

Questão 1 - Primeiro reverso

Problema

Faça com que a função pegue o parâmetro `str` que está sendo passado e retorne a string na ordem inversa.

Entrada

O programa receberá a variável **texto** do tipo string. A variável irá conter uma sentença comum.

Saída

O retorno deverá ser do tipo string e conter o valor passado na entrada de forma inversa.

Exemplos

Entrada	Saída
Hello World and Coders	sredoC dna dlroW olleH
O Hexa Vem!	!meV axeH O

Questão 2 - Vogal ou Consoante

Problema

Faça com que a função pegue o parâmetro **letra** que está sendo passado e retorne “Vogal” ou “Consoante”.

Entrada

O programa receberá a variável **letra** (string). A variável irá conter uma letra qualquer do alfabeto.

Saída

O retorno deverá ser do tipo string e apresentar se a letra digitada é uma vogal retornando “Vogal” ou uma consoante, retornando “Consoante”.

Exemplos

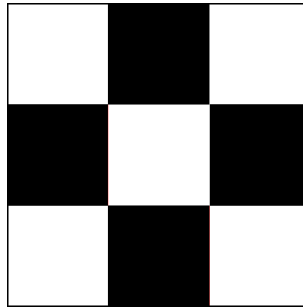
Entrada	Saída
a	Vogal
b	Consoante

Questão 3 - Xadrez

Problema

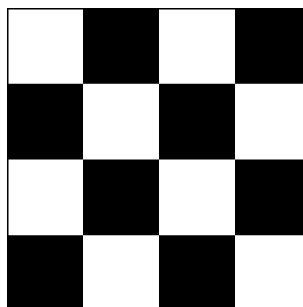
Quantas casas brancas e quantas casas pretas contém um tabuleiro de xadrez $N \times N$?

Tabuleiro 3x3:



5 casas brancas e 4 casas pretas

Tabuleiro 4x4:



8 casas brancas e 8 casas pretas

Entrada

O programa receberá N (*int*) ($2 \leq N \leq 100$)

Saída

O programa deverá mostrar o número de casas brancas e o número de casas pretas do tabuleiro. (*string*)

Exemplos

Entrada	Saída
3	5 casas brancas e 4 casas pretas
4	8 casas brancas e 8 casas pretas

Questão 4 - Lista de Compras

Problema

Valentina é uma mulher muito dedicada e costuma trabalhar até tarde todos os dias. Para economizar tempo, ela faz a lista de compras do mercado em um aplicativo e costuma anotar cada item na mesma hora que percebe a falta dele em casa.

O problema é que o aplicativo não exclui itens duplicados e como Valentina é distraída, anota o mesmo item mais de uma vez e a lista acaba ficando extensa. Sua tarefa como programadora e amiga de Valentina é melhorar o aplicativo de notas desenvolvendo um código que exclui os itens duplicados da lista de compras e que os ordena alfabeticamente.

Entrada

Cada lista de compra consiste de uma única linha que contém de 1 a 1000 itens ou palavras compostas apenas de letras minúsculas (de 1 a 20 letras), sem acentos e separadas por um espaço em ordem alfabética.

Saída

A saída contém uma linha com as listas de compras de Valentina, sem itens repetidos e em ordem alfabética.

Exemplos

Entrada	Saída
carne laranja picles suco laranja picles	carne laranja suco picles
laranja pera laranja pera pera	laranja pera

Questão 5 - Número Solitário

Problema

Será dado a você um vetor com **N** números, onde todos estarão em pares. Porém um desses números acabou ficando sem par, você consegue identificar qual é esse número ?

Por exemplo, **A** = {1, 1, 3, 3, 5, 5, 5}, o número que ficou sozinho foi o 5

Entrada

Cada teste contém um vetor com números inteiros entre 1 e 10^5 .

Saída

Para cada caso de teste imprima apenas os números que ficaram sozinhos.

Exemplos

Entrada	Saída
[1, 3, 4, 3, 1]	[4]
[1, 1, 1]	[1]
[2, 1, 1, 3]	[2, 3]

Questão 6 - Diamantes e Areia

Problema

John está trabalhando numa mina de diamantes. Seu trabalho é extrair a maior quantidade de diamantes possível "<>". Ele precisa retirar todas as partículas de areia "." que ele achar, neste processo o diamantes pode ser extraído e assim novos diamantes serão formados. Se ele obter uma entrada: "<... << .. >>> >>>". 3 diamantes serão formados. O primeiro vem de "<..>", resultando " <... <>> >>>". O segundo diamante então é removido "<.....> >>>". O terceiro diamante então é removido e deixado no final " >>>", sem a possibilidade de extrair mais diamantes.

Entrada

O programa receberá **line_to_extract** (*string*), uma linha de até 1000 caracteres que contém caracteres como "<", ">" e ".".

Saída

O programa deverá mostrar o número de diamantes extraídos.

Exemplos

Entrada	Saída
<..><..>>	3
<<<..<.....<<<<....>	1

Questão 7 - Triângulo de Pascal

Problema

O triângulo de Pascal (alguns países, nomeadamente em França, é conhecido como Triângulo de Tartaglia) é um triângulo numérico infinito formado por números binomiais Binomial, onde n representa o número da linha e k representa o número da coluna, iniciando a contagem a partir do zero. O triângulo foi descoberto pelo matemático chinês Yang Hui, e 500 anos depois várias de suas propriedades foram estudadas pelo francês Blaise Pascal. Cada número do triângulo de Pascal é igual à soma do número imediatamente acima e do antecessor do número de cima.

