

ACSL
American Computer Science League
2015 - 2016 **Contest #2**
ACSL STRING
SENIOR DIVISION

PROBLEM. For this program you will be asked to code algorithms that replicates the ACSL string function STR defined below:

STR (float_expression, length, decimal) – Converts a given floating point number to a formatted string value.

Length - The desired total length of the resulting string which includes all digits, decimals and signs.

Decimal - The number of decimal places to display in the resulting string. The result will be rounded to the given number of decimal places if necessary.

A length value that does not allow a value to be printed in the given format would cause an error. The # symbol will be printed to show the desired format. Example: STR (523.125, 5, 2) would produce ###.##

A length value greater than the length of the float expression causes the correct result to be right justified with the # symbol printed to the left to fill the desired length. Example: STR(523.125, 9, 2) would produce #####523.13 .

For non-negative numbers, rounding up occurs when the digit to the right is 5 or more. Rounding down occurs when the digit to the right is less than point 5. If the value is negative the rule that applies is -0.75 rounds down to -0.8 and -0.74 rounds up to -0.7.

INPUT: There will be 5 lines of input. Each line of input will contain a rational number (with or without a sign, and 2 non-negative integers. The 3 values represent the arguments of the STR command.

OUTPUT: For each line of input print the result of applying the STR command as described above.

SAMPLE INPUT

1. 523.125, 6, 2
2. +523.125, 6, 1
3. -523.163, 6, 1
4. 523.125, 4, 2
5. -523.12, 6, 1

SAMPLE OUTPUT

1. 523.13
2. +523.1
3. -523.2
4. #.##
5. -523.1