

Competidor(a): _____

Número de inscrição: _____ – _____ (opcional)



OBI2019

Caderno de Tarefas

Modalidade **Programação • Nível Júnior • Fase Estadual**

14 de agosto de 2019

A PROVA TEM DURAÇÃO DE **2 HORAS**

Promoção:



Sociedade Brasileira de Computação

Apoio:



Instruções

LEIA ATENTAMENTE ESTAS INSTRUÇÕES ANTES DE INICIAR A PROVA

- Este caderno de tarefas é composto por 5 páginas (não contando a folha de rosto), numeradas de 1 a 5. 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 tarefas 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.py*; soluções na linguagem Python 3 devem ser arquivos com sufixo *_py3.py*; 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 esquecida

Nome do arquivo: `nota.c`, `nota.cpp`, `nota.pas`, `nota.java`, `nota.js`, `nota_py2.py` ou `nota_py3.py`

João aprendeu na escola que a média de dois números é o valor da soma desses dois números dividido por dois. Ou seja, a média de dois números A e B é $M = \frac{A+B}{2}$.

A professora contou para João as notas que ele tirou nas duas provas de Geografia. As duas notas são números inteiros entre 0 e 100. João prontamente calculou a média das duas provas, que também resultou em um número inteiro.

Mas João é muito esquecido, e agora não consegue lembrar-se das duas notas que tirou na prova. Ele consegue se lembrar de apenas uma das notas das provas. Por sorte, ele consegue se lembrar também da média entre as duas notas.

Você pode ajudar João a determinar a nota da outra prova?

Entrada

A primeira linha contém um número inteiro A , indicando a nota de uma prova. A segunda linha contém um número inteiro M , indicando a média entre as duas notas das provas.

Saída

Seu programa deve produzir uma única linha, com apenas um número inteiro, a nota da outra prova, que João não consegue recordar.

Restrições

- $0 \leq A \leq 100$
- $0 \leq M \leq 100$

Informações sobre a pontuação

- Para um conjunto de casos de testes valendo 10 pontos, as duas notas das provas são iguais.

Exemplo de entrada 1 100 70	Exemplo de saída 1 40
Exemplo de entrada 2 80 75	Exemplo de saída 2 70
Exemplo de entrada 3 1 50	Exemplo de saída 3 99

Tabela do campeonato

Nome do arquivo: `tabela.c`, `tabela.cpp`, `tabela.pas`, `tabela.java`, `tabela.js`, `tabela_py2.py` ou `tabela_py3.py`

Joana adora futebol e conseguiu um estágio no clube pelo qual torce. Na sala de troféus Joana encontrou uma caderneta em que um antigo conselheiro do clube anotou os resultados de todos os campeonatos que o time participou.

Para cada campeonato há uma anotação na caderneta informando cinco dados: o número de jogos, o número de pontos, o número de vitórias, o número de empates e o número de derrotas do time naquele campeonato. Em todos os campeonatos, cada vitória vale três pontos, cada empate vale um ponto e derrotas não valem pontos.

Infelizmente, a caderneta era muito velha e alguns desses números estavam ilegíveis. Joana, no entanto, percebeu que, para cada campeonato, se algum número estivesse ilegível ele poderia ser computado utilizando os números legíveis daquele campeonato.

Sua tarefa é escrever um programa para completar as informações encontradas na caderneta.

Entrada

A entrada consiste de uma única linha que contém cinco inteiros J, P, V, E e D , indicando respectivamente o número de jogos, o número de pontos, o número de vitórias, o número de empates e o número de derrotas para um campeonato. Um ou dois desses números tem o valor -1, indicando que o número é ilegível na caderneta. Todos os casos da entrada têm solução.

Saída

Seu programa deve produzir uma única linha, com exatamente cinco números inteiros, os valores corrigidos da caderneta para o campeonato da entrada. Os números devem ser dados na mesma ordem fornecida na entrada: o número de jogos, o número de pontos, o número de vitórias, o número de empates e o número de derrotas, separados por exatamente um espaço em branco. Note que no resultado devem valer as equações $J = V + E + D$ e $P = 3 \times V + E$.

