

As moedas de Super Mario

Autor: Nikolas Sobral

Super Mario é uma das franquias de videogame mais populares e icônicas de todos os tempos. O jogo foi criado pela Nintendo em 1985 e apresenta o personagem Mario como o protagonista principal. O objetivo do jogo é controlá-lo enquanto corre e salta por uma série de níveis repletos de obstáculos, inimigos e moedas.



Sabendo disso, a sua tarefa é somar o valor das moedas coletadas pelo jogador que tenha valor maior ou igual ao mínimo informado pelo jogo.

Entrada

A primeira linha de entrada é composta por um inteiro **"N"** ($1 \leq N \leq 100$), representando o número de moedas coletadas. A seguir, em cada uma das próximas **"N"** linhas, será dado uma string **"C"** e um inteiro **"V"** ($1 \leq V \leq 100$), representando respectivamente, a cor da moeda e o seu valor. A última linha é composta de um inteiro **"E"** ($1 \leq E \leq 50$), representando o valor mínimo das moedas que devem ser somadas.

Saída

A saída será composta uma linha para cada moeda somada, na ordem dada na entrada, onde em cada linha você deverá imprimir a cor da moeda seguida de seu valor. Por fim, imprima na última linha, o valor total das moedas com valor maior ou igual a **"E"**. Caso não haja moedas com valor maior ou igual a **"E"**, imprima simplesmente **0 (zero)**.

Exemplos

Entrada	Saída
3 Azul 20 Vermelho 10 Laranja 15 15	Azul 20 Laranja 15 35
3 Verde 45 Azul 25 Vermelho 15 1	Verde 45 Azul 25 Vermelho 15 85
3 Laranja 100 Amarelo 15 Verde 25 50	Laranja 100 100

Faxina

Autor: Gabriel Dahia

Você decidiu se livrar dos livros em sua casa. Seu critério para decidir se um livro vai ser doado ou não é a quantidade de consoantes em seu título. Caso o título possua mais do que T consoantes, ele será doado; caso contrário, ele ficará em sua estante.

Sua tarefa é determinar, dados os títulos dos livros, quais serão doados.

Entrada

A primeira linha da entrada contém dois inteiros N e T , respectivamente o número de livros na estante e o número máximo de consoantes permitidas no título de cada livro. As próximas N linhas contêm os títulos dos livros, que tem no máximo 20 símbolos. Estes são apenas letras minúsculas e espaços.

Saída

Para cada livro, seu programa deve imprimir 0 caso o livro deva ser doado e 1 caso ele fique na estante, seguidos de um fim de linha.

Limites

- $1 \leq N \leq 10^5$;
- $1 \leq T \leq 20$.

Exemplos

Entrada	Saída
3 4	
harry potter	0
senhor dos aneis	0
aleph	1

Inventário caótico

Autor: Gustavo Amaral

Jônatas da Silva após várias horas jogando Dark Souls III percebeu que seu inventário está caótico e não consegue encontrar seus itens. **Seu objetivo é identificar se um item está no inventário.**



Entrada

A entrada cosistirá de várias linhas, onde em cada uma teremos **uma string representando o nome de determinado item. A entrada com o nome dos itens termina** quando for digitado **a palavra “fim”**. Por fim, **será dado o nome do item que Jônatas quer saber se está no inventário dele.**

Saída

Deverá ser impresso **“item encontrado”** caso o item esteja no inventário ou **“voce ainda nao descobriu este item”**, caso contrário.

Entrada	Saída
Storm Ruler Scholar's Candlestick Broadsword Anri's Straight Sword Sunlight Straight Sword Gael's Greatsword Ringed Knight Paired Greatswords fim Carthus Shotel	voce ainda nao descobriu este item
Frayed Blade Follower Sabre Sellsword Twinblades Harald Curved Greatsword Thrall Axe Butcher Knife fim Sellsword Twinblades	item encontrado

Organização de autores

Visconde de Sabugosa é um conhecido personagem do clássico Sítio do pica-pau amarelo. Por ser um assíduo leitor, muitas vezes ele precisa se lembrar dos nomes de muitos autores das clássicas obras literárias.

Desta forma, ajude o Visconde criando um programa que abrevie os nomes destes autores, fazendo com que ele se lembre mais facilmente.



Entrada

A primeira linha consiste em um inteiro '**N**' ($1 < N \leq 100$), representando o número de nomes de autores que o programa irá receber. Cada uma das próximas '**N**' linhas contém o nome completo (**caracteres minúsculos**) dos autores. Os nomes **não** terão preposições e nem caracteres especiais (maria **das** dores = maria dores; joão **de** jesus = joao jesus).

