{10} = 0.124_8$ .

These two methods can be combined to handle decimal numbers with both integer and fractional parts, using the first on the integer part and the second on the fractional part.

#### Method of successive duplication

To convert integer decimals to octal, prefix the number with "0.". Perform the following steps for as long as digits remain on the right side of the radix: Double the value to the left side of the radix, using *octal* rules, move the radix point one digit rightward, and then place the doubled value underneath the current value so that the radix points align. If the moved radix point crosses over a digit that is 8 or 9, convert it to 0 or 1 and add the carry to the next leftward digit of the current value. *Add octally* those digits to the left of the radix and simply drop down those digits to the right, without modification.

Example:

```

0.4 9 1 8 decimal value
+0
-----
 4.9 1 8
+1 0
-----
 6 1.1 8
+1 4 2
-----
 7 5 3.8
+1 7 2 6
-----
1 1 4 6 6. octal value

```

### Octal to decimal conversion

To convert a number  $k$  to decimal, use the formula that defines its base-8 representation:

$$k = \sum_{i=0}^n (a_i \times 8^i)$$

In this formula,  $a_i$  is an individual octal digit being converted, where  $i$  is the position of the digit (counting from 0 for the right-most digit).

Example: Convert  $764_8$  to decimal:

$$764_8 = 7 \times 8^2 + 6 \times 8^1 + 4 \times 8^0 = 448 + 48 + 4 = 500_{10}$$

For double-digit octal numbers this method amounts to multiplying the lead digit by 8 and adding the second digit to get the total.

Example:  $65_8 = 6 \times 8 + 5 = 53_{10}$

### Method of successive duplication

To convert octals to decimals, prefix the number with "0.". Perform the following steps for as long as digits remain on the right side of the radix: Double the value to the left side of the radix, using *decimal* rules, move the radix point one digit rightward, and then place the doubled value underneath the current value so that the radix points align. *Subtract decimally* those digits to the left of the radix and simply drop down those digits to the right, without modification.

Example:

```

0.1 1 4 6 6 octal value
-0
-----
 1.1 4 6 6
- 2
-----
 9.4 6 6
- 1 8
-----

```

$$\begin{array}{r}
 76.66 \\
 - 152 \\
 \hline
 614.6 \\
 - 1228 \\
 \hline
 4918. \text{ decimal value}
 \end{array}$$

## Octal to binary conversion

To convert octal to binary, replace each octal digit by its binary representation.

Example: Convert  $51_8$  to binary:

$$\begin{aligned}
 5_8 &= 101_2 \\
 1_8 &= 001_2
 \end{aligned}$$

Therefore,  $51_8 = 101001_2$ .

## Binary to octal conversion

The process is the reverse of the previous algorithm. The binary digits are grouped by threes, starting from the least significant bit and proceeding to the left and to the right. Add leading zeroes (or trailing zeroes to the right of decimal point) to fill out the last group of three if necessary. Then replace each trio with the equivalent octal digit.

For instance, convert binary  $1010111100$  to octal:

001	010	111	100
1	2	7	4

Therefore,  $1010111100_2 = 1274_8$ .

Convert binary  $11100.01001$  to octal:

011	100	.	010	010
3	4	.	2	2

Therefore,  $11100.01001_2 = 34.22_8$ .

## Octal to hexadecimal conversion

The conversion is made in two steps using binary as an intermediate base. Octal is converted to binary and then binary to hexadecimal, grouping digits by fours, which correspond each to a hexadecimal digit.

For instance, convert octal  $1057$  to hexadecimal:

To binary:

1	0	5	7
001	000	101	111

then to hexadecimal:

0010	0010	1111
------	------	------

2	2	F
---	---	---

Therefore,  $1057_8 = 22F_{16}$ .

## Hexadecimal to octal conversion

Hexadecimal to octal conversion proceeds by first converting the hexadecimal digits to 4-bit binary values, then regrouping the binary bits into 3-bit octal digits.

For example, to convert  $3FA5_{16}$ :

To binary:

3	F	A	5
0011	1111	1010	0101

then to octal:

0	011	111	110	100	101
0	3	7	6	4	5

Therefore,  $3FA5_{16} = 37645_8$ .

