

NÚMERO INTEIRO GIGANTE

Inteiro.cpp

Em ciência da computação, um inteiro é um dado do tipo integral, um tipo de dado que representa algum subconjunto finito dos inteiros matemáticos. Tipos de dados integrais podem ser de tamanhos diferentes e podem ou não ser permitido conterem valores negativos. Inteiros são normalmente representados em um computador como um grupo de dígitos binários. O tamanho do grupo varia de modo que o conjunto de tamanhos de inteiros disponíveis variam entre tipos diferentes de computadores. O hardware do computador quase sempre fornece uma maneira de representar um registro do processador ou endereço de memória como um inteiro.

Diferentes CPUs suportam diferentes tipos de dados inteiros. Geralmente é aceito tanto tipos com sinal quanto sem sinal, a variação é maior na largura de bits suportada. A tabela a seguir lista larguras geralmente usadas em hardware por processadores.

Largura (bits)	Bloco com Sinal	Bloco sem Sinal
8	de -128 a +127	de 0 a +255
16	de -32768 a +32767	de 0 a +65535
32	de -2147483648 a +2147483647	de 0 a +4294967295
64	de -9223372036854775808 a +9223372036854775807	de 0 a +18446744073709551615
128	de -170141183460469231731687303715884105728 a +170141183460469231731687303715884105727	de 0 a +340282366920938463463374607431768211455

Linguagens de programação de alto nível fornecem mais possibilidades. Pode-se instanciar um tipo de dado com o dobro da largura máxima permitida pela arquitetura, ou criar um campo de bits com largura menor que o mínimo definido.

Algumas linguagens suportam inteiros de precisão arbitrária (também conhecidos como bignum), que podem, inclusive, crescer infinitamente até os limites físicos de memória do sistema computacional.

Um pacote bignum permite a representação de números inteiros tão extensos quanto for desejado e fornece um conjunto de operações aritméticas sobre esses números. Em muitas bibliotecas o bignum é representado por uma cadeia de caracteres, ou seja, uma string. Implemente uma biblioteca bignum armazenando os valores em uma string. Crie um Tipo Abstrato para representar o seu bignum e implemente as seguintes funções para operarem sobre o novo tipo:

Protótipo	Descrição
<code>struct tInteiro inteiro(std::string);</code>	Recebe uma cadeia de caracteres e retorna um inteiro gigante.
<code>struct tInteiro soma(struct tInteiro, struct tInteiro);</code>	Recebe dois inteiros gigantes e retorna outro inteiro gigante com a adição de ambos.
<code>struct tInteiro sub(struct tInteiro, struct tInteiro);</code>	Recebe dois inteiros gigantes e retorna outro inteiro gigante com a subtração de ambos..
<code>struct tInteiro mult(struct tInteiro, struct tInteiro);</code>	Recebe dois inteiros gigantes e retorna outro inteiro gigante com o produto de ambos.

Entrada

A entrada é composta de várias linhas, em cada linha haverá três informações: um número inteiro positivo ou nulo N1, um operador aritmético OP e outro número inteiro positivo ou nulo N2. A largura máxima de N1 e N2 será de 100 dígitos e ambos são apresentados através de uma cadeia de caracteres e OP pode ser um dos três símbolos: +, - ou *.

A entrada termina com marcador de final de arquivo (EOF).

Saída

A saída é composta por várias linhas, cada linha com um número inteiro gigante representando o resultado da operação informada na entrada. Após a impressão do valor resultante quebre uma linha, inclusive após o último valor.

Exemplos

Entrada
1 + 1 53 - 49 120 * 5 6346 - 234 2356145 - 1235001 1564454 + 864515 165452645 - 14856525 6598565869 * 56525654656 9879999249849814684186465464984 + 19781869416782190196701000789 1789999999488118188888888888880000 - 999898845116500114878644466 7654899182491219499819819849 * 464441000001451004 6245624561354534514514515599700009860523523456 * 2 6245624561354534514514515599700009860523523456 * 1 6245624561354534514514515599700009860523523456 * 0
Saída
2 4 600 6112 1121144 2428969 150596120 372988255535962536064 9899781119266596874383166465773 178999000049966702388774010235534 3555249031226511765049208439948525557880178396 12491249122709069029029031199400019721047046912 6245624561354534514514515599700009860523523456 0