

Centro de Educação Superior a Distância do
Estado do Rio de Janeiro – CEDERJ

Curso de Tecnologia em Sistemas de Computação – TSC

EAD-05.009 Fundamentos de Programação

Caderno de Exercícios

Aula 11

(Estrutura de Dados: Dicionário)

Professores

Dante Corbucci Filho
Leandro A. F. Fernandes

Instruções

- Utilize Python 3 e a IDE PyCharm na elaboração de soluções para os problemas propostos;
- A entrada de cada problema deve ser lida da entrada padrão (teclado);
- A saída de cada problema deve ser escrita na saída padrão (tela);
- Siga o formato apresentado na descrição da saída, caso contrário não é garantido que a saída emitida será considerada correta;
- Na saída, toda linha deve terminar com o caractere `'\\n'` ;
- Utilize o URI Online Judge (<http://www.urionlinejudge.com.br>) e submeta sua solução para correção automática.

Referências Autorais

Os exercícios apresentados nesta lista foram extraídos do URI Online Judge (<http://www.urionlinejudge.com.br>). Acesse a URL apresentada abaixo do título de cada problema para proceder com a correção automática de sua solução e, também, para consultar a autoria do enunciado.

Problema A: Ida à Feira

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1281>

Dona Parcinova costuma ir regularmente à feira para comprar frutas e legumes. Ela pediu então à sua filha, Mangojata, que a ajudasse com as contas e que fizesse um programa que calculasse o valor que precisa levar para poder comprar tudo que está em sua lista de compras, considerando a quantidade de cada tipo de fruta ou legume e os preços destes itens.



Entrada

A primeira linha de entrada contém um inteiro **N** que indica a quantidade de idas à feira de dona Parcinova (que nada mais é do que o número de casos de teste que vem a seguir). Cada caso de teste inicia com um inteiro **M** que indica a quantidade de produtos que estão disponíveis para venda na feira. Seguem os **M** produtos com seus preços respectivos por unidade ou Kg. A próxima linha de entrada contém um inteiro **P** ($1 \leq P \leq M$) que indica a quantidade de diferentes produtos que dona Parcinova deseja comprar. Seguem **P** linhas contendo cada uma delas um texto (com até 50 caracteres) e um valor inteiro, que indicam respectivamente o nome de cada produto e a quantidade deste produto.

Saída

Para cada caso de teste, imprima o valor que será gasto por dona Parcinova no seguinte formato: R\$ seguido de um espaço e seguido do valor, com 2 casas decimais, conforme o exemplo abaixo.

Exemplo

Entrada	Saída
2	R\$ 15.37
4	R\$ 15.73
mamao 2.19	
cebola 3.10	
tomate 2.80	
uva 2.73	
3	
mamao 2	
tomate 1	
uva 3	
5	
morango 6.70	
repolho 1.12	
brocolis 1.71	
tomate 2.80	
cebola 2.81	
4	
brocolis 2	
tomate 1	
cebola 1	
morango 1	

Problema B: Composição de Jingles








<https://www.urionlinejudge.com.br/judge/pt/problems/view/1430>

A. C. Marcos está dando os primeiros passos para ser um compositor de jingles. Ele está tendo alguns problemas, mas ao menos ele está criando melodias agradáveis e ritmos atrativos.

Na música, uma nota tem um tom (sua frequência, resultando em quão grave ou agudo é o som) e uma duração (por quanto tempo a nota soa). Neste problema, estamos interessados apenas na duração das notas.

Um jingle é dividido em uma sequência de compassos, e um compasso é formado de uma série de notas.

A duração de uma nota é indicada pela sua forma. Neste problema, iremos utilizar letras maiúsculas para indicar a duração de uma nota. A seguinte tabela lista todas as notas disponíveis:

Notas							
Identificador	W	H	Q	E	S	T	X
Duração	1	1/2	1/4	1/8	1/16	1/32	1/64

A duração de um compasso é a soma da duração de suas notas. Nos jingles de Marcos, cada compasso tem a mesma duração. Como Marcos é apenas um iniciante, seu famoso professor Johann Sebastian III o ensinou que a duração de um compasso deve ser sempre 1.

