

Prova-03-REC

Prof. Msc. Elias Batista Ferreira
Prof. Dr. Gustavo Teodoro Laureano
Profa. Dra. Luciana Berretta
Prof. Dr. Thierson Rosa Couto

Sumário

1	Aliteração	2
2	Pares e Ímpares (+++)	3
3	Frequência de <i>Strings</i> (++++)	4

1 Aliteração



(+++)

Uma aliteração ocorre quando duas ou mais palavras consecutivas de um texto possuem a mesma letra inicial (ignorando maiúsculas e minúsculas). Sua tarefa é desenvolver um programa que identifique, a partir de uma sequência de palavras, o número de aliterações que essa sequência possui.

Entrada

A entrada contém diversos casos de testes. Cada caso é expresso como um texto em uma única linha, contendo de 1 a 100 palavras separadas por um único espaço, cada palavra tendo de 1 a 50 letras minúsculas ou maiúsculas ('A'-'Z', 'a'-'z'). A entrada termina em EOF.

Saída

Para cada caso de teste imprima o número de aliterações existentes no texto informado, conforme exemplos abaixo.

Exemplo

Entrada
He has four fanatic fantastic fans
There may be no alliteration in a sequence
Round the rugged rock the ragged rascal ran
area artic Soul Silly subway ant artic none
Saída
2
0
2
3

2 Pares e Ímpares (+++)



(+++)

Considerando a entrada de valores inteiros não negativos, ordene estes valores segundo o seguinte critério: • Primeiro os Pares • Depois os Ímpares Sendo que deverão ser apresentados os pares em ordem crescente e depois os ímpares em ordem decrescente.

Entrada

A primeira linha de entrada contém um único inteiro positivo $N(1 < N < 10^5)$ que corresponde ao número de linhas de entrada que vêm logo a seguir. As próximas N linhas conterão, cada uma delas, um valor inteiro não negativo.

Saída

Apresente todos os valores lidos na entrada segundo a regra descrita acima. Cada número deve ser impresso em uma linha. Após imprimir último número quebre a linha.

Exemplo

Entrada
10
4
32
34
543
3456
654
567
87
6789
98
Saída
4
32
34
98
654
3456
6789
567
543
87

3 Frequência de *Strings* (++++)



(++++)

Escreva um programa que leia um texto e calcule a quantidade de vezes que cada *string* aparece no texto. Uma *string* é uma sequência de caracteres delimitada pelos seguintes símbolos " . , ! ? () [] { } ".

Entrada

Seu programa deve ler uma *string* que terá no máximo 2048 caracteres.

Saída

Para cada *string* presente no texto de entrada, seu programa apresentar uma linha contendo a *string* avaliada e quantidade de vezes que ela aparece no texto. A *string* deve ser apresentada entre parênteses e sua frequência é um número inteiro.

Exemplo

Entrada	Saída
tal pai, tal o filho	(tal)2 (pai)1 (tal)2 (o)1 (filho)1

Entrada	Saída
1 2 3 4 5 5 1 2	(1)2 (2)2 (3)1 (4)1 (5)2 (1)2 (2)2

Entrada	Saída
Ola mundo! Eu gosto de programacao.	(Ola)1 (mundo)1 (Eu)1 (gosto)1 (de)1 (programacao)1