

Lumberjarck - Cortando árvores



Um jogo muito popular nos planaltos centrais é o *Lumberjack* um jogo que testa a sua concentração agilidade e rapidez!

Este joguinho é muito simples. A ideia consiste em controlar um lenhador que vai cortando uma enorme árvore. O seu objetivo é cortar a árvore rapidamente e desviar dos galhos que vão ficando cada vez mais baixos.

A imagem acima representa uma imagem renderizada do jogo, neste momento vamos nos preocupar com a maneira em que o jogo é processado, vamos imaginar que temos uma visão matrix do jogo. Que nada mais é que uma matriz com algumas representações mais simplórias do que a imagem pronta, veja a representação abaixo:

```
~~~~~  
---|||  
   |||  
   |||---  
L  |||  
L  |||  
~~~~~
```

A imagem acima o L representa o lenhador (*Lumberjack*), que possui altura de 2 pixels. O tronco da árvore tem grossura de 3 pixels e é representado pelo caractere |. Os galhos são representados por três caracteres -. Sempre que o lenhador bate na árvore, ele começa a derrubar pela parte mais baixa. E para derrubar cada nível o lenhador precisa bater duas vezes.

Considerando a imagem acima, o lenhador está do lado esquerdo e vai começar a derrubar a árvore, após a primeira batida a árvore fica da seguinte forma:

```
~~~~~  
---|||  
   |||  
   |||---  
L  |||  
L  >||  
~~~~~
```

Após a segunda batida a árvore têm a sua base arrancada e toda a árvore cai um nível, ficando como abaixo:

```
~~~~~  
  
---|||  
   |||  
  L |||---  
  L |||  
~~~~~
```

Note que toda a árvore caiu um nível e nenhum novo tronco foi impresso, isso porque a nossa árvore tem um tamanho pré-definido.

Para este exercício você deverá implementar a visão matrix deste joguinho, ou seja, desenhar a imagem conforme a especificação abaixo:

- A imagem é representada por uma matriz de 11 colunas (sempre) e n mínimo 5 linhas;
- O tronco possui largura de três pixels, representado pelo caractere | iniciando sempre na coluna 5.
- O lenhador é representado por dois caracteres L nas duas linhas mais abaixo e sempre começa na coluna 3 ou 9;
- Os galhos são representados por 3 caracteres - que ficam ou à esquerda ou direita da árvore;

O lenhador pode receber dois comandos: - T para trocar de lado, ou seja, se ele estiver posicionado na coluna 3, muda para a coluna 9, e vice e versa; - Se o lado contrário possui um galho, o lenhador **não** troca de lado - B para bater na árvore com o machado, se o tronco mais próximo do lenhador for o caractere | ele vira o caractere > se na coluna 5 ou < se for na coluna 7. - Se já for o caracter < ou > essa linha da árvore é derrubada e todos os galhos descem um nível

Se o lenhador derrubar um nível da árvore e um galho bater na cabeça do lenhador, i.e, um galho do mesmo lado do lenhador tiver que ocupar a célula em que o L está, é fim de jogo.

No exemplo abaixo o lenhador está em uma situação perdida, pois se derrubar mais um nível o galho do lado dele vai machucá-lo:

```
~~~~~  
      |||  
      |||  
----|||  
  L  |||  
  L >||---  
~~~~~
```

Entrada

A entrada é composta por um único caso de teste. A primeira linha do caso de teste é composta por um único número inteiro L ($5 \leq L \leq 1000$) representando o tamanho total da entrada em linhas. A segunda linha é composta por um único inteiro G ($2 \leq G \leq L/2$) indicando quantos galhos estão na árvore, depois serão lidas G linhas, cada uma, contendo 2 inteiros G_{Lado} ($G_{Lado} \in \{E, D\}$) e G_{Linha} ($1 \leq G_{Linha} \leq L$) representando, respectivamente, o lado que o galho está, **D** para direito e **E** para esquerdo, e a linha em que o galho está. A seguir uma linha contendo um único caractere indicando o lado em que o lenhador começa. A seguir será lida uma única linha contendo uma string C (de tamanho $1 \leq |C| \leq 2000$) contendo os comandos que o lenhador recebeu.

