

Competidor(a): \_\_\_\_\_

Número de inscrição: \_\_\_\_\_ – \_\_\_\_\_ (opcional)



# OBI2021

## Caderno de Tarefas

Modalidade Programação • Nível 1 • Fase 2 (Turno B)

1 de setembro de 2021

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 8 páginas (não contando a folha de rosto), numeradas de 1 a 8. 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 3 devem ser arquivos com sufixo *.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.

# Recorde

Nome do arquivo: “`recorde.x`”, onde  $x$  deve ser `c`, `cpp`, `pas`, `java`, `js` ou `py`

Atletas conseguem resultados cada vez melhores! O *recorde mundial* de uma determinada modalidade esportiva é o melhor resultado conseguido por um atleta nessa modalidade, em competições oficiais. Competições oficiais incluem campeonatos mundiais, como os campeonatos mundiais de ginástica, atletismo ou natação, e também as Olimpíadas.

Como as Olimpíadas acontecem a cada quatro anos e competições oficiais acontecem todos os anos, é possível que o melhor resultado obtido em Olimpíadas em uma dada modalidade seja um resultado pior do que o recorde mundial para aquela modalidade. Por isso, nas provas das Olimpíadas são sempre mencionados dois recordes: o recorde olímpico (melhor resultado que já foi obtido em Olimpíadas) e o recorde mundial (melhor resultado em qualquer competição oficial, incluindo as Olimpíadas).

Nesta tarefa, dados o resultado de uma prova nas Olimpíadas e os recordes mundial e olímpico para essa prova, sua tarefa é determinar se o resultado é um novo recorde mundial e/ou um novo recorde olímpico.

## Entrada

A entrada é composta por três linhas. A primeira linha é um inteiro  $R$ , o melhor resultado obtido por um atleta numa prova das Olimpíadas. A segunda linha é um inteiro  $M$ , o recorde mundial para essa prova. A terceira linha é um inteiro  $L$ , o recorde olímpico para essa prova. Para as provas desta tarefa, quanto menor o valor melhor o resultado.

## Saída

Seu programa deve produzir duas linhas. A primeira linha deve ser `RM` se o resultado é um recorde mundial, ou `*` (asterisco) caso contrário. A segunda linha deve ser `RO` se o resultado é um recorde olímpico, ou `*` (asterisco) caso contrário.

## Restrições

- $1 \leq R \leq 1000$
- $1 \leq M \leq 1000$
- $1 \leq L \leq 1000$

## Exemplos

<b>Exemplo de entrada 1</b> 97 95 98	<b>Exemplo de saída 1</b> * RO
<b>Exemplo de entrada 2</b> 10 10 10	<b>Exemplo de saída 2</b> * *

<b>Exemplo de entrada 3</b>	<b>Exemplo de saída 3</b>
78 80 81	RM RO

# Potência

Nome do arquivo: “potencia.x”, onde  $x$  deve ser `c`, `cpp`, `pas`, `java`, `js` ou `py`

A profa. Vilma preparou uma tarefa de programação sobre a *operação de potenciação*. Para lembrar, seja um número real  $n$  e um número inteiro  $p$  igual ou maior do que zero, então a operação de potenciação  $n^p$  tem o valor de  $n$  multiplicado por ele mesmo  $p$  vezes (se  $p = 0$  o resultado da operação de potenciação é 1). Por exemplo,  $2^3 = 2 \times 2 \times 2 = 8$  e  $102^0 = 1$ .

A tarefa preparada pela profa. Vilma foi a seguinte: *Escreva um programa para calcular o valor das seguintes expressões contendo operações de potenciação:*

$$\begin{array}{c} 2^4 + 12^3 \\ 300^3 + 15^2 + 4^2 \end{array}$$

Veja que cada termo das expressões tem a forma  $n^p$  onde  $n$  e  $p$  são números inteiros e  $p$  tem apenas um dígito decimal.

No entanto, quando a profa. Vilma colocou o enunciado da tarefa na Internet, a formatação do enunciado foi corrompida, fazendo com que as expressões aparecessem assim para os alunos:

$$\begin{array}{c} 24 + 123 \\ 3003 + 152 + 42 \end{array}$$

Note que por exemplo  $2^4$  virou 24,  $12^3$  virou 123,  $300^3$  virou 3003 e assim por diante, ou seja, as operações de potenciação desapareceram!

Nesta tarefa, você deve escrever um programa para calcular o valor das expressões da tarefa original da profa. Vilma, sabendo que a formatação do enunciado foi corrompida conforme explicado acima.

## Entrada

A primeira linha da entrada contém um número inteiro  $N$ , o número de termos da expressão. Cada uma das  $N$  linhas seguintes contém um inteiro  $T_i$ , indicando um termo da expressão com formatação corrompida.

