

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 4

(Funções, Passagem de Parâmetros e Recursividade)

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: Triângulo

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1929>

Ana e suas amigas estão fazendo um trabalho de geometria para o colégio, em que precisam formar vários triângulos, numa cartolina, com algumas varetas de comprimentos diferentes. Logo elas perceberam que não dá para formar triângulos com três varetas de comprimentos quaisquer: se uma das varetas for muito grande em relação às outras duas, não dá para formar o triângulo.

Neste problema, você precisa ajudar Ana e suas amigas a determinar se, dados os comprimentos de quatro varetas, é ou não é possível selecionar três varetas, dentre as quatro, e formar um triângulo.

Entrada

A entrada é composta por apenas uma linha contendo quatro números inteiros A, B, C e D ($1 \leq A, B, C, D \leq 100$).

Saída

Seu programa deve produzir apenas uma linha contendo apenas um caractere, que deve ser 'S' caso seja possível formar o triângulo, ou 'N' caso não seja possível formar o triângulo.

Exemplo

Entrada	Saída
6 9 22 15	S
14 40 12 60	N

Problema B: Plano de Dieta

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1248>

O doutor deu a você a sua dieta, na qual cada caractere corresponde a algum alimento que você deveria comer. Você também sabe o que você tem comido no café da manhã e no almoço, nos quais cada caractere corresponde a um tipo de alimento que você deveria ter comido aquele dia. Você decidiu que irá comer todo o restante de sua dieta durante o jantar, e você quer imprimi-la como uma String (ordenada em ordem alfabética). Se você trapaceou de algum modo (ou por comer muito de tipo de alimento, ou por comer algum alimento que não está no plano de dieta), você deveria imprimir a cadeia "CHEATER" (significa trapaceiro), sem as aspas.

Entrada

A entrada contém vários casos de teste. A primeira linha de entrada contém um inteiro N que indica a quantidade de casos de teste. Cada caso de teste é composto por três linhas, cada uma delas com uma string com até 26 caracteres de 'A'-'Z' ou vazia, representando respectivamente os alimentos da dieta, do café da manhã e do almoço.

Saída

Para cada caso de teste imprima uma string que representa os alimentos que você deveria consumir no jantar, ou "CHEATER" caso você tenha trapaceado na sua dieta.

Exemplo

Entrada	Saída
5 ABCD AB C ABEDCS EDSMB MSD A IWANTSODER SOW RAT	D ABCDES CHEATER DEIN

Problema C: Ajude Girafales

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1911>

Minutos antes do término das aulas, professor Girafales passa uma lista de presença. Certo dia, ele resolveu conferir as assinaturas e notou que alguns alunos assinavam diferente em algumas aulas e desconfiou que alguém poderia estar assinando por eles. Como o professor possui muitos alunos e pouco tempo (o café com dona Florinda é prioridade), ele pediu sua ajuda para validar as assinaturas. Uma assinatura é considerada falsa se houver mais de uma diferença entre a original e a que estiver sendo checada. Considere diferença uma troca de maiúscula para minúscula ou o contrário.

Entrada

Haverá diversos casos de testes. A primeira linha de cada caso inicia com um inteiro N ($1 \leq N \leq 50$) representando a quantidade de alunos de sua turma. As próximas N linhas serão da seguinte forma:

Nome do aluno Assinatura Original

A seguir haverá um inteiro M ($0 \leq M \leq N$), representando a quantidade de alunos que compareceram a uma aula. M linhas seguem, no seguinte formato:

Nome do aluno Assinatura na aula

Todos os alunos possuem apenas o primeiro nome na lista, nenhum nome se repete e todos os nomes contêm no máximo 20 letras (a-z A-Z). A entrada termina com $N = 0$, a qual não deve ser processada.

Saída

Para cada caso, exiba uma única linha, a quantidade de assinaturas falsas encontradas.

Exemplo

Entrada	Saída
4	1
Chaves ChAvEs	2
Kiko kiko	
Nhonho NHONHO	
Chiquinha CHIquinHa	
3	
Chaves ChAvEs	
Kiko kIKO	
Chiquinha CHIquinHA	
2	
Jadson jadsON	
Crishna Crishna	
2	
Crishna CRISHNA	
Jadson JADson	
0	

Problema D: Projeta sua Senha

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2287>

Por questões de segurança, muitos bancos hoje em dia estão alterando a forma como seus clientes digitam as senhas nos caixas eletrônicos, pois alguém pode postar-se atrás do cliente e ver as teclas à medida em que ele as digita.

Uma alternativa bastante utilizada tem sido associar os dez dígitos a cinco letras, de forma que cada letra esteja associada a dois dígitos, conforme o exemplo abaixo:

A	1
	7

B	3
	9

C	0
	8

D	5
	6

E	2
	4

As associações entre números e letras são mostradas como botões numa tela sensível ao toque, permitindo que o cliente selecione os botões correspondentes à senha. Considerando a disposição dos botões da figura acima, a senha 384729 seria digitada como BCEAEB (note que a mesma sequência de letras seria digitada para outras senhas, como por exemplo 982123).

Cada vez que o cliente usa o caixa eletrônico, as letras utilizadas são as mesmas (de ‘A’ a ‘E’), com os botões nas mesmas posições, mas os dígitos são trocados de lugar. Assim, caso um intruso veja (mesmo que mais de uma vez) a sequência de letras digitada, não é possível notar facilmente qual a senha do cliente do banco.

Dada uma sequência de associações entre letras e números, e as letras digitadas pelo cliente do banco para cada uma dessas associações, você deve escrever um programa para determinar qual é a senha do cliente.