Saída

Para cada nome de autor recebido, o programa deve abreviar todos os nomes, com exceção do último. Cada nome abreviado deverá ser impresso com a primeira letra em maiúsculo, seguido de ponto e um espaço, e por fim, após as abreviações, imprima o último sobrenome sem abreviação.

Entrada	Saída
3 maria andrade ferreira felipe barros silva matos joao barbosa	M. A. Ferreira F. B. S. Matos J. Barbosa
4 rubisley lemes manoel mendonca filho rodrigo souza karl apaza aguero	R. Lemes M. M. Filho R. Souza K. A. Aguero

Fonte da maioria dos nomes: <https://gerador-nomes.>

Palíndromos

Autor: Ubiratan Neto

Luiz e seus colegas estão participando de uma série de jogos na aula de português em sua escola. Em um dos jogos, é dado uma palavra para cada um deles, e eles precisam dizer se ela é um palíndromo ou não. Uma palavra é um palíndromo se ela é a mesma palavra lida de trás pra frente. Por exemplo “arara”, “osso”, “ralar”, etc. Luiz deseja vencer o jogo mesmo que seja trapaceando, e pediu a você que fizesse um programa que respondesse se uma palavra era um palíndromo ou não.

Entrada

A primeira linha da entrada possui uma palavra S , contendo apenas letras minúsculas, a palavra dada por Luiz.

Saída

A saída deve conter numa única linha a palavra “Sim”, caso S seja um palíndromo, e “Nao” caso “S” não seja um palíndromo.

Limites

- $1 \leq S \leq 100$

Exemplos

Entrada	Saída
arara	Sim

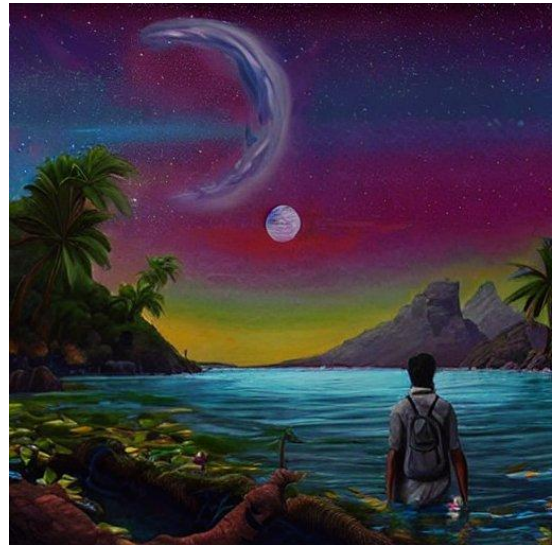
Entrada	Saída
reviver	Sim

Entrada	Saída
rezar	Nao

Pulando até o tesouro!

Autor: José Roberto Cardoso

Um explorador está em uma emocionante busca por um tesouro escondido em uma ilha misteriosa. Para chegar até o local exato do tesouro, ele precisa atravessar uma série de obstáculos ao longo do caminho. O explorador decidiu que a melhor forma de superar esses obstáculos é criando um mapa. O mapa será representado por um vetor de números inteiros, onde cada elemento representa a altura de um obstáculo no caminho. O explorador precisa escrever um programa que determine a quantidade de obstáculos que ele conseguirá pular, considerando que ele tem uma habilidade máxima de salto.



Escreva um programa que recebe como entrada um vetor de alturas dos obstáculos e a habilidade máxima de salto do explorador, retorne a quantidade máxima de obstáculos que ele conseguirá pular. Ressaltando que obstáculos com altura maior do que a habilidade máxima de salto do explorador não serão ultrapassados e nem mesmo os demais a partir dali.

Entrada

A entrada contém na primeira linha um inteiro “N” ($1 \leq N \leq 100$), representando o número de obstáculos no caminho do explorador. Na próxima linha serão dados “N” inteiros “A” ($1 \leq A \leq 10$), representando as alturas dos “N” obstáculos. Por fim, na última linha teremos um inteiro “S” ($1 \leq S \leq 15$), representando a altura máxima do salto do explorador.

Saída

Imprima na primeira linha a quantidade de obstáculos que o explorador conseguirá pular durante a sua aventura na ilha misteriosa. Na segunda linha imprima “1” se o explorador venceu todos os obstáculos, e “0” (zero), caso contrário.

