

Disciplina: Programação Estruturada. Prof. Flavio Barbieri Gonzaga.

Data de Entrega: 20/02/2017. Valor: 2,5 pontos.

Os critérios de correção foram explicados na vídeo aula, disponível no Moodle.

Ao longo dos últimos anos têm acontecido uma revolução na tecnologia de interface com usuário. A principal causa dessa revolução consiste na popularização de smart phones, e por consequência, pela introdução de telas sensíveis ao toque. Essa nova forma de interação com a máquina vem substituindo por muitas vezes o mouse e o teclado, pelo menos, fisicamente.

Você foi contratado para contruir um simulador para o gerenciamento de janelas, a ser usado em uma próxima geração de smart phones. Essa geração possuirá uma tela retangular, que poderá exibir zero ou mais janelas de aplicativos. O seu simulador deve suportar as seguintes operações:

- **ABRE** x y w h - abre uma nova janela cujo canto superior esquerdo se localiza nas coordenadas (x, y) , contendo w pixels de largura, e h pixels de altura.
- **FECHA** x y - fecha uma janela aberta que inclua o pixel localizado nos pontos (x, y) . Isso permite ao usuário tocar em qualquer ponto da janela para fechá-la.
- **REDIMENSIONA** x y w h - configura as dimensões da janela que possua os pixels (x, y) para a largura w e altura h . A coordenada do pixel no canto superior esquerdo não é modificada.
- **MOVE** x y dx dy - move a janela que possua o pixel (x, y) . O movimento é pelo menos dx pixels na horizontal, ou dy pixels na vertical, podendo inclusive ser ambos.

Os comandos **ABRE** e **REDIMENSIONA** só podem ser executados se a janela resultante não sobrepor nenhuma outra janela, e não se estender além dos limites da tela. O comando **MOVE** moverá a janela a quantidade de pixels que for possível. Por exemplo, se dx for 30, mas a janela só puder mover 15 pixels, então a mesma será movida por 15 pixels.

O comando **MOVE** tem ainda a propriedade de empurrar outras janelas. Veja a Figura 1, onde a janela A recebe no comando **MOVE** uma quantidade de pixels em que ela empurra as janelas B e C.

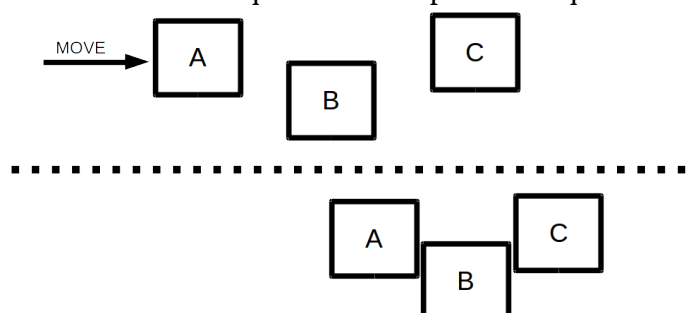


Figura 1 – Exemplo do comando **MOVE**.

Caso o comando **MOVE** seja executado em diagonal, a janela que estiver no caminho daquela que está sendo movida também será empurrada, no mesmo sentido da janela original. Esse cenário é ilustrado na Figura 2, onde o comando **MOVE** é executado na Janela A, e a mesma encontrará a Janela B em seu caminho.

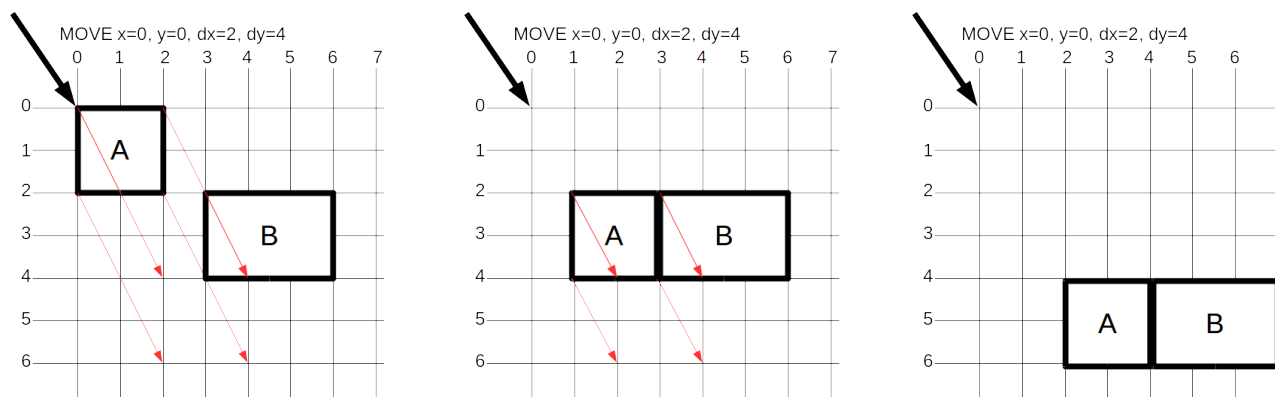


Figura 2 – Janela A sendo movida na diagonal e empurrando a Janela B.