Entrada

A entrada é composta de vários conjuntos de testes. A primeira linha de um conjunto de testes contém um inteiro N , que indica o número de associações entre letras e números e as senhas digitadas ($2 \leq N \leq 10$). As N linhas seguintes contêm as entradas da seguinte forma: 10 dígitos, em ordem de associação, para as letras de ‘A’ a ‘E’ (2 dígitos para a letra A, 2 para a B e assim sucessivamente) e 6 letras que representam a senha codificada conforme os dígitos anteriores. As N associações fornecidas em um conjunto de testes serão sempre suficientes para definir univocamente a senha do cliente. O final da entrada é indicado por $N = 0$.

Saída

Para cada conjunto de teste da entrada, seu programa deve produzir três linhas na saída. A primeira linha deve conter um identificador do conjunto de teste, no formato “Teste n ”, onde n é numerado sequencialmente a partir de 1. A segunda linha deve conter a senha

do cliente, com um espaço após cada dígito. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo

Entrada	Saída
2	Teste 1
1 7 3 9 0 8 5 6 2 4 B C E A E B	3 8 4 7 2 9
9 0 7 5 8 4 6 2 3 1 E C C B D A	
3	Teste 2
0 1 2 3 4 5 6 7 8 9 B C D D E E	2 5 6 7 8 9
1 3 5 4 6 8 7 9 0 2 E B C D C D	
3 2 0 4 5 9 7 6 8 1 A C D D E C	
0	

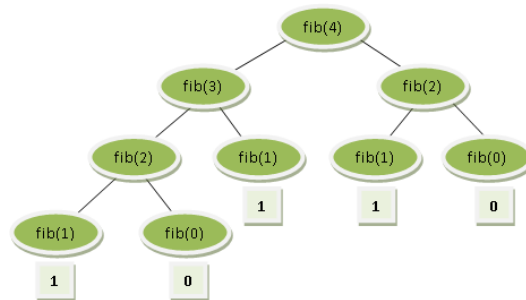
Problema E: Fibonacci, Quantas Chamadas?

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1029>

Quase todo estudante de Ciência da Computação recebe em algum momento no início de seu curso de graduação algum problema envolvendo a sequência de Fibonacci. Tal sequência tem como os dois primeiros valores 0 (zero) e 1 (um) e cada próximo valor será sempre a soma dos dois valores imediatamente anteriores. Por definição, podemos apresentar a seguinte fórmula para encontrar qualquer número da sequência de Fibonacci:

```
fib(0) = 0
fib(1) = 1
fib(n) = fib(n-1) + fib(n-2);
```

Uma das formas de encontrar o número de Fibonacci é através de chamadas recursivas. Isto é ilustrado a seguir, apresentando a árvore de derivação ao calcularmos o valor fib(4), ou seja o 5º valor desta sequência:



Desta forma,

- $\text{fib}(4) = 1 + 0 + 1 + 1 + 0 = 3$
- Foram feitas 8 calls, ou seja, 8 chamadas recursivas

Entrada

A primeira linha da entrada contém um único inteiro N, indicando o número de casos de teste. Cada caso de teste contém um inteiro X ($1 \leq X \leq 39$).

Saída

Para cada caso de teste de entrada deverá ser apresentada uma linha de saída, no seguinte formato: $\text{fib}(n) = \text{num_calls calls} = \text{result}$, aonde num_calls é o número de chamadas recursivas, tendo sempre um espaço antes e depois do sinal de igualdade, conforme o exemplo abaixo.

Exemplo

Entrada	Saída
2	fib(5) = 14 calls = 5
5	fib(4) = 8 calls = 3
4	

Problema F: Lucro

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1310>

George é dono de um circo e traz seu circo de cidade em cidade. Ele sabe o quanto de receita ele pode obter em qualquer dia de uma série de dias em uma cidade. Ele também sabe o custo constante diário para manter o seu circo. George quer trazer seu circo à cidade para a série de dias que resulta em maior lucro. Por exemplo, se em uma determinada cidade o custo for de \$ 20 por dia em um exemplo com 6 dias, sendo que as receitas previstas por dia são {\$ 18, \$ 35, \$ 6, \$ 80, \$ 15, \$ 21}, George pode obter o máximo de lucro trazendo o seu circo para esta cidade do dia 2 ao dia 4. Desta forma ele pode lucrar $(35 + 80 + 6) - (3 * 20) = \$ 61$.

Nota: A série de dias que George traz seu circo para a cidade pode ser entre 0 e o número máximo de dias, inclusive. Obviamente, se George traz seu circo para a cidade por 0 dias, ele obtém \$ 0 de lucro.

Entrada

A entrada contém muitos casos de teste. A primeira linha de cada caso de teste contém um inteiro N ($1 \leq N \leq 50$) que representa o número de dias que George pode trazer o seu circo para a cidade. A segunda linha do caso de teste contém um número inteiro custoPorDia ($0 \leq \text{custoPorDia} < 1000$) que representa o custo em manter o circo na cidade. Segue N linhas (uma por cada dia), contendo cada um inteiro receita ($0 \leq \text{receita} < 1000$) representa a receita que o circo obtém em cada dia. O final da entrada é indicado por EOF (fim de arquivo).

Saída

Para cada caso de teste imprima o máximo de dinheiro que George pode ganhar trazendo o seu circo para a cidade de acordo com o exemplo abaixo.

Exemplo

Entrada	Saída
6	61
20	0
18	
35	
6	
80	
15	
21	
4	
40	
30	
20	
10	
38	