## Real numbers

---

### Fractions

Due to having only factors of two, many octal fractions have repeating digits, although these tend to be fairly simple:

Decimal base Prime factors of the base: <b>2, 5</b> Prime factors of one below the base: <b>3</b> Prime factors of one above the base: <b>11</b> Other Prime factors: <b>7 13 17 19 23 29 31</b>			Octal base Prime factors of the base: <b>2</b> Prime factors of one below the base: <b>7</b> Prime factors of one above the base: <b>3</b> Other Prime factors: <b>5 13 15 21 23 27 35 37</b>		
Fraction	Prime factors of the denominator	Positional representation	Positional representation	Prime factors of the denominator	Fraction
1/2	<b>2</b>	<b>0.5</b>	<b>0.4</b>	<b>2</b>	1/2
1/3	<b>3</b>	<b>0.3333... = 0.<math>\overline{3}</math></b>	<b>0.2525... = 0.<math>\overline{25}</math></b>	<b>3</b>	1/3
1/4	<b>2</b>	<b>0.25</b>	<b>0.2</b>	<b>2</b>	1/4
1/5	<b>5</b>	<b>0.2</b>	<b>0.146<math>\overline{3}</math></b>	<b>5</b>	1/5
1/6	<b>2, 3</b>	<b>0.1<math>\overline{6}</math></b>	<b>0.1<math>\overline{25}</math></b>	<b>2, 3</b>	1/6
1/7	<b>7</b>	<b>0.14285<math>\overline{7}</math></b>	<b>0.<math>\overline{1}</math></b>	<b>7</b>	1/7
1/8	<b>2</b>	<b>0.125</b>	<b>0.1</b>	<b>2</b>	1/10
1/9	<b>3</b>	<b>0.1<math>\overline{1}</math></b>	<b>0.0<math>\overline{7}</math></b>	<b>3</b>	1/11
1/10	<b>2, 5</b>	<b>0.1</b>	<b>0.06<math>\overline{314}</math></b>	<b>2, 5</b>	1/12
1/11	<b>11</b>	<b>0.0<math>\overline{9}</math></b>	<b>0.056427213<math>\overline{5}</math></b>	<b>13</b>	1/13
1/12	<b>2, 3</b>	<b>0.08<math>\overline{3}</math></b>	<b>0.05<math>\overline{2}</math></b>	<b>2, 3</b>	1/14
1/13	<b>13</b>	<b>0.07692<math>\overline{3}</math></b>	<b>0.047<math>\overline{3}</math></b>	<b>15</b>	1/15
1/14	<b>2, 7</b>	<b>0.0714285<math>\overline{14}</math></b>	<b>0.04<math>\overline{1}</math></b>	<b>2, 7</b>	1/16
1/15	<b>3, 5</b>	<b>0.06<math>\overline{6}</math></b>	<b>0.042<math>\overline{1}</math></b>	<b>3, 5</b>	1/17
1/16	<b>2</b>	<b>0.0625</b>	<b>0.04</b>	<b>2</b>	1/20
1/17	<b>17</b>	<b>0.058823529411764<math>\overline{7}</math></b>	<b>0.0360741<math>\overline{7}</math></b>	<b>21</b>	1/21
1/18	<b>2, 3</b>	<b>0.05<math>\overline{5}</math></b>	<b>0.034<math>\overline{1}</math></b>	<b>2, 3</b>	1/22
1/19	<b>19</b>	<b>0.05263157894736842<math>\overline{1}</math></b>	<b>0.03274<math>\overline{5}</math></b>	<b>23</b>	1/23
1/20	<b>2, 5</b>	<b>0.05</b>	<b>0.0314<math>\overline{6}</math></b>	<b>2, 5</b>	1/24
1/21	<b>3, 7</b>	<b>0.047619<math>\overline{21}</math></b>	<b>0.03<math>\overline{1}</math></b>	<b>3, 7</b>	1/25
1/22	<b>2, 11</b>	<b>0.04<math>\overline{5}</math></b>	<b>0.02721350564<math>\overline{11}</math></b>	<b>2, 13</b>	1/26
1/23	<b>23</b>	<b>0.043478260869565217391<math>\overline{3}</math></b>	<b>0.02620544131<math>\overline{11}</math></b>	<b>27</b>	1/27
1/24	<b>2, 3</b>	<b>0.041<math>\overline{6}</math></b>	<b>0.02<math>\overline{5}</math></b>	<b>2, 3</b>	1/30
1/25	<b>5</b>	<b>0.04</b>	<b>0.0243656050753412172<math>\overline{7}</math></b>	<b>5</b>	1/31
1/26	<b>2, 13</b>	<b>0.038461<math>\overline{5}</math></b>	<b>0.02354<math>\overline{1}</math></b>	<b>2, 15</b>	1/32
1/27	<b>3</b>	<b>0.03<math>\overline{7}</math></b>	<b>0.02275<math>\overline{5}</math></b>	<b>3</b>	1/33
1/28	<b>2, 7</b>	<b>0.03571428<math>\overline{56}</math></b>	<b>0.02<math>\overline{2}</math></b>	<b>2, 7</b>	1/34
1/29	<b>29</b>	<b>0.0344827586206896551724137931<math>\overline{29}</math></b>	<b>0.0215173454106475626043236713<math>\overline{29}</math></b>	<b>35</b>	1/35
1/30	<b>2, 3, 5</b>	<b>0.03<math>\overline{3}</math></b>	<b>0.02104<math>\overline{1}</math></b>	<b>2, 3, 5</b>	1/36
1/31	<b>31</b>	<b>0.032258064516129<math>\overline{31}</math></b>	<b>0.02041<math>\overline{1}</math></b>	<b>37</b>	1/37
1/32	<b>2</b>	<b>0.03125</b>	<b>0.02</b>	<b>2</b>	1/40

