

Cursor do Vi

No modo normal, o cursor do editor *vi* é controlado pelas teclas:

1. **h** : pra movê-lo um caractere à esquerda
2. **j** : pra movê-lo uma linha abaixo
3. **k** : pra movê-lo uma linha acima
4. **l** : pra movê-lo um caractere à direita

Além disso, o *vi* permite digitar um valor inteiro antes do comando para indicar quantas vezes o comando deve ser repetido. Assim, para mover o cursor 11 linhas abaixo, por exemplo, basta digitar **11j**.

Pede-se que você escreva um pequeno programa que *simule* o controle do cursor do *vi*. O programa deve partir da posição inicial do cursor. Depois deve ler vários comandos de movimentação do cursor e deve ir atualizando a posição do cursor de acordo com os comandos fornecidos.

Importante Por simplicidade, você **não precisa se preocupar** com o caso em que o usuário passa para posições com valores negativos das posições de linha e coluna. Nenhum teste irá considerar esse caso.

Entrada

A primeira linha da entrada indica o número da linha e da coluna da posição inicial do cursor. As demais linhas da entrada contêm comandos e as respectivas repetições separados por um único espaço. A última linha da entrada é uma linha vazia que deve ser usada como *sentinela* indicativo do fim da entrada e não deve ser tratada como um comando de movimentação do cursor. O texto abaixo dá um exemplo de entrada válida.

```
10 10
11 j
5 l
8 h
```

Saída

A saída do programa é um texto contendo a posição do cursor após cada um dos comandos dados, formatados como indica o exemplo abaixo. A listagem abaixo dá um exemplo de saída válida (correspondente à entrada acima).

```
[21 10]
[21 15]
[21 7]
```

Exemplos de Entrada/Saída

Espera-se que o programa seja *interativo* (que apresente cada linha de saída assim que o usuário digitar o comando). Assim, quando executarmos o programa no *shell* veremos linhas de entrada e saída intercaladas, como mostra a listagem abaixo.

```
$ pythoon3 cursor.py
10 10
11 j
[21 10]
5 l
[21 15]
8 h
[21 7]

$
```

Última atualização por daltonserey, 4 horas atrás

Para enviar sua resposta:

1. faça o *checkout* da atividade com o comando `tst checkout 5317473391869952`;
2. faça a atividade, criando sua resposta em um arquivo no diretório do *checkout*;
3. faça o *commit* com o comando `tst commit (nome-do-seu-arquivo)`.