## Saída

Seu programa deve produzir uma linha, contendo um único número inteiro, o valor da soma dos termos da expressão, sabendo que a formatação dos termos foi corrompida como explicado acima.

## Restrições

- $1 \leq N \leq 10$
- $10 \leq T_i \leq 9999$  para  $1 \leq i \leq N$
- O resultado é menor do que  $10^9$ .

## Exemplos

Exemplo de entrada 1	Exemplo de saída 1
2 24 123	1744

<b>Exemplo de entrada 2</b> 3 303 152 42	<b>Exemplo de saída 2</b> 27241
<b>Exemplo de entrada 3</b> 6 12 13 24 20 31 43	<b>Exemplo de saída 3</b> 86

# Cálculo rápido

Nome do arquivo: “calculou.x”, onde  $x$  deve ser `c`, `cpp`, `pas`, `java`, `js` ou `py`

Algumas pessoas conseguem fazer cálculos matemáticos com uma velocidade impressionante. Laurinha tem essa habilidade! Um cálculo que ela consegue fazer muito rapidamente é, dados três números inteiros  $S$ ,  $A$ , e  $B$ , determinar quantos números do intervalo  $[A, B]$  têm a soma de seus dígitos igual a  $S$ .

Por exemplo, se  $S = 3$ ,  $A = 10$  e  $B = 30$ , então a resposta é 3, pois existem três números no intervalo  $[10, 30]$  cuja soma dos dígitos é igual a três: 12, 21 e 30.

Sua tarefa é escrever um programa de computador para, dados os três números, tentar calcular a resposta mais rapidamente do que Laurinha consegue.

## Entrada

A primeira linha da entrada contém um número inteiro  $S$ , o valor da soma dos dígitos. A segunda e a terceira linhas contêm respectivamente os inteiros  $A$  e  $B$ .

## Saída

Seu programa deve produzir uma única linha, contendo um único inteiro, quantos números no intervalo dado têm a soma de dígitos indicada.

## Restrições

- $1 \leq S \leq 36$
- $1 \leq A \leq 10000$
- $1 \leq B \leq 10000$
- $A \leq B$

## Exemplos

<b>Exemplo de entrada 1</b> 3 10 30	<b>Exemplo de saída 1</b> 3
<b>Exemplo de entrada 2</b> 15 1 20	<b>Exemplo de saída 2</b> 0
<b>Exemplo de entrada 3</b> 1 1 10000	<b>Exemplo de saída 3</b> 5

# Lista palíndroma

Nome do arquivo: “lista.x”, onde x deve ser c, cpp, pas, java, js ou py

Uma palavra é chamada de palíndromo se a primeira letra da palavra é igual à última letra da palavra, a segunda letra é igual à penúltima letra, a terceira letra é igual à antepenúltima letra, e assim por diante. Por exemplo, as palavras *osso* e *sopapos* são palíndromos.

Nesta tarefa estamos interessados não em palavras, mas em listas de números inteiros. Nesse caso, vamos definir que uma lista é palíndroma se  $L[i] = L[N - i + 1]$ , onde  $L[i]$  representa o  $i$ -ésimo elemento da lista (note que nesta notação o índices variam de 1 a  $N$ ).

Você pode modificar uma lista usando a operação de *contração*, que é definida da seguinte forma: escolha dois elementos adjacentes da lista e substitua os dois elementos por um único elemento de valor igual à soma dos elementos substituídos. Note que ao efetuar uma operação de contração o número de elementos da lista decresce de um elemento.

Dada uma lista de números inteiros, você deve escrever um programa para determinar o menor número de operações de contração que devem ser realizadas de modo que a lista resultante seja palíndroma.

## Entrada

A primeira linha da entrada contém um inteiro  $N$ , o número de elementos da lista. A segunda linha contém  $N$  inteiros  $L_i$ , os elementos da lista.

## Saída

Seu programa deve produzir uma única linha, contendo um único inteiro, o menor número de operações de contração necessárias para tornar a lista palíndroma.

## Restrições

- $1 \leq N \leq 10^6$
- $1 \leq L_i \leq 10^9$ , para  $1 \leq i \leq N$

## Informações sobre a pontuação

- Para um conjunto de casos de testes valendo 30 pontos,  $N \leq 10$ .
- Para um conjunto de casos de testes valendo outros 30 pontos  $N \leq 10^3$ .
- Para um conjunto de casos de testes valendo outros 40 pontos, nenhuma restrição adicional.

## Exemplos

<b>Exemplo de entrada 1</b> 5 10 60 20 40 10	<b>Exemplo de saída 1</b> 1
<b>Exemplo de entrada 2</b> 5 999 1 999 1 999	<b>Exemplo de saída 2</b> 0

<b>Exemplo de entrada 3</b>	<b>Exemplo de saída 3</b>
4 10 40 30 20	2