## Irrational numbers

The table below gives the expansions of some common irrational numbers in decimal and octal.



Number	Positional representation	
	Decimal	Octal
$\sqrt{2}$ (the length of the diagonal of a unit square)	1.414 213 562 373 095 048...	1.3240 4746 3177 1674...
$\sqrt{3}$ (the length of the diagonal of a unit cube)	1.732 050 807 568 877 293...	1.5666 3656 4130 2312...
$\sqrt{5}$ (the length of the diagonal of a 1×2 rectangle)	2.236 067 977 499 789 696...	2.1706 7363 3457 7224...
$\varphi$ (phi, the golden ratio = $(1+\sqrt{5})/2$ )	1.618 033 988 749 894 848...	1.4743 3571 5627 7512...
$\pi$ (pi, the ratio of circumference to diameter of a circle)	3.141 592 653 589 793 238 462 643 383 279 502 884 197 169 399 375 105...	3.1103 7552 4210 2643...
$e$ (the base of the natural logarithm)	2.718 281 828 459 045 235...	2.5576 0521 3050 5355...

## See also

- Computer numbering formats
- Octal games, a game numbering system used in combinatorial game theory
- Split octal, a 16-bit octal notation used by the Heath Company, DEC and others
- Squawk code, a 12-bit octal representation of Gillham code
- Syllabic octal, an octal representation of 8-bit syllables used by English Electric

