

INSERIR ELEMENTOS NA CABEÇA DA LISTA LIGADA

`InserirElementos.[c | cpp | java | cs]`

Dado o ponteiro para o nó cabeça de uma lista ligada e um número inteiro, insira-o na cabeça da lista ligada. Você deve criar um novo nó com o inteiro dado e inseri-lo na cabeça da lista; ao final retorne um ponteiro para o nó cabeça da lista ligada modificada pela inserção do novo nó. Atenção: o ponteiro para o nó cabeça da lista ligada pode ser nulo.

Realize o exercício com a construção de um procedimento, conforme o seguinte cabeçalho:

```
void insertHead(struct Node *, int);
```

Onde o primeiro argumento é um ponteiro para o nó cabeça da lista e o segundo argumento é um inteiro a ser inserido na lista ligada.

Após construir a lista ligada, mostre-a na tela iniciando pelo nó cabeça da lista.

Entrada

O programa receberá como entrada várias listas ligadas que devem se tratadas uma por vez.

A primeira linha da entrada consiste de um inteiro **Q**, representando a quantidade de listas ligadas que a entrada possui.

A seguir haverá uma sucessão de **Q** blocos com até duas linhas cada bloco, representando cada uma das **Q** listas que você deverá tratar.

Na primeira linha de cada bloco terá um inteiro **N**, representando o tamanho da lista ligada. E na segunda linha haverá uma sucessão de **N** números inteiros

separados por um espaço em branco cada, representando os **N** elementos da lista ligada. Atenção: Se **N=0**, então a segunda linha do bloco não existirá.

Ao ler cada um dos **N** elementos, você deverá inseri-los na cauda da lista ligada.

Restrições:

$$1 \leq Q < 100$$

$$0 \leq N < 10000$$

$$-2^{31} \leq L_i \leq 2^{31}-1, \text{ sendo: } 0 \leq i < N$$

Saída

Após ler cada uma das listas ligadas e construí-las, imprima todos os seus elementos, separados por um espaço em branco cada e após o último elemento de cada lista ligada, inclusive a última, quebre uma linha.

Atenção: caso a lista ligada não possua elementos, não imprima nada.

Exemplos

Entrada	Saída
5	2 1
0	5 4 1 2
2	56 45 34
1 2	5 4 3 2 1
4	
2 1 4 5	
3	
34 45 56	
5	
1 2 3 4 5	