

## MANIPULANDO ARRAYS

ManipulandoArrays.[ c | cpp | java | cs ]

Dado um vetor **V**, indexado a partir do índice 1, de tamanho **N**, você deve realizar **M** operações sobre a lista e ao final deverá imprimir o maior valor dentre os **N** elementos do vetor. Cada operação é composta por três números inteiros **A**, **B** e **K**, e trata-se de somar o valor de **K** a todos os elementos do vetor que estiveram entre **A** e **B** (ambos inclusive). Por exemplo, considerando um vetor **V** de tamanho 3, inicialmente **V** = [0, 0, 0] após realizar a operação **A** = 2, **B** = 3 e **K** = 30, o vetor **V** após a operação será: **V** = [0, 30, 30], ou seja, o valor 30 foi somado aos elementos entre os índices 2 e 3. Atenção: o vetor inicia-se no índice 1 e seu programa deve estar otimizado, pois o limite de tempo será de 2 segundos em JAVA e outras linguagens ou 1 segundo se sua solução for em Linguagem C.

### Entrada

A entrada é composta de apenas um caso de teste.

Na primeira linha do caso de teste haverá 2 números inteiros **N** e **M**, separados por um espaço em branco, sendo que **N** representando a quantidade de elementos do vetor e **M** a quantidade de operações que será realizadas sobre o vetor.

Nas **M** próximas linhas haverá 3 números inteiros separados por um espaço em branco cada, **A**, **B** e **K**, representando o índice **A**, o índice **B** e o valor a ser adicionado aos elementos do intervalo. Atenção: o vetor é indexado de 1 até **N**.

### Restrições:

- $3 \leq N \leq 10^7$
- $1 \leq M \leq 2 \times 10^5$
- $1 \leq A \leq B \leq N$
- $0 \leq K \leq 10^9$

## Saída

A saída é composta de apenas uma linha com o número inteiro representando o maior valor presente no vetor **V** após as **M** operações. Após a impressão do valor salte uma linha.

## Exemplos

Entrada	Saída
5 3 1 2 100 2 5 100 3 4 100	200