

# Arredondamento por Valor de Corte

Por TopCoder\*  EUA

Timelimit: 1

Frequentemente, ao arredondar um número real para um inteiro nós o fazemos para cima se a parte fracionária é maior ou igual a 0,5 e para baixo se a parte fracionária é menor do que 0,5. Neste problema você recebe uma string **num** contendo um número real e uma string **cutoff** contendo um valor de corte. A string **cutoff** será formatada exatamente como "0.####", onde cada '#' representa um dígito ('0'-'9'). Pelo menos um dos dígitos da parte fracionária de **cutoff** será diferente de zero. Sua tarefa é arredondar **num** para cima se a parte fracionária é maior do que o valor de corte e para baixo caso contrário, devolvendo o resultado como um inteiro. Para evitar problemas com imprecisão de representação em ponto flutuante a parte fracionária de **num** não será exatamente igual a **cutoff**. Assim, o método tradicional de arredondamento descrito na frase inicial seria representado por **cutoff** = "0.5000"

## Entrada

A entrada contém vários casos de teste. Cada caso de teste é composto por duas linhas. A string **num** está na primeira linha e a string **cutoff** fica na segunda linha. A string **num** é formada por 1 ou mais dígitos ('0' a '9') com um ponto decimal opcional ('.'). A string **num** tem de 1 a 10 caracteres. A string **cutoff** é formatada exatamente como "0.####", onde cada '#' representa um dígito ('0' a '9'). Além disso, a parte fracionária de **num** NÃO será exatamente igual a **cutoff**.

O final da entrada é determinado por EOF.

## Saída

Para cada caso de teste da entrada seu programa deve gerar uma linha de saída somente com a parte inteira de **num** arredondada de acordo com o valor de corte em **cutoff**.

| Exemplo de Entrada | Exemplo de Saída |
|--------------------|------------------|
| 003.656930         | 4                |
| 0.5000             | 1                |
| .001               | 2                |
| 0.0001             | 135              |
| 1.99999999         | 135              |
| 0.9999             | 1357             |
| 135                | 1                |
| 0.6531             |                  |
| 135.               |                  |
| 0.6531             |                  |
| 1356.13671         |                  |
| 0.1367             |                  |
| 0.00010001         |                  |
| 0.0001             |                  |

\* Este problema é de autoria do TopCoder ([www.topcoder.com/tc](http://www.topcoder.com/tc)) e foi adaptado por M. C. Pinto para utilização (autorizada) no URI OJ.

\* A reprodução não autorizada deste problema sem o consentimento por escrito de TopCoder, Inc. é estritamente proibida.