Restrições

- $1 \leq J \leq 100$; $0 \leq P \leq 300$; $0 \leq V \leq 100$; $0 \leq E \leq 100$; $0 \leq D \leq 100$.
- Na entrada, um ou dois dos valores entre J, P, V, E e D é igual a -1 (e somente nesses casos as restrições acima não são obedecidas).

Informações sobre a pontuação

- Para um conjunto de casos de testes valendo 10 pontos, $P, V, E, D \geq 0$ e $J = -1$.
- Para um conjunto de casos de testes valendo outros 40 pontos, apenas um dos valores da entrada é -1 .

<p>Exemplo de entrada 1</p> <p>10 20 6 2 -1</p>	<p>Exemplo de saída 1</p> <p>10 20 6 2 2</p>
<p>Exemplo de entrada 2</p> <p>-1 64 18 10 10</p>	<p>Exemplo de saída 2</p> <p>38 64 18 10 10</p>
<p>Exemplo de entrada 3</p> <p>-1 47 14 -1 9</p>	<p>Exemplo de saída 3</p> <p>28 47 14 5 9</p>

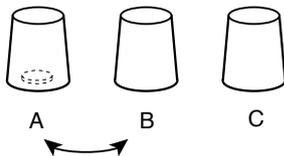
Jogo dos copos

Nome do arquivo: `copos.c`, `copos.cpp`, `copos.pas`, `copos.java`, `copos.js`, `copos_py2.py` ou `copos_py3.py`

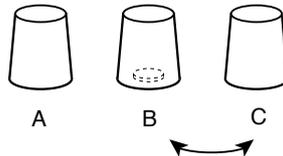
Uma brincadeira muito comum e divertida entre dois jogadores usa uma moeda e três copos opacos (ou seja, não é possível ver o que está dentro do copo olhando pela lateral do copo). Os três copos são colocados com a boca para baixo, em uma linha, um ao lado do outro, em posições que vamos chamar de A, B e C. Uma moeda é colocada embaixo de um dos copos.

Na brincadeira, um jogador chamado banca realiza um movimento para trocar a posição de dois copos, arrastando os copos de tal modo que se a moeda está em baixo de um dos copos envolvidos no movimento, ela continua embaixo do mesmo copo após a troca de posição. O jogador banca pode realizar três tipos de movimento, ilustrados na figura abaixo:

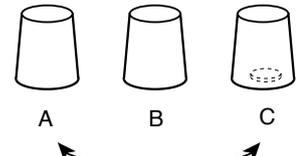
1. Trocar o copo na posição A com o copo na posição B.
2. Trocar o copo na posição B com o copo na posição C.
3. Trocar o copo na posição A com o copo na posição C.



Movimento de tipo 1:
troca os copos nas posições A e B



Movimento de tipo 2:
troca os copos nas posições B e C



Movimento de tipo 3:
troca os copos nas posições A e C

O jogador banca realiza vários movimentos de troca tentando confundir o outro jogador, chamado espectador. Ao final o jogador espectador deve dizer em qual posição está a moeda.

Por exemplo, considere que inicialmente a moeda está embaixo do copo na posição A e que o jogador banca realiza uma sequência de apenas três trocas, executando um movimento do tipo 1, após o qual a moeda termina embaixo do copo na posição B, seguido de um movimento do tipo 2, após o qual a moeda termina embaixo do copo na posição C, seguido de um movimento do tipo 3, após o qual a moeda termina embaixo do copo na posição A.

Nesta tarefa, dadas a descrição da sequência de movimentos e a posição inicial da moeda, você deve escrever um programa que determine a posição final da moeda após todos os movimentos.

Entrada

A primeira linha contém um inteiro N , o número de movimentos que o jogador banca realiza. A segunda linha contém um caractere, entre A, B e C, indicando a posição inicial da moeda. Cada uma das N linhas seguintes contém um inteiro, indicando o tipo de movimento efetuado pelo jogador banca na sequência.

Saída

Seu programa deve produzir uma única linha, com um único caractere entre A, B e C, a posição em que a moeda se encontra ao final da sequência de movimentos.

Restrições

- $1 \leq N \leq 1000$

Exemplo de entrada 1 3 A 1 2 3	Exemplo de saída 1 A
Exemplo de entrada 2 6 C 1 2 3 3 1 1	Exemplo de saída 2 B
Exemplo de entrada 3 1 B 3	Exemplo de saída 3 B