Por exemplo, Marcos escreveu uma composição contendo cinco compassos, dentre os quais quatro possuem a duração correta e um está errado. No exemplo abaixo, cada compasso é delimitado com barras e cada nota é representada como na tabela acima.

`/HH/QQQQ/XXXTXTEQH/W/HW/`

Marcos gosta de computadores assim como de música. Ele quer que você escreva um programa que determine, para cada uma de suas composições, quantos compassos possuem a duração correta.

Entrada

A entrada contém vários casos de teste. Cada caso de teste é descrito em uma única linha contendo uma string cujo tamanho está entre 3 e 200 caracteres, inclusive, representando uma composição. Uma composição começa e termina com uma barra '/'. Compassos em uma composição são separados por uma barra '/'. Cada nota em um compasso é representada pela letra correspondente segundo a descrição acima. Você pode assumir que cada composição contém ao menos um compasso e que cada

compasso contém ao menos uma nota. Todos os caracteres na entrada serão barras ou uma das sete letras maiúsculas usadas para representar as notas.

O último caso de teste é seguido por uma linha contendo um único asterisco.

Saída

Para cada caso de teste, seu programa deve imprimir uma única linha contendo um único inteiro, o número de compassos que possuem a duração correta.

Exemplo

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

Problema C: Morse

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2338>

Todos os seus amigos estão loucos pelo filme "Interstellar". Agora eles estão postando textos no Facebook em Código Morse. Você precisa escrever um código para decodificar Morse!

Este Código Morse contém somente letras minúsculas (a-z) e espaços em branco.

Nota - Código Morse é uma forma de codificar caracteres em uma sequência de pontos e traços. Cada caracter tem uma codificação fixa. Um exemplo bem famoso é o Morse de SOS "...---..." . S é codificado como "..." (ponto ponto ponto) e O é codificado como "---" (traço traço traço). Neste problema, um ponto é denotado por "." e um traço por "-". Símbolos são separados por ".", letras são separadas por "..." e palavras são separadas por ".....". Sendo assim, SOS é codificado como =.=...==.=.=.=...=.=. Codificações para todos os caracteres podem ser encontradas no link abaixo.

<http://morsecode.scphillips.com/morse2.html>

Entrada

A entrada consiste em múltiplos casos de teste.

A primeira linha de cada entrada contém o número de casos de teste (t)($1 \leq t \leq 10$).

As próximas t linhas contém os Códigos Morse. O tamanho de cada código é menor que 1000 caracteres.

Saída

A saída é dividida em linhas, que representam a mensagem decodificada de cada caso de teste.

Exemplo

Entrada	Saída
2 ==.=.=.=...==.=.=.=. =.=...==.=.=.=...=.=.	gg sos

Problema D: Etiquetas de Noel

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2482>

Como de costume, neste ano Noel recebeu muitos pedidos de presentes. Só que em função de alguns imprevistos, não terá como entregar todos os presentes pessoalmente neste ano. Daí então decidiu utilizar o velho e bom correio tradicional, para alguns pedidos que podem ser entregues por carta.

Para esta tarefa, pediu ajuda ao elfo Evergreen Xadada, para que ele imprimisse etiquetas a todos os envelopes que serão destinados a algumas destas crianças, cujo pedido pode ser entregue por carta. Cada uma destas etiquetas deverá conter apenas o nome da criança e a saudação "Feliz Natal" no respectivo idioma desta criança. Para auxiliar nesta tarefa, Noel disponibilizou uma tabela com vários idiomas e o nome e o país de cada uma das crianças selecionadas, de acordo com o exemplo abaixo. Você deve ajudar Evergreen fazendo um programa que imprima estas etiquetas.



