

**2013 - 2014**

**ACSL**  
**American Computer Science League**

---

**PRINT FORMATTING**  
 Senior Division

**Contest #2**

PROBLEM: In the ACSL programming language formatting output is accomplished using the ACSL FORMAT command.

The syntax of the command is : FORMAT string, value

The string part of the command consists of a series of &'s that may or may not have imbedded characters. The table below defines the various combinations that will be used for this program.

String	Value	Result
&&&&	25	The 4 &'s define a field of 4 characters. The value will be right justified in the field. *'s fill on the left. Output **25
&,&&&&&	12345	A comma anywhere in the string inserts a comma after every third character with a character following. Start on the right. Output *12,345
&&&.&&&&	12.34	The decimal point defines how many decimal places will print. If there are fewer digits than characters then 0's are printed. Output *12.3400.
&&&.&&	12.345	If there are fewer characters than digits then the value is rounded to that number of characters. Output *12.35
\$&&&&.&&	123.45	The \$ places a \$ immediately to the left of the value. No *'s are printed. No space is allowed. Output \$123.45
*\$&&&&&&.&&	123.45	The *\$ fills the leading blank spaces with *'s and inserts the \$ as above. Output ***\$123.45
&&&E	25376	The E at the end of the &'s means to write the value in exponential form rounding when there are more digits than characters and filling in with 0's otherwise. Output 2.54E4

INPUT: There will be 5 lines of input. Each line will contain a character string and a rational number value.

OUTPUT: For each input line, print the result of implementing the formatting string on the given value.

**SAMPLE INPUT**

1. &&&&&, 456
2. &&&&&,&, 1000000
3. \$&&&&.&&, 123.38
4. &&&.&&&, 23.49
5. &&&.&&&, 23.4999
6. &&&E, 45

**SAMPLE OUTPUT**

1. \*\*\*456
2. 1,000,000
3. \$123.38
4. \*23.490
5. \*23.500
6. 4.50E1