David, o fera do seu time de programação competitiva, descobriu que a soma da i -ésima linha de um triângulo de pascal é 2^i . Ele quer agora descobrir a soma do triângulo inteiro, de N linhas. Mas como ele achou que este problema era muito trivial para merecer a atenção dele, ele decidiu tentar resolver um problema sobre grafos bipartidos (um tópico muito mais difícil) e assim, sobrou para você encontrar a solução deste problema.

Entrada

As linhas contêm um inteiro N ($1 \leq N \leq 31$), o número de linhas do Triângulo de Pascal.

Saída

Para cada caso de teste, a saída deve conter uma linha com um inteiro S , a soma do triângulo de pascal de N linhas.

Exemplos

Entrada	Saída
1	1
2	3
5	31
31	2147483647

Questão 8 - Compondo Jingles








Problema

Marcos está dando seus primeiros passos para aprender composição de Jingles. Mas ele está tendo muita dificuldade para obter resultados, mas pelo menos ele está conseguindo produzir melodias atraentes.

Na música, cada nota tem seu tom (sua frequência, resultando o quão alto ou baixo é seu som) e sua duração (quão longa a nota deve ser). Neste problema estamos interessados apenas na duração das notas.

Um jingle é dividido numa sequência de compassos, e o compasso é formado por uma série de notas.

A duração da nota é dada pelo seu formato. Neste problema nós vamos utilizar letras em maiúsculo para indicar a duração das notas. A lista a seguir contém o valor de cada nota disponível:

Notes							
Identifier	W	H	Q	E	S	T	X
Duration	1	1/2	1/4	1/8	1/16	1/32	1/64

A duração do compasso é a soma das durações de notas do mesmo. No Jingle de Marcos, cada compasso tem a mesma duração. Como Marcos ainda é um iniciante, o seu famoso professor Johann Sebastian III ensinou a ele que a duração de todo compasso deve ser 1.

Por exemplo, Marcos escreveu uma composição de 5 compassos, na qual os 4 primeiros contém a duração correta e o último contém a duração incorreta. No exemplo abaixo, cada compasso está separado por uma barra, e cada nota está sendo representada por uma letra da tabela acima.

/HH/QQQQ/XXXTXTEQH/W/HW/

Marcos gosta de programação tanto quanto música. Ele quer escrever um programa que determina, para cada uma de suas composições, quais compassos têm a duração correta.

Entrada

O programa receberá uma string que poderá conter entre 3 a 200 caracteres representando a composição de Marcos. A composição começa e termina com uma barra “/”. Cada compasso é dividido dentro de uma composição também por uma barra “/”. Cada nota é representada por uma letra em maiúsculo da tabela apresentada acima. Você deve assumir que cada composição tem pelo menos 1 nota. Todos os caracteres da entrada serão “/” ou uma das representações de notas da tabela.

Saída

O programa deverá mostrar o número de compassos correto da composição.

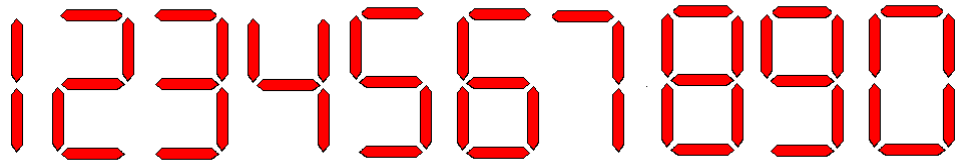
Exemplos

Entrada	Saída
/HH/QQQQ/XXTXTEQH/W/HW/	4
/W/W/SQHES/	3
/WE/TEX/THES/	0

Questão 9 - Leds

Problema

John quer montar um painel contendo diversos números com LEDs. Ele não tem muitos leds e não sabe se tem a quantidade necessária para montar seu painel. Considerando a imagem abaixo, faça um algoritmo para ajudar John a descobrir quantos leds será necessário para ele montar seu painel.



Entrada

O programa receberá uma string entre $(1 \leq V \leq 10100)$ correspondente ao número que John deseja montar no painel.

Saída

O programa deverá retornar o número necessário de leds para representar aquela string no painel seguido da palavra 'leds'.

Exemplos

Entrada	Saída
1	2 leds
115380	27 leds
2819311	29 leds

Questão 10 - Contador de Dígitos

Problema

Diana está trabalhando para escrever uma lista com todos números inteiros entre **A** e **B**. Ela deseja saber a contagem de cada dígito entre 1 e 9 que irá ser utilizado nesta contagem.

Entrada

O programa deve receber 2 dois números inteiros em forma de string.

Saída

O programa deverá apresentar a contagem de cada dígito entre 1 e 9 da contagem dada entre **A** e **B** em uma única linha. (*string*)

Exemplos

Entrada	Saída
1 9	0 1 1 1 1 1 1 1 1
12 321	61 169 163 83 61 61 61 61 61 61
12345678 12345679	0 2 2 2 2 2 2 2 1 1