Entrada

A entrada é composta por um único caso de teste. A primeira linha de entrada contém um inteiro N ($1 < N < 100$) que indica a quantidade de traduções da palavra "Feliz Natal" existentes na entrada. As próximas $N * 2$ linhas contém respectivamente o nome de uma língua seguido da tradução de "Feliz Natal" para esta língua. Segue um inteiro M ($1 < M < 100$) que indica a quantidade de crianças que receberão as cartas. As próximas $M * 2$ linhas conterão, respectivamente, o nome da criança e a língua nativa desta criança.

Obs.: É garantido que nenhuma tradução apareça repetida ou duplicada e os países de todas as crianças estejam presentes na relação dos países.

Saída

Seu programa deverá imprimir todas as etiquetas de acordo com a entrada, conforme o exemplo abaixo, sempre com uma linha em branco após a impressão de cada uma das etiquetas, inclusive após a última etiqueta.

Exemplo

Entrada	Saída
2 frances Joyeux Noel ingles Merry Christmas 1 John Brolargo ingles	John Brolargo Merry Christmas

Entrada	Saída
7 frances Joyeux Noel ingles Merry Christmas alemao Frohe Weihnachten espanhol Feliz Navidad italiano buon natale polones Wesotych Swiat portugues Feliz Natal 5 Joao Paulo Silva portugues Pedro Guerra portugues Pietro Gonsalez espanhol Karol Kosinski polones John Smith ingles	Joao Paulo Silva Feliz Natal Pedro Guerra Feliz Natal Pietro Gonsalez Feliz Navidade Karol Kosinski Wesotych Swiat John Smith Merry Christmas

Problema E: Papagaio Poliglota

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2850>

Humberto tem um papagaio muito esperto. Quando está com as duas pernas no chão, o papagaio fala em português. Quando levanta a perna esquerda, fala em inglês. Por fim, quando levanta a direita fala em francês. Nico, amigo de Humberto, ficou fascinado com o animal. Em sua emoção perguntou: “E quando ele levanta as duas?”. Antes que Humberto pudesse responder, o papagaio gritou: “Ai eu caio, idiota!”.

Entrada

A entrada consiste de diversos casos de teste. Cada caso de teste consiste uma string informando qual a situação de levantamento de pernas do papagaio.

Saída

Para cada condição de levantamento de pernas do papagaio, imprima a linguagem que ele utilizará. Caso ele levante as duas pernas, imprima “caiu”. Quebre uma linha a cada caso de teste.

Exemplo

Entrada	Saída
esquerda direita nenhuma as duas	ingles frances portugues caiu

Problema F: Código Secreto

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2727>

Joana gosta de brincar de fingir ser uma agente secreta com suas amigas Bruna, Jaqueline e Laura. Joana e Bruna criaram um código secreto para se comunicar sem que suas inimigas descubram seus planos. O código secreto funciona da seguinte forma:

- A letra 'a' é representada por um único ponto '.'
- A letra 'b' é representada por dois pontos '..'
- A letra 'c' é representada por três pontos '...'
- As demais letras seguem a lógica anterior, porém cada conjunto de pontos está separado por um espaço e sempre com um conjunto a mais de pontos, como no exemplo abaixo:

```
. → a
.. → b
... → c
. . → d
. . . → e
. . . . → f
. . . . . → g
. . . . . . → h
. . . . . . . → i
```

O seu objetivo é criar um programa que decifre as mensagens secretas e ajudar Jaqueline e Laura descobrirem o que Joana e Bruna estão planejando.

Entrada

A entrada contém vários casos de teste. A primeira linha de cada teste deverá conter um inteiro ($N \leq 50$), que representa a quantidade de letras a serem decifradas e as N linhas seguintes contêm o código de cada letra.

Saída

Uma string representando a letra do alfabeto correspondente ao código de entrada. Cada string deve estar separada da outra por uma nova linha.

Exemplo

Entrada	Saída
2	o
... ..	i
... ..	o
3	l
... ..	a
... ..	
.	