



Universidade de Brasília

Departamento de Ciência da Computação

Introdução à Ciência da Computação - 113913

Lista de Exercícios 8

Dicionários

Observações:

- As listas de exercícios serão corrigidas por um **corretor automático**, portanto é necessário que as entradas e saídas do seu programa estejam conforme o padrão especificado em cada questão (exemplo de entrada e saída). Por exemplo, a não ser que seja requisitado na questão, não use mensagens escritas durante o desenvolvimento do seu código como “Informe a primeira entrada”. Estas mensagens não são tratadas pelo corretor, portanto a correção irá resultar em **resposta errada**, mesmo que seu código esteja correto;
- As questões estão em **ordem de dificuldade**. Cada lista possui 7 exercícios, sendo 1 questão fácil, 3 a 4 médias e 2 a 3 difíceis;
- Leia com atenção e faça **exatamente** o que está sendo pedido.

Antônimos

Antônima é uma jovem que tem um disfunção cerebral sem cura. Sempre que ela tenta falar um nome (substantivo ou adjetivo), ela fala o antônimo. Por exemplo, se ela quisesse elogiar a Ana com “Você é linda”, ela diria “Você é horrível”.

Seus amigos, cansados de tentar entender se ela estava os ofendendo ou demonstrando carinho, procuraram um programador para resolver o problema deles.

Não demorou muito, e encontraram você. Seu trabalho é criar um programa que, fornecido um dicionário de antônimos e uma frase de Antônima, deve decifrar o que ela quis dizer.

Entrada

A primeira linha da entrada consiste de um inteiro N, o número de entradas do dicionário fornecido.

As próximas N linhas contêm, cada uma, duas strings O e A, a palavra de Origem e seu Antônimo, respectivamente.

A última linha da entrada, contém, enfim, a frase de Antônima.

Considere que pontuação sempre estará separada por espaços

Saída

Seu programa deve imprimir uma única linha, a frase “traduzida”.

Exemplos

Entrada	Saída
2 feio bonito grande pequeno você é grande e muito feio	você é pequeno e muito bonito

Entrada	Saída
3 cheiroso fedido competente incompetente comprometido relaxado eu acho que você é um grande de um competente comprometido , seu cheiroso de meia!	eu acho que você é um grande de um incompetente relaxado , seu fedido de meia !

Borderline

O Transtorno de Personalidade Limítrofe (Borderline Personality Disorder, em inglês), é um transtorno de personalidade que se caracteriza pela abrupta mudança de comportamento, impulsividade e instabilidade de afetos, e é muitas vezes confundida com bipolaridade.

Joana tem uma variante de borderline tão rara que só existe em questões de listas de exercícios. Ela tem um conjunto de palavras críticas que desencadeiam uma alteração entre seus arquétipos de personalidades.

Tristes por usar essas palavras sem querer durante conversas do dia-a-dia e ver sua amiga explodindo com eles, os amigos de Joana pediram a você que escrevesse um programa que ajudasse-os a melhor controlar a situação.

Entrada

A primeira linha da entrada consiste de um inteiro N, o número de palavras críticas para Joana. As próximas N linhas contém, cada uma, duas strings T e P separadas por espaço, a palavra crítica e a personalidade que ela desencadeia, respectivamente.

A última linha da entrada consiste de uma string, a frase que os amigos de Joana querem falar. Considere que os sinais de pontuação estarão sempre isolados com espaços

Saída

Seu programa deve imprimir uma única linha contendo “Tudo bem!” caso a frase não contenha nenhuma palavra crítica, ou os nomes das personalidades que serão desencadeadas, separadas por espaço, na ordem de input.

Exemplos

Entrada	Saída
3 boldo explosiva tampa chorosa beijo calorosa Você pôs a tampa no boldo ?	chorosa explosiva

Entrada	Saída
2 dedos animada tinta criativa Quantos irmãos você tem , mesmo ?	Tudo bem!