Entrada	Saída
8 2 3 1 2 4 2 1 3 3	4 0
4 4 2 1 2 4	4 1

Vamos jogar um jogo

Autor: Danilo de A. Peleteiro



E agora?!?! Você foi capturado por Jigsaw em mais um de seus planos contra aqueles que ele julga não valorizar a própria vida! Para provar que Jigsaw está enganado, você deverá resolver um de seus peculiares enigmas e assim garantir a sua liberdade (talvez). Você encontrou uma gravação que explica passo a passo o que deverá ser feito. Em sua sala, haverá um papel com uma frase acompanhada de um número e uma palavra. O que Jigsaw deseja é muito simples: **que você diga se a quantidade de ocorrências daquela palavra na frase é a mesma da escrita no papel**. Caso seja, você deverá dizer “**SIM!**”, do contrário, deverá falar “**NAO!**”. Um detalhe essencial é que **todas as letras são minúsculas** e **Jigsaw ignora espaços em branco na frase no momento de contar as ocorrências da palavra**.

Tarefa

Para sua sorte, você encontrou um computador velho na sala onde está, e como é conhecido por ser viciado em programar, decidiu desenvolver um programa que o auxiliasse (e, quem sabe, outros futuros prisioneiros) nesse enigma. Portanto, você deverá computar a frase solicitada por Jigsaw e posteriormente avaliar se existe a quantidade ‘**Q**’ de ocorrências de uma dada palavra **P**.

Entrada

A primeira linha da entrada consiste de uma string **S**, que indica a frase a ser avaliada. A segunda linha contém um inteiro ‘**Q**’ ($1 \leq Q \leq 30$), informando a quantidade de ocorrências, seguido de uma palavra **P**, que indica o que deve ser detectado na frase **S**.

Saída

Seu programa deverá imprimir a quantidade de ocorrências de **P** em uma linha. Na outra, deverá imprimir “**SIM!**” caso essa quantidade seja igual à ‘**Q**’ e, caso contrário, deverá imprimir “**NAO!**”.

Entrada	Saída
eu quero jogar um jogo jogando limpo 3 jog	3 SIM!
xhuisyd xnzyxe nxnzzz zx x ify zzuzzzz z zjx 4 zz	6 NAO!
eu adoro sao joao e eu amo suas comidas 3 ua	3 SIM!

AMONG US

Among us é um jogo que foi lançado em 2018 e tem ganhado bastante destaque na internet durante a quarentena. Apesar de à primeira vista não ser um jogo que chame muita atenção, existem vários fatores que fazem a experiência de jogar ser muito divertida.

Em Among Us, os jogadores são astronautas que devem realizar uma série de atividades em uma nave, desde arrumar a fiação até jogar o lixo fora. Só que entre eles há um impostor, que tem o objetivo de sabotar essas atividades e matar os outros tripulantes. Essa é a dinâmica que serve como palco quando o ápice de cada partida acontece de verdade: as discussões em grupo para saber quem é o impostor. Nelas os jogadores tripulantes podem exercitar suas habilidades investigativas, enquanto o jogador impostor sua habilidade de dissimulação, e o melhor, desfazer amizades.



Agora chegou a grande hora, para aumentar ainda mais a competitividade do jogo um rank de melhor impostor será lançado, e será baseado em quantidade de assassinatos durante as partidas desde a criação da conta no jogo até agora!

Você como programador foi contratado para fazer um código que receba a informação de quantos jogadores possuem uma conta no jogo, a quantidade de assassinatos de cada um, e imprima de forma crescente a quantidade de assassinatos de cada jogador.

Entrada

A entrada consiste de duas linhas. A primeira contém um inteiro '**N**' ($1 \leq N \leq 100000$), representando a quantidade de jogadores no jogo, e a segunda contém '**N**' inteiros **X1, X2... XN** ($0 \leq X_i \leq 1000000 \mid i = 1, 2, \dots, N$), representando a quantidade de assassinatos de cada um dos '**N**' jogadores.

Saída

A saída consiste de '**N**' inteiros separados por espaço, representando a quantidade de assassinatos cometidos pelos jogadores e apresentados de forma crescente.

Exemplo de entrada	Exemplo de saída
8 29 84 26 1 398 42 4 0	0 1 4 26 29 42 84 398
3 1000000 8 1	1 8 1000000

Obs1: Exige-se uma solução que não use ordenação, apenas vetores e o que foi aprendido na disciplina até o momento.

Obs2: É garantido que não há jogadores com a mesma quantidade de assassinatos na entrada do programa.