



OBI2012

Caderno de Tarefas

Modalidade **Programação • Nível 1, Fase 2**

26 de maio de 2012

A PROVA TEM DURAÇÃO DE 4 HORAS

Promoção:



Sociedade Brasileira de Computação

Patrocínio:



Fundação Carlos Chagas

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.
- 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 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; e soluções na linguagem Python devem ser arquivos com sufixo *.py*. Para problemas diferentes você pode escolher trabalhar com linguagens diferentes, mas apenas uma solução, em uma única linguagem, deve ser submetida para cada problema.
- Ao final da prova, para cada solução que você queira submeter para correção, copie o arquivo fonte para o seu diretório de trabalho ou disquete, 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*, *print*, *write*
- Procure resolver o problema 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.

Carnaval

Nome do arquivo fonte: `carnaval.c`, `carnaval.cpp`, `carnaval.pas`, `carnaval.java`, ou `carnaval.py`

O Carnaval é um feriado celebrado normalmente em fevereiro; em muitas cidades brasileiras, a principal atração são os desfiles de escolas de samba. As várias agremiações desfilam ao som de seus sambas-enredos e são julgadas pela liga das escolas de samba para determinar a campeã do Carnaval.

Cada agremiação é **avaliada** em vários quesitos; em cada quesito, cada escola recebe **cinco notas** que variam de 5,0 a 10,0. A nota final da escola em um dado quesito é a soma das **três notas centrais** recebidas pela escola, **excluindo a maior e a menor** das cinco notas.

Como existem muitas escolas de samba e muitos quesitos, o presidente da liga pediu que você escrevesse um programa que, dadas as notas da agremiação, calcula a sua nota final num dado quesito.

Entrada

A entrada contém uma única linha, contendo cinco números N_i ($1 \leq i \leq 5$), todos com uma casa decimal, indicando as notas recebidas pela agremiação em um dos quesitos.

Saída

Seu programa deve imprimir uma única linha, contendo um único número com exatamente uma casa decimal, a nota final da escola de samba no quesito considerado.

Restrições

- $5.0 \leq N_i \leq 10.0$

Exemplos

Entrada 6.4 8.2 8.2 7.4 9.1	Saída 23.8
Entrada 10.0 10.0 5.0 5.0 10.0	Saída 25.0

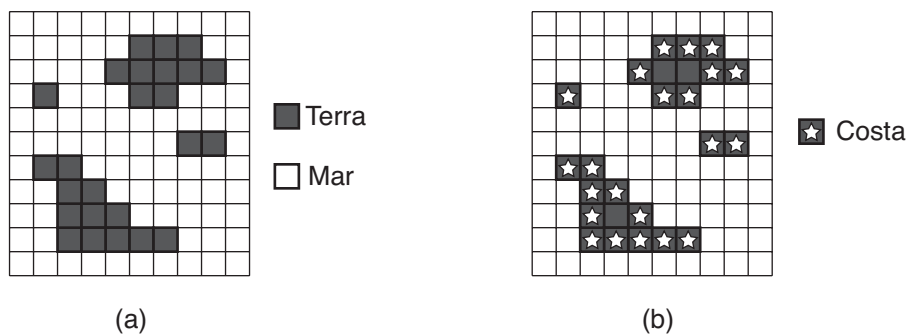
Costa

Nome do arquivo fonte: `costa.c`, `costa.cpp`, `costa.pas`, `costa.java`, ou `costa.py`

A Nlogônia é um país tropical, com muitas belezas naturais internacionalmente famosas; dentre elas, encontram-se as belas praias que compõem o arquipélago do país, que todo verão recebem milhões de turistas estrangeiros.

O Ministério do Turismo da Nlogônia está preparando o país para a chegada dos turistas, mas para fazer seu planejamento, precisa saber a extensão da costa nlogônica. Para isso, ele gerou um mapa que divide o território nacional em vários quadrados, que podem ser ocupados por água ou por terra; considera-se que um quadrado é parte da costa nlogônica se ele é um quadrado ocupado por terra que tem um lado em comum com um quadrado ocupado por água.

Na figura abaixo, (a) mostra um trecho do mapa gerado e (b) mostra os quadrados do trecho dado que são costa.



Como a Nlogônia é um país muito grande, o ministro do turismo pediu que você escrevesse um programa que, dado o mapa da Nlogônia, determina a extensão da costa nlogônica.

Entrada

A primeira linha da entrada contém dois inteiros M e N indicando, respectivamente, o número de linhas e o número de colunas do mapa. Cada uma das M linhas seguintes contém N caracteres: um caractere '.' indica que aquele quadrado do território é ocupada por água; um caractere '#' indica que aquele quadrado do território é ocupada por terra.

Considere que todo o espaço fora da área do mapa é ocupado por água.

Saída

Seu programa deve imprimir uma única linha contendo um único inteiro, indicando quantos quadrados do território fazem parte da costa da Nlogônia.

Restrições

- $1 \leq M, N \leq 1000$

Exemplos

Entrada	Saída
5 5#.. .###. ..#..	4

Entrada	Saída
10 10###..#####. .#...##...##. .##..... ..##..... ..###..... ..#####...	22

Guerra por território

Nome do arquivo fonte: `guerra.c`, `guerra.cpp`, `guerra.pas`, `guerra.java`, ou `guerra.py`

Tombólia do Oeste e Tombólia do Leste travaram uma guerra durante 50 anos. O motivo da guerra era o tamanho do território de cada país. Pelo bem da população dos dois países, os governos resolveram fazer um tratado para finalizar a guerra. O tratado consiste em fazer um **divisão justa**, e certamente contínua, do território. Eles resolveram pedir sua ajuda para calcular o ponto de **divisão do território**. Depois de tantos anos de guerra, os países não podem lhe pagar uma viagem para ver previamente o território que será dividido. Ao invés disso, eles prepararam uma **lista** a_1, a_2, \dots, a_N de inteiros que indicam o tamanho de cada seção do território. A seção a_1 é vizinha da seção a_2 que por sua vez é vizinha da seção a_3 e assim por diante. Os governos querem **uma divisão em uma seção k** de tal forma que $a_1 + a_2 + \dots + a_k = a_{k+1} + a_{k+2} + \dots + a_N$.

Sua tarefa é dada uma lista de inteiros positivos a_1, a_2, \dots, a_N , **determinar a seção k** tal que soma dos comprimentos das seções a_1 até a_k é igual a soma dos comprimentos das seções a_{k+1} até a_N .

Entrada

A primeira linha da entrada contém um inteiro N indicando o número de seções do território. A segunda linha da entrada contém N inteiros a_1, a_2, \dots, a_N separados por um único espaço que indicam os comprimentos das seções.

Saída

Seu programa deve imprimir uma única linha contendo um inteiro que indica a seção do território onde acontecerá a divisão.

Nota: é garantido que sempre existe uma divisão que satisfaz as condições dos países.

Informações sobre a pontuação

- Em um conjunto de casos de teste que totaliza 50 pontos, $N \leq 1000$.

Restrições

- $1 \leq N \leq 10^5$.
- $1 \leq a_i \leq 100$, para $i = 1, 2, \dots, N$.

Exemplos

Entrada	Saída
4 5 3 2 10	3

Entrada	Saída
9 2 8 2 8 4 4 4 4 4	4