## References

1. Ascher, Marcia (1992). "Ethnomathematics: A Multicultural View of Mathematical Ideas". *The College Mathematics Journal*. **23** (4): 353–355. doi:10.2307/2686959 (https://doi.org/10.2307%2F2686959). JSTOR 2686959 (https://www.jstor.org/stable/2686959).
2. Avelino, Heriberto (2006). "The typology of Pame number systems and the limits of Mesoamerica as a linguistic area" (http://linguistics.berkeley.edu/~avelino/Avelino\_2006.pdf) (PDF). *Linguistic Typology*. **10** (1): 41–60. doi:10.1515/LINGTY.2006.002 (https://doi.org/10.1515%2FLINGTY.2006.002).
3. Winter, Werner (1991). "Some thoughts about Indo-European numerals" (https://books.google.com/books?id=S-hmNOLuDGsC&lpg=PA170&pg=PA13). In Gvozdanović, Jadranka (ed.). *Indo-European numerals*. Trends in Linguistics. **57**. Berlin: Mouton de Gruyter. pp. 13–14. ISBN 3-11-011322-8. Retrieved 2013-06-09.
4. Wilkins, John (1668). *An Essay Towards a Real Character and a Philosophical Language* (https://books.google.com/?id=BCctZjBtiEYC&pg=PA190). London. p. 190. Retrieved 2015-02-08.
5. Donald Knuth, *The Art of Computer Programming*
6. See H. R. Phalen, "Hugh Jones and Octave Computation," *The American Mathematical Monthly* 56 (August–September 1949): 461-465.
7. James Anderson, On Octal Arithmetic [title appears only in page headers], Recreations in Agriculture, Natural-History, Arts, and Miscellaneous Literature (https://books.google.com/books?id=olhHAAAAYAAJ&pg=PA437), Vol. IV, No. 6 (February 1801), T. Bensley, London; pages 437-448.
8. Alfred B. Taylor, Report on Weights and Measures (https://archive.org/details/reportonweights00taylgoog), Pharmaceutical Association, 8th Annual Session, Boston, 1859-09-15. See pages 48 and 53.
9. Alfred B. Taylor, Octonary numeration and its application to a system of weights and measures, *Proc. Amer. Phil. Soc.* Vol XXIV (https://books.google.com/books?id=KsAUAAAAYAAJ&pg=PA296), Philadelphia, 1887; pages 296-366. See pages 327 and 330.
10. "TB Code Sheet". *Dr. Dobb's Journal of Computer Calisthenics & Orthodontia, Running Light Without Overbyte*. **1** (1). December 1975.
11. Microsoft Corporation (1987). "Constants, Variables, Expressions and Operators" (http://www.antonis.de/qbebooks/gwasman/chapter%206.html). *GW-BASIC User's Manual* (http://www.antonis.de/qbebooks/gwasman/). Retrieved 2015-12-12.