Carros

A1A é um lava-jato na região de Albuquerque que recebe um grande volume de carros todos os dias. Seu dono, Bogdan Wolynetz, tem tido problemas recentes com alguns de seus funcionários e precisa de alguma forma de gerenciar os carros sujos e necessitados de cera de seus clientes.

Para isso, ele decidiu contratar um desenvolvedor python, você, para criar um sistema de gerenciamento sólido para seu estabelecimento.

Entrada

Cada linha da entrada começa com uma letra 'E', 'R' ou 'F', indicando o tipo de transação a ser realizada.

Se a transação for do tipo 'E' (entrar), o resto da linha conterá o primeiro nome do cliente (sem espaços), e qual o tipo de serviço a ser realizado ('cera' ou 'lavagem').

Caso a transação for 'R' (retirar), o resto da linha conterá somente o primeiro nome do cliente a ter o carro retirado.

Por fim, a última linha da entrada será sempre do tipo 'F', indicando o fechamento do lava-jato.

Saída

Para cada transação do tipo 'R', seu programa deve imprimir qual o último serviço que foi realizado no carro em questão. Caso não houver nenhum carro cadastrado sob o nome requisitado, imprimir 'Usuário não cadastrado'.

Exemplos

Entrada	Saída
E Walter cera R Heisenberg R Walter F	Usuário não cadastrado cera

Entrada	Saída
R Roberto R André E André lavagem E Rafael cera E André cera R André R André F	Usuário não cadastrado Usuário não cadastrado cera cera

Depita

Reginalda quer abrir uma startup revolucionária no Vale do Silício, a Depita, que irá alugar depósitos de marmitas. É uma tecnologia revolucionária: quando você chega para trabalhar, pela manhã, deposita seu almoço em uma Depita™, e saca só mais tarde, quando for comer.

Porém, Reginalda não tem muita experiência desenvolvendo, então decidiu terceirizar a parte de software da sua nova empresa. Você, sabendo da incrível oportunidade de poder revolucionar a indústria, logo se prontificou para resolver o problema de Reginalda pela modesta quantia de 3 mil dólares.

Entrada

A primeira linha da entrada consiste de um inteiro N , o número de iterações subsequentes. As próximas N linhas contêm, cada uma, uma string sem espaços S , o nome do depositante, e uma string que pode conter espaços D , a descrição da marmita depositada.

Se o mesmo depositante realizar o depósito múltiplas vezes, somente a última instância deve ser contabilizada, pois a Depita™ é programada para jogar fora a marmita anterior.

Saída

A primeira linha da saída deve conter um inteiro J , o número de depositantes diferentes.

As próximas J linhas devem conter, cada uma, a descrição de cada uma das marmitas depositadas, ordenadas em ordem alfabética do nome do depositante.

Exemplos

Entrada	Saída
3 João Bife do oião Ricardo Salsicha recheada com queijo João Cuzcuz com leite	2 Cuzcuz com leite Salsicha recheada com queijo

Entrada	Saída
2 Zé abacates Alberto sashimi à passarinho	2 sashimi à passarinho abacates

Enumeração

Gilsernardo é um curioso bibliotecário que deseja catalogar todas as obras de Shakespeare. Desde os títulos, em ordem lexicográfica, até cada uma das palavras em cada um dos livros. Só que ele percebeu que é palavra demais para ele organizar no seu bloco de notas (884.647, para ser mais exato), então ele decidiu chamar o seu amigo programador, você, para auxiliá-lo nesta atividade.

Seu trabalho é escrever um programa que, dado um trecho de obra literária, organize-a por frequência de palavras.

Entrada

A entrada consiste de apenas uma linha, a obra em questão. Perceba que todas as palavras estarão separadas por espaços, mas pode haver pontuação e variação de capitalização, que não deverão ser levados em conta na contagem. Por exemplo, “abacate”, “Abacate”, “ABACATE”, “abacate.” e demais variantes sempre contam como a mesma palavra.

