

Simulador

Nome do arquivo fonte: `simulador.c`, `simulador.cpp`, ou `simulador.pas`

Um novo processador, denominado Faíska, está sendo desenvolvido para a empresa SBC. Este novo processador tem apenas duas instruções: *inversão* e *soma*, descritas a seguir.

- *Inversão*: dados dois endereços de memória X e Y , a operação `inverte(X,Y)` inverte a posição de palavras da memória de forma que
 - a palavra no endereço X troca de posição com a palavra de memória da posição Y ;
 - a palavra no endereço $X + 1$ troca de posição com a palavra de memória da posição $Y - 1$;
 - a palavra no endereço $X + 2$ troca de posição com a palavra de memória da posição $Y - 2$;
 - e assim por diante, até que $X \geq Y$.
- *Soma*: dados dois endereços de memória X e Y , a operação `soma(X,Y)` imprime a soma das palavras de memória entre os endereços X e Y (inclusive).

Por exemplo, se a memória contém inicialmente, a partir da primeira posição de memória (endereço igual a 1) os valores `[1,2,3,4,5,6,7,8]`, a operação `inverte(3,7)` deixa a memória igual a `[1,2,7,6,5,4,3,8]`. Então, nesse estado, a execução de `soma(1,3)` produz a saída 10.

Tarefa

Sua tarefa é escrever um programa que, dada uma sequência de instruções do Faíska, simule a execução e produza o mesmo resultado que o Faíska produziria.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do *dispositivo de entrada padrão* (normalmente o teclado).

A primeira linha da entrada contém dois números inteiros N e M , representando respectivamente o número palavras na memória ($1 \leq N \leq 10^9$) e o número de instruções do programa ($1 \leq M \leq 3000$). Cada uma das M linhas seguintes contém uma instrução do Faíska. Cada instrução é composta de um caractere descrevendo a instrução ('I' para inversão e 'S' para soma), seguido de um espaço, seguido de dois inteiros indicando os argumentos da instrução.

Inicialmente a configuração da memória é tal que cada palavra tem como conteúdo o seu próprio endereço. Em outras palavras, o conteúdo inicial da memória é `[1,2,3,...,N]`. Há pelo menos uma instrução `soma` em cada caso de teste.

Saída

Seu programa deve imprimir, na *saída padrão*, uma sequência de números inteiros, um em cada linha, indicando a saída gerada pelo Faíska.

Informações sobre a pontuação

- Em um conjunto de casos de teste que totaliza 30 pontos, $N \leq 100$ e $M \leq 100$.
- Em um conjunto de casos de teste que totaliza 70 pontos, $N \leq 10^4$ e $M \leq 10^3$.

Exemplos

Entrada	Saída
10 2 I 1 5 S 3 7	19

Entrada	Saída
15 4 S 2 11 I 10 15 I 1 10 S 5 10	65 21