Ratios

Ratios (a type of number that represents fractions)

- Unlike C++ and Java, Common Lisp has a built-in data type called ***ratio*** that represents fractions ***exactly***, with no rounding error.
- Ratios are always ***printed in lowest terms*** (but they need not be written in lowest terms): You can write 6/8, but this ratio would be printed as 3/4.
- If m and n are integers, n is not 0, and n does not divide m, then the value of (/ m n) is a ratio. For example, the value of (/ 12 9) is 12/9 = 4/3.
- There is no space before or after the / in a ratio: 5/7 ***cannot*** be written as 5 /7 or 5/ 7.
- In Common Lisp, a number is said to be ***rational*** if it is *either* an integer *or* a ratio.
- The functions +, –, *, and / accept rational and floating point arguments: If each argument is rational, so is the result; if any argument is a floating point number and all arguments are real, the result is a floating point number.

  Note: This is true for Clisp on **mars** if Clisp is started using the **cl** command. But if you use Clisp with default settings on a PC or Mac, or start Clisp using the **clisp** command on **mars**, then Clisp will return the integer 0 (and not the floating point number 0.0) as the result of multiplying 0 by a floating point number or dividing 0 by a nonzero floating point number.