Saída

Primeiramente, o seu programa, deverá imprimir as primeiras 5 linhas do estado inicial, e depois deverá mostrar o estado após a execução de cada comando. Desenhe um conjunto de 11 ~ no início e fim da tela do jogo.

Caso o galho bata no lenhador, você deve imprimir a mensagem ****morreu****, caso um seja feito um comando inválido, imprima ****beep****.

O programa deverá ser finalizado após a mensagem ****morreu****.

Consulte os exemplos para entender melhor o formato de saída

Exemplos

Exemplo de entrada

```
10  
3  
D 3  
E 6
```

D 9
D
TBBBBBBTBB

Saída para o exemplo de entrada acima

```

~~~~~
    |||
    |||
    ||| ---
    ||| L
    ||| L
~~~~~
~~~~~
    |||
    |||
    ||| ---
L  |||
L  |||
~~~~~
~~~~~
    |||
    |||
    ||| ---
L  |||
L >||
~~~~~
~~~~~
---|||
    |||
    |||
L  ||| ---
L  |||
~~~~~
~~~~~
---|||
    |||
    |||
L  ||| ---
L >||
~~~~~
~~~~~
    |||
---|||
    |||
L  |||
L  ||| ---
~~~~~
~~~~~
    |||
---|||
    |||
L  |||
L >|| ---
~~~~~
~~~~~
    |||
    |||
---|||
L  |||
L  |||
~~~~~
~~~~~
    |||

```

```

    |||
---|||
    ||| L
    ||| L
~~~~~
~~~~~
    |||
    |||
---|||
    ||| L
    ||< L
~~~~~
~~~~~
    ||| ---
    |||
    |||
---||| L
    ||| L
~~~~~

```

Exemplo de entrada

```

10
3
D 3
E 3
D 9
D
BBTT

```

Saída para o exemplo de entrada acima

```

~~~~~
    |||
    |||
---||| ---
    ||| L
    ||| L
~~~~~
~~~~~
    |||
    |||
---||| ---
    ||| L
    ||< L
~~~~~
**morreu**

```

Exemplo de entrada

```

10
3
D 2
E 6
D 9
E
T

```

Saída para o exemplo de entrada acima

```

~~~~~
    |||
    |||
    |||
L ||| ---

```

```

    L |||
~~~~~
**beep**

```

Exemplo de entrada

```

10
1
D 2
E
BBBBBBBBBBBBBBBBBB

```

Saída para o exemplo de entrada acima

```

~~~~~
    |||
    |||
    |||
L ||| ---
L |||
~~~~~
~~~~~
    |||
    |||
    |||
L ||| ---
L >||
~~~~~
~~~~~
    |||
    |||
    |||
L ||| ---
L >||
~~~~~
~~~~~
    |||
    |||
    |||
L ||| ---
L >||
~~~~~
~~~~~
    |||
    |||
    |||
L |||
L |||
~~~~~
~~~~~
    |||
    |||
    |||
L |||
L >||
~~~~~
~~~~~
    |||
    |||
    |||
L |||
L |||
~~~~~
~~~~~
    |||

```

```

      |||
      |||
    L |||
    L >||
~~~~~
~~~~~

      |||
      |||
      |||
    L |||
    L |||
~~~~~
~~~~~

      |||
      |||
      |||
    L |||
    L >||
~~~~~
~~~~~

      |||
      |||
      |||
    L |||
    L |||
~~~~~
~~~~~

      |||
      |||
      |||
    L |||
    L >||
~~~~~
~~~~~

      |||
      |||
    L |||
    L |||
~~~~~
~~~~~

      |||
      |||
    L |||
    L >||
~~~~~
~~~~~

      |||
      |||
    L |||
    L |||
~~~~~
~~~~~

      |||
      |||
    L |||
    L >||
~~~~~
~~~~~

```

L |||
L |||
~~~~~  
~~~~~

L |||
L >||
~~~~~  
~~~~~

L
L |||
~~~~~

*Author: Bruno Ribas*