

Competidor(a): _____

Número de inscrição: _____ – _____ (opcional)



OBI2020

Caderno de Tarefas

Modalidade **Programação** • **Nível Júnior** • **Fase Local** • **Turno B**

15 e 16 de setembro de 2020

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 6 páginas (não contando a folha de rosto), numeradas de 1 a 6. 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.

Relógio de Atleta

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

Uma empresa está desenvolvendo um novo relógio eletrônico para atletas de alto desempenho. Uma das funcionalidades do novo relógio é que ele vai medir a *frequência cardíaca atual* (batidas do coração por minuto) e a *capacidade de oxigenação atual* do atleta. O relógio também vai calcular e armazenar a *frequência cardíaca em repouso* do atleta.

Os projetistas querem que o relógio emita um aviso para o atleta de que ele ou ela deve:

- *diminuir* o ritmo do exercício se a frequência cardíaca atual é maior do que três vezes a frequência cardíaca em repouso **ou** a capacidade de oxigenação atual é menor do que 95;
- *aumentar* o ritmo do exercício se a frequência cardíaca atual é menor do que duas vezes frequência cardíaca em repouso **e** a capacidade de oxigenação atual é maior do que 97;
- *manter* o ritmo de exercício se nenhuma das condições anteriores ocorrer.

Dadas a frequência cardíaca em repouso, a frequência cardíaca atual e a capacidade de oxigenação atual obtidas pelo relógio, escreva um programa que produza a sugestão que o relógio deve emitir.

Entrada

A entrada é composta por três linhas, cada uma contendo um único número inteiro. A primeira linha contém o inteiro R , a frequência cardíaca em repouso do atleta. A segunda linha contém o inteiro F , a frequência cardíaca atual do atleta. A terceira linha contém o inteiro C , a capacidade de oxigenação atual do atleta.

Saída

Seu programa deve produzir uma única linha, contendo uma única palavra, em letras minúsculas, que deve ser ‘aumentar’, ‘diminuir’, ou ‘manter’, de acordo com os critérios estabelecidos acima.

Restrições

- $35 \leq R \leq 100$
- $35 \leq F \leq 200$
- $50 \leq C \leq 100$

<p>Exemplo de entrada 1</p> <p>60 119 98</p>	<p>Exemplo de saída 1</p> <p>aumentar</p>
<p>Exemplo de entrada 2</p> <p>60 190 100</p>	<p>Exemplo de saída 2</p> <p>diminuir</p>
<p>Exemplo de entrada 3</p> <p>50 100 95</p>	<p>Exemplo de saída 3</p> <p>manter</p>

Exemplo de entrada 4	Exemplo de saída 4
50 100 94	diminuir

Divisão do Tesouro

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

O Capitão Olho Roxo e seus marinheiros encontraram uma arca com uma grande quantidade de moedas de ouro idênticas. Para a divisão das moedas, todos concordaram com a seguinte sugestão do Capitão:

- cada marinheiro exceto o Capitão deveria receber exatamente o mesmo número de moedas; e
- o Capitão deveria receber o dobro de moedas que um marinheiro recebe.

Pode ser que o fato de o Capitão ser o único com uma pistola a bordo tenha contribuído para a concordância de todos, mas também contribuiu o fato de que na forma proposta a divisão era perfeita, não sobrando ou faltando moedas.

Dados o número de moedas na arca e o número de marinheiros, escreva um programa para determinar quantas moedas o Capitão Olho Roxo recebeu.

Entrada

A primeira linha da entrada contém um número inteiro A , o número de moedas na arca. A segunda linha contém um inteiro N , o número de marinheiros (não contando o Capitão).

Saída

Seu programa deve produzir na saída uma única linha, contendo um único inteiro, o número de moedas que o Capitão Olho Roxo deve receber.

Restrições

- $3 \leq A \leq 10000$
- $1 \leq N \leq 1000$

<p>Exemplo de entrada 1</p> <p>221 11</p>	<p>Exemplo de saída 1</p> <p>34</p>
<p>Exemplo de entrada 2</p> <p>1000 8</p>	<p>Exemplo de saída 2</p> <p>200</p>
<p>Exemplo de entrada 3</p> <p>3 1</p>	<p>Exemplo de saída 3</p> <p>2</p>

Camisetas da Olimpíada

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

A Olimpíada Municipal de Programação vai distribuir camisetas para os melhores colocados, e por isso solicitou que os premiados informassem o tamanho preferido da camiseta, entre os tamanhos pequeno e médio.

A empresa que confeccionou as camisetas, por uma falha, pode ter se enganado na quantidade de camisetas para cada tamanho. Foram produzidas camisetas em número suficiente para todos os premiados, mas talvez não do tamanho preferido.

Dadas a lista com os tamanhos preferidos pelos premiados e a quantidade de camisetas de cada tamanho produzidas pela empresa, escreva um programa para determinar se todos os premiados receberão camisetas do tamanho escolhido.

Entrada

A primeira linha contém um inteiro N , o número de premiados. A segunda linha contém N inteiros T_i , indicando os tamanhos solicitados pelos premiados, sendo que $T_i = 1$ representa o tamanho pequeno e $T_i = 2$ representa o tamanho médio. A terceira linha contém um inteiro P , o número de camisetas de tamanho pequeno produzidas. A quarta e última contém um inteiro M , o número de camisetas de tamanho médio produzidas.

Saída

Seu programa deve produzir uma única linha, contendo um único caractere, que deve ser a letra maiúscula ‘S’ se todos os premiados serão atendidos com a camiseta do tamanho que escolheram, ou a letra maiúscula ‘N’ caso contrário.

Restrições

- $1 \leq N \leq 1000$
- $0 \leq P \leq 1000$
- $0 \leq M \leq 1000$
- $N \leq P + M$
- $1 \leq X_i \leq 2$ para $1 \leq i \leq N$

Informações sobre a pontuação

- Para um conjunto de casos de testes valendo 20 pontos, $1 \leq N \leq 10$.

<p>Exemplo de entrada 1</p> <p>5 1 1 2 1 2 3 2</p>	<p>Exemplo de saída 1</p> <p>S</p>
<p>Exemplo de entrada 2</p> <p>4 2 2 2 2 1 3</p>	<p>Exemplo de saída 2</p> <p>N</p>

Exemplo de entrada 3	Exemplo de saída 3
6 1 1 1 2 1 1 4 4	N