

Instruções

LEIA ATENTAMENTE ESTAS INSTRUÇÕES ANTES DE INICIAR A PROVA

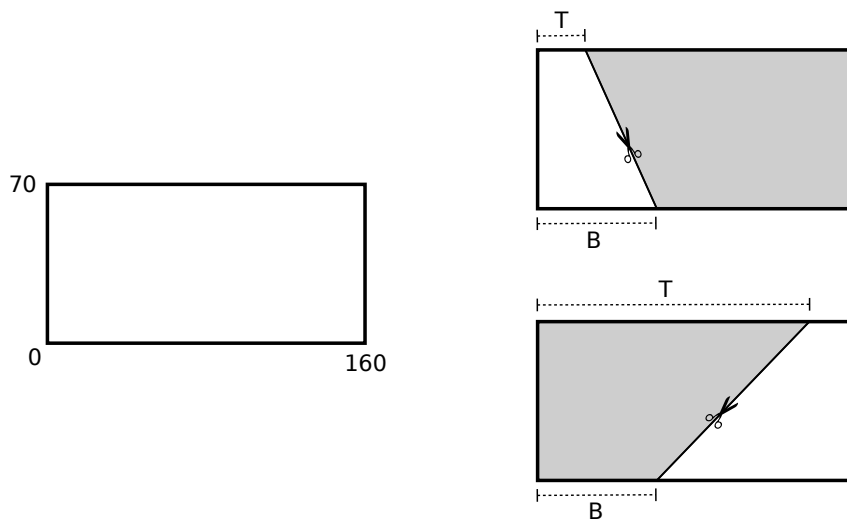
- Este caderno de tarefas é composto por 7 páginas (não contando a folha de rosto), numeradas de 1 a 7. Verifique se o caderno está completo.
- A prova deve ser feita individualmente.
- É proibido consultar a Internet, livros, anotações ou qualquer outro material durante a prova. É permitida a consulta ao *help* do ambiente de programação se este estiver disponível.
- As tarefas têm o mesmo valor na correção.
- A correção é automatizada, portanto siga atentamente as exigências da tarefa quanto ao formato da entrada e saída de seu programa; em particular, seu programa não deve escrever frases como “Digite o dado de entrada:” ou similares.
- Não implemente nenhum recurso gráfico nas suas soluções (janelas, menus, etc.), nem utilize qualquer rotina para limpar a tela ou posicionar o cursor.
- As tarefas **não** estão necessariamente ordenadas, neste caderno, por ordem de dificuldade; procure resolver primeiro as questões mais fáceis.
- Preste muita atenção no nome dos arquivos fonte indicados nas tarefas. Soluções na linguagem C devem ser arquivos com sufixo *.c*; soluções na linguagem C++ devem ser arquivos com sufixo *.cc* ou *.cpp*; soluções na linguagem Pascal devem ser arquivos com sufixo *.pas*; soluções na linguagem Java devem ser arquivos com sufixo *.java* e a classe principal deve ter o mesmo nome do arquivo fonte; soluções na linguagem Python 2 devem ser arquivos com sufixo *.py2*; soluções na linguagem Python 3 devem ser arquivos com sufixo *.py3*; e soluções na linguagem Javascript devem ter arquivos com sufixo *.js*.
- Na linguagem Java, **não** use o comando *package*, e note que o nome de sua classe principal deve usar somente letras minúsculas (o mesmo nome do arquivo indicado nas tarefas).
- Para tarefas diferentes você pode escolher trabalhar com linguagens diferentes, mas apenas uma solução, em uma única linguagem, deve ser submetida para cada tarefa.
- Ao final da prova, para cada solução que você queira submeter para correção, copie o arquivo fonte para o seu diretório de trabalho ou pen-drive, conforme especificado pelo seu professor.
- Não utilize arquivos para entrada ou saída. Todos os dados devem ser lidos da entrada padrão (normalmente é o teclado) e escritos na saída padrão (normalmente é a tela). Utilize as funções padrão para entrada e saída de dados:
 - em Pascal: *readln*, *read*, *writeln*, *write*;
 - em C: *scanf*, *getchar*, *printf*, *putchar*;
 - em C++: as mesmas de C ou os objetos *cout* e *cin*.
 - em Java: qualquer classe ou função padrão, como por exemplo *Scanner*, *BufferedReader*, *BufferedWriter* e *System.out.println*
 - em Python: *read*, *readline*, *readlines*, *input*, *print*, *write*
 - em Javascript: *scanf*, *printf*
- Procure resolver a tarefa de maneira eficiente. Na correção, eficiência também será levada em conta. As soluções serão testadas com outras entradas além das apresentadas como exemplo nas tarefas.

Nota cortada

Nome do arquivo: “**nota.x**”, onde **x** deve ser **c**, **cpp**, **pas**, **java**, **js**, **py2** ou **py3**

Se pegarmos uma nota de 100 reais e a cortarmos, usando uma tesoura, em dois pedaços, quanto vale cada um dos pedaços? A regra é simples: se um dos pedaços possuir estritamente mais da metade da área da nota original, então ele vale 100 reais; e o outro pedaço não vale nada. Veja que se cada pedaço possuir exatamente metade da área original, então nenhum dos dois tem valor.

Felix e Marzia decidiram fazer um corte, em linha reta, que comece no lado inferior da nota, a base, e termine no lado superior, o topo. A nota é um retângulo de comprimento 160 centímetros e altura 70 centímetros, como mostrado na parte esquerda da figura abaixo. Felix sempre vai ficar com o pedaço mais à esquerda da nota e Marzia com o pedaço mais à direita. A parte direita da figura ilustra dois possíveis cortes. No de cima, Marzia ficaria claramente com o maior pedaço, que vale 100 reais; e no de baixo, dá para ver que Felix é quem ficaria com o maior pedaço.



O corte reto vai começar na base a uma distância de B centímetros a partir do lado esquerdo da nota; e terminar no topo a uma distância de T centímetros também a partir do lado esquerdo da nota. Veja a indicação na parte direita da figura.

Neste problema, dados os valores B e T , seu programa deve computar quem vai ficar com o pedaço que vale 100 reais, ou se o valor da nota se perdeu.

Entrada

A primeira linha da entrada contém um inteiro B representando a distância do ponto inicial do corte, na base, para o lado esquerdo da nota. A segunda linha da entrada contém um inteiro T representando a distância do ponto final do corte, no topo, para o lado esquerdo da nota.

Saída

Seu programa deve imprimir uma linha contendo um número inteiro: 1, se Felix ficou com o pedaço que vale 100 reais; 2, se Marzia ficou com o pedaço que vale 100 reais; ou 0, se o valor da nota se perdeu.

Restrições

- $0 < B < 160$
- $0 < T < 160$

Informações sobre a pontuação

- Em um conjunto de casos de teste somando 25 pontos, $B = T$