Entrada de dados

O seu programa deverá ler um arquivo chamado entrada.txt. Neste arquivo haverão até 10 projetos diferentes de smart phones, cada projeto com o seu tamanho de tela, e as respectivas operações. A primeira linha de cada projeto contém dois números inteiros positivos x_{max} e y_{max} , que indicam as dimensões horizontal e vertical da tela, medidas em pixels. As variáveis x_{max} e y_{max} são limitadas ao valor máximo de 10^9 . O pixel localizado no topo à esquerda da tela possui as coordenadas (0, 0). As demais linhas do arquivo possuem os comandos a serem executados no projeto. Os comandos são aqueles já descrito anteriormente na página 1 da presente proposta, com os respectivos parâmetros. Um ou mais espaços separam o nome do comando e os parâmetros. Existem também um ou mais espaços entre os parâmetros. Os parâmetros dos comandos são números inteiros, que satisfaçam às seguintes condições: $0 \leq x < x_{max}$, $0 \leq y < y_{max}$, $1 \leq w, h \leq 10^9$, e $|dx|, |dy| \leq 10^9$. Para cada projeto poderão haver no máximo 256 comandos. O que separa um projeto do outro é a existência de uma linha em branco entre os mesmos.

Saída de dados

Execute a simulação dos comandos na ordem em que aparecem no arquivo de entrada. Caso ocorra algum erro, exiba o número do comando, o nome do comando, a mensagem adequada da seguinte lista, e ignore os resultados de se simular aquele comando.

- nao ha janela na posicao – saída gerada para os comandos FECHA, REDIMENSIONA e MOVE – caso não haja janela que inclua o pixel especificado.
- janela nao cabe – para os comandos ABRE e REDIMENSIONA – se a janela resultante for sobrepor outra janela, ou se estender além dos limites da tela.
- movido d' ao inves de d – para o comando MOVE – se o comando pedir para se mover uma janela d pixels, mas ela só puder ser movida d', pois encontrou o limite da tela. Os valores de d e d' são os números absolutos de pixels requisitados e movidos respectivamente. A janela permanece movida nesse caso, mas somente para a distância d'.

Após a execução de todos os comandos, e a exibição de possíveis erros, indique o número de janelas que permanecem abertas. Para cada janela aberta, exiba as coordenadas do canto superior esquerdo da mesma, bem como a sua largura e altura. Toda a saída deve ser exibida na tela, e gravada em um arquivo chamado saida.txt.

Exemplo de entrada (contendo dois projetos idênticos, apenas para fins ilustrativos):
entrada.txt

```
320 200
ABRE 50 50 10 10
ABRE 70 55 10 10
ABRE 90 50 10 10
REDIMENSIONA 55 55 40 40
REDIMENSIONA 55 55 15 15
MOVE 55 55 40 0
FECHA 55 55
FECHA 110 60
MOVE 95 55 0 -100

320 200
ABRE 50 50 10 10
ABRE 70 55 10 10
ABRE 90 50 10 10
REDIMENSIONA 55 55 40 40
REDIMENSIONA 55 55 15 15
MOVE 55 55 40 0
FECHA 55 55
FECHA 110 60
MOVE 95 55 0 -100
```

Saída esperada para as entradas contidas no arquivo entrada.txt:

saida.txt

```
Comando 4: REDIMENSIONA - janela nao cabe
Comando 7: FECHA - nao ha janela na posicao
Comando 9: MOVE - movido 50 ao inves de 100
2 janela(s):
90 0 15 15
115 50 10 10

Comando 4: REDIMENSIONA - janela nao cabe
Comando 7: FECHA - nao ha janela na posicao
Comando 9: MOVE - movido 50 ao inves de 100
2 janela(s):
90 0 15 15
115 50 10 10
```

Referência Bibliográfica: Desafio de programação baseado na ACM International Collegiate Programming Contest.

Bom trabalho!