12. "2.4.1 Numeric Constants". *CP/M-86 - Operating System - Programmer's Guide* ([http://www.bitsavers.org/pdf/digitalResearch/cpm-86/CPM-86\\_Programmers\\_Guide\\_Jan83.pdf](http://www.bitsavers.org/pdf/digitalResearch/cpm-86/CPM-86_Programmers_Guide_Jan83.pdf)) (PDF) (3 ed.). Pacific Grove, California, USA: Digital Research. January 1983 [1981]. p. 9. Archived ([https://web.archive.org/web/20200227225328/http://www.bitsavers.org/pdf/digitalResearch/cpm-86/CPM-86\\_Programmers\\_Guide\\_Jan83.pdf](https://web.archive.org/web/20200227225328/http://www.bitsavers.org/pdf/digitalResearch/cpm-86/CPM-86_Programmers_Guide_Jan83.pdf)) (PDF) from the original on 2020-02-27. Retrieved 2020-02-27. [1] (<https://web.archive.org/web/20200227132338/https://www.z80cpu.eu/mirrors/oldcomputers.dyndns.org/public/pub/manuals/cpm86pg.pdf>) (1+viii+122+2 pages)
13. Küveler, Gerd; Schwoch, Dietrich (2013) [1996]. *Arbeitsbuch Informatik - eine praxisorientierte Einführung in die Datenverarbeitung mit Projektaufgabe* (<https://books.google.com/books?id=b8-dBgAAQBAJ>) (in German). Vieweg-Verlag, reprint: Springer-Verlag. doi:10.1007/978-3-322-92907-5 (<https://doi.org/10.1007%2F978-3-322-92907-5>). ISBN 978-3-528-04952-2. 978-3-32292907-5. Retrieved 2015-08-05.
14. Küveler, Gerd; Schwoch, Dietrich (2007-10-04). *Informatik für Ingenieure und Naturwissenschaftler: PC- und Mikrocomputertechnik, Rechnernetze* (<https://books.google.com/books?id=xQbvPYxceY0C>) (in German). 2 (5 ed.). Vieweg, reprint: Springer-Verlag. ISBN 978-3-83489191-4. 978-3-83489191-4. Retrieved 2015-08-05.
15. Paul, Matthias R. (1997-07-30). *NWDOS-TIPs — Tips & Tricks rund um Novell DOS 7, mit Blick auf undokumentierte Details, Bugs und Workarounds* (<http://www.antonis.de/dos/dos-tuts/mpdostip/html/nwdostip.htm>). *MPDOSTIP*. Release 157 (in German) (3 ed.). Archived (<https://web.archive.org/web/20161104235829/http://www.antonis.de/dos/dos-tuts/mpdostip/html/nwdostip.htm>) from the original on 2016-11-04. Retrieved 2014-08-06. (NB. NWDOSTIP.TXT is a comprehensive work on Novell DOS 7 and OpenDOS 7.01, including the description of many undocumented features and internals. It is part of the author's yet larger MPDOSTIP . ZIP collection maintained up to 2001 and distributed on many sites at the time. The provided link points to a HTML-converted older version of the NWDOSTIP . TXT file.)
16. Paul, Matthias R. (2002-03-26). "Updated CLS posted" (<https://marc.info/?l=freedos-dev&m=101717593306186&w=2>). freedos-dev mailing list. Archived (<https://archive.today/20190427173821/http://marc.info/?l=freedos-dev&m=101717593306186&w=2>) from the original on 2019-04-27. Retrieved 2014-08-06.
17. *CCI Multiuser DOS 7.22 GOLD Online Documentation*. Concurrent Controls, Inc. (CCI). 1997-02-10. HELP.HLP.
18. Haskell 98 Lexical Structure (<https://www.haskell.org/onlinereport/lexemes.html#sect2.5>)
19. OCaml: 7.1 Lexical conventions (<http://caml.inria.fr/pub/docs/manual-ocaml/lex.html>)
20. Python 3: [https://docs.python.org/3.1/reference/lexical\\_analysis.html#integer-literals](https://docs.python.org/3.1/reference/lexical_analysis.html#integer-literals)
21. Perl 6: [http://perlcabal.org/syn/S02.html#Radix\\_markers](http://perlcabal.org/syn/S02.html#Radix_markers) Archived ([https://web.archive.org/web/20141031161348/http://perlcabal.org/syn/S02.html#Radix\\_markers](https://web.archive.org/web/20141031161348/http://perlcabal.org/syn/S02.html#Radix_markers)) 31 October 2014 at the Wayback Machine
22. RubySpec: [https://github.com/kostya/rubyspec/blob/master/core/string/to\\_i\\_spec.rb](https://github.com/kostya/rubyspec/blob/master/core/string/to_i_spec.rb)
23. Tcl: <http://wiki.tcl.tk/498>
24. PHP.Watch - PHP 8.1: Explicit Octal numeral notation <https://php.watch/versions/8.1/explicit-octal-notation>
25. ECMAScript 6th Edition draft: <https://people.mozilla.org/~jorendorff/es6-draft.html#sec-literals-numeric-literals> Archived (<https://web.archive.org/web/20131216202526/https://people.mozilla.org/~jorendorff/es6-draft.html#sec-literals-numeric-literals>) 16 December 2013 at the Wayback Machine
26. "Why does the radix for JavaScript's parseInt default to 8?" (<https://stackoverflow.com/questions/5600366/why-does-the-radix-for-javascripts-parseint-default-to-8>). *Stack Overflow*. 2011-04-08.
27. "parseInt()" ([https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/parseInt](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/parseInt)), *Mozilla Developer Network (MDN)*, "If the input string begins with "0" (a zero), radix is assumed to be 8 (octal) or 10 (decimal). Exactly which radix is chosen is implementation-dependent. ECMAScript 5 clarifies that 10 (decimal) should be used, but not all browsers support this yet"

## External links

---

- [Octomatics \(http://www.octomatics.org\)](http://www.octomatics.org) is a numeral system enabling simple visual calculation in octal.
  - [Octal converter \(https://octal2decimal.converter.page\)](https://octal2decimal.converter.page) performs bidirectional conversions between the octal and decimal system.
- 

Retrieved from "<https://en.wikipedia.org/w/index.php?title=Octal&oldid=1045245006>"

---

**This page was last edited on 19 September 2021, at 16:07 (UTC).**

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.