Saída

Seu programa deve imprimir múltiplas linhas. Cada linha deve conter uma string W e um inteiro Q , a palavra capitalizada e sua quantidade no trecho fornecido, respectivamente. A saída deve estar em ordem de Q maior para Q menor.

Exemplos

Entrada	Saída
Come, night, come, Romeo, come, thou day in night, For thou wilt lie upon the wings of night Whiter than new snow on a raven's back.	Come 3 Night 3 Thou 2 Romeo 1 Day 1 In 1 For 1 Wilt 1 Lie 1 Upon 1 The 1 Wings 1 Of 1 Whiter 1 Than 1 New 1 Snow 1 On 1 A 1 Raven's 1 Back 1

Entrada	Saída
Oh, I have bought the mansion of love, But not possessed it, and though I am sold, Not yet enjoyed.	I 2 Not 2 Oh 1 Have 1 Bought 1 The 1 Mansion 1 Of 1 Love 1 But 1 Possessed 1 It 1 And 1 Though 1 Am 1 Sold 1 Yet 1 Enjoyed 1

Função de Ackermann

Robernardo é um jovem professor de computação que anda estudando complexidade de algoritmos, e esbarrou na função de Ackermann. Esta função é conhecida pela sua capacidade de crescer absurdamente com pequenas entradas. Ela cresce mais rápido do que qualquer função exponencial!

Ela pode ser definida por:

$$F_{Ack}(x, y) = \begin{cases} y + 1, & \text{se } x = 0 \\ F_{Ack}(x - 1, 1), & \text{se } x > 0 \text{ e } y = 0 \\ F_{Ack}(x - 1, F_{Ack}(x, y - 1)), & \text{caso contrário} \end{cases}$$

Porém, Robernardo anda com dificuldades em calcular os valores de Ackermann na mão. Por isso, ele procurou o melhor programador python que conhece, você, para criar um programa que compute os valores de Ackermann.

Entrada

A primeira e única linha da entrada contém apenas dois inteiros X e Y, as entradas da função de Ackermann.

Saída

Seu programa deve imprimir um único inteiro R, o valor de retorno de $F_{Ack}(X, Y)$.

Exemplos

Entrada	Saída
2 1	5

Entrada	Saída
3 6	509

Guloso

Bobernardo é um grande produtor de um soro muito especial, uma injeção que te dá capacidades especiais de se transformar em um titã comedor de gente. O problema é que a produção do soro envolve muitos passos independentes, mas que têm horários específicos de começo e término, e ele só pode executar uma deles de cada vez.

Tendo dificuldade em otimizar a produção de soro, Bobernardo pediu a sua ajuda para auxiliá-lo na sua tarefa de aniquilar a humanidade.

Seu trabalho é desenvolver um programa que, dados os horários de início e fim de muitas atividades (que podem ser concorrentes ou não), diga qual o número máximo de atividades daquele dia.

Entrada

A primeira linha da entrada consiste de um inteiro N , o número de atividades propostas. As próximas N linhas contém, cada uma, três strings T , C e F , o título da atividade, e os horários de começo e de fim, no formato **hh:mm** de 24h no dia, respectivamente. Suponha que nenhum horário irá atravessar meia-noite e que duas atividades diferentes nunca terão o mesmo nome.

Saída

Seu programa deve imprimir várias linhas na saída padrão, das quais a primeira deve conter um único inteiro M , o número máximo de atividades a serem realizadas naquele dia.

As próximas linhas devem conter os nomes das M atividades a serem realizadas naquele dia, em ordem de execução, separadas por espaços.

Exemplos

Entrada	Saída
3 Retirada de sangue 07:15 08:00 Testes com macacos 07:30 09:00 Futebol titã 08:10 08:20	2 Retirada de sangue Futebol titã

Entrada	Saída
5 Oração do dia 01:20 18:00 Oração de fechamento 19:00 21:00 Desmembramento casual 03:00 04:00 café da manhã 05:00 06:00 apocalipse zumbi 05:50 22:00	3 Desmembramento casual café da manhã Oração de fechamento