

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 08**

*(Estruturas de Dados: Lista e Lista de Lista)*

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: O Despertar da Força

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2163>

Há muito tempo atrás, em uma galáxia muito, muito distante...

Após o declínio do Império, sucateiros estão espalhados por todo o universo procurando por um sabre de luz perdido. Todos sabem que um sabre de luz emite um padrão de ondas específico: 42 cercado por 7 em toda a volta. Você tem um sensor de ondas que varre um terreno com  $N \times M$  células. Veja o exemplo abaixo para um terreno  $4 \times 7$  com um sabre de luz nele (na posição (2, 4)).

11	12	7	7	7	13	14
15	6	7	42	7	7	42
98	-5	7	7	7	42	7
-1	42	3	9	7	7	7

Você deve escrever um programa que, dado um terreno  $N \times M$ , procura pelo padrão do sabre de luz nele. Nenhuma varredura tem mais do que um padrão de sabre de luz.

### Entrada

A primeira linha da entrada tem dois números positivos  $N$  e  $M$ , representando, respectivamente, o número de linhas e de colunas varridos no terreno ( $3 \leq N, M \leq 1000$ ). Cada uma das próximas  $N$  linhas tem  $M$  inteiros, que descrevem os valores lidos em cada célula do terreno ( $-100 \leq T_{ij} \leq 100$ , para  $1 \leq i \leq N$  e  $1 \leq j \leq M$ ).

### Saída

A saída é uma única linha com 2 inteiros  $X$  e  $Y$  separados por um espaço. Eles representam a coordenada  $(X, Y)$  do sabre de luz, caso encontrado. Se o terreno não tem um padrão de sabre de luz,  $X$  e  $Y$  são ambos zero.

**Exemplo**

Entrada	Saída
4 7 11 12 7 7 7 13 14 15 6 7 42 7 7 42 98 -5 7 7 7 42 7 -1 42 3 9 7 7 7	2 4

Entrada	Saída
4 7 11 12 7 7 7 13 14 15 6 7 12 7 7 42 98 -5 7 7 7 42 7 -1 42 3 9 7 7 7	0 0

Entrada	Saída
3 3 7 7 7 7 42 7 7 7 7	2 2

## Problema B: Crepúsculo em Portland

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2168>

No crepúsculo, a cidade de Portland fica cheia de vampiros e lobisomens. Entretanto, nenhum deles quer ser visto enquanto passeiam pelo centro.

Vão ser instaladas câmeras de vigilância em cada esquina do centro de Portland. A cada mês, um mapa atualizado com as câmeras já em funcionamento é disponibilizado no site da prefeitura.

Uma quadra é considerada segura se existem câmeras em, pelo menos, duas de suas quatro esquinas. No centro de Portland todas as quadras são quadrados de mesmo tamanho.

Sua tarefa é, dado o mapa das câmeras em funcionamento nas esquinas, indicar o status de todas as quadras do centro.

### Entrada

A primeira linha da entrada tem um inteiro positivo  $N$  ( $1 \leq N \leq 100$ ). Nas próximas  $N+1$  linhas, existem  $N+1$  números, que indicam, para cada esquina, a presença ou ausência de uma câmera de vigilância em funcionamento. O número 1 indica que existe uma câmera funcionando na esquina, enquanto o número zero indica que não há câmera funcionando.

### Saída

A saída é dada em  $N$  linhas. Cada linha tem  $N$  caracteres, indicando se a quadra correspondente é segura ou insegura. Se uma quadra é segura, mostre o caractere S; se não é segura, mostre o caractere U.

### Exemplo

Entrada	Saída
1 1 0 0 0	U

Entrada	Saída
2 1 0 0 1 1 0 0 0 1	SU SS

Entrada	Saída
3 1 1 0 1 1 0 1 0 1 0 0 1 0 1 1 0	SSS SUS SSS

## Problema C: Pouca Frequência

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1277>

Os estudantes da tua universidade recentemente adquiriram o desagradável hábito de cabular as aulas. Para enfrentar este problema o seu Conselho de Professores decidiu somente permitir que estudantes com ao menos 75% de presença prestem os exames. A partir de uma lista de nomes de estudantes e seus respectivos registros de frequência, imprima o nome dos estudantes que não obtiveram o mínimo de presença às aulas e que consequentemente não poderão prestar os exames.

### Entrada

A entrada possui diversos casos de testes. A primeira linha da entrada contém um inteiro **T**, que indica o número de casos de testes que se seguem.

Cada caso de teste é composto por três linhas:

- A primeira linha de um caso de teste irá conter um inteiro **N** ( $0 \leq N \leq 100$ ) que indica o número de estudantes na turma.
- A segunda linha conterá **N** nomes de estudantes com até 50 caracteres cada nome, separados por um único espaço. Todos os nomes irão conter somente letras maiúsculas e minúsculas ('A'-'Z', 'a'-'z').
- A terceira linha conterá **N** registros de frequência, correspondentes aos respectivos estudantes da linha anterior. Os registros virão separados por um único espaço, e contêm apenas os caracteres 'A', 'P' e 'M'. Um 'P' indica que o estudante estava presente à aula, 'A' indica que ele estava ausente (ele cabulou à aula) e 'M' mostra que, apesar de não ir à aula, ele entregou um atestado médico, então esta aula não deverá ser considerada no cálculo da frequência do estudante. Registros de frequência conterão ao menos um caracter 'A' ou 'P'.

### Saída

Para cada caso de teste imprima os nomes de todos os estudantes que não cumpriram a presença mínima requerida, separados por um espaço. Não deixe espaços sobrando no final da linha.

**Exemplo**

Entrada	Saída
4	Justin
1	Justin
Justin	
PAAPP	Arjun Nikhil Taneja
2	
Justin Chris	
PAAPP PPPPA	
1	
Sunny	
PPPAM	
4	
Mansi Arjun Nikhil Taneja	
PPPPMPPAPP AAMAAPP PPPPAAP	
PPPAAMPP	

## Problema D: Frase Completa

<https://www.urionlinejudge.com.br/judge/pt/problems/view/1551>

Seu professor de português não para de trazer coisas novas para a sala, e hoje não foi diferente. Existe uma cidade, segundo seu professor, onde as pessoas levam muito a sério a forma como elas se comunicam. Em especial, quando duas pessoas estão conversando, elas pensam muito nas frases antes de dizê-las, de forma a garantir que tal frase seja uma “frase completa”, ou talvez uma “frase quase completa”.

Considerando o nosso alfabeto de 26 letras, uma frase é dita “completa” quando ela contém todas as letras do alfabeto contidas nela. De modo semelhante, uma frase é dita “quase completa” se ela não é completa, porém contém ao menos metade das letras do alfabeto contidas nela. Quando uma frase não é “completa” e nem “quase completa”, ela é dita “mal elaborada”.

Seu professor lhe deu uma tarefa muito difícil: dadas várias frases trocadas entre vários habitantes da cidade citada, diga em qual das categorias acima a frase se encaixa.

### Entrada

A primeira linha contém um inteiro  $N$ , indicando o número de casos de teste a seguir.

Cada caso de teste contém uma linha, contendo letras minúsculas, espaços em branco e/ou vírgulas. O número de caracteres de cada linha é no mínimo 3 e no máximo 1000, contando com os espaços.

### Saída

Para cada caso de teste, imprima uma linha contendo uma das seguintes frases: “frase completa”, quando a frase é considerada completa; “frase quase completa”, quando a frase não é completa, mas é considerada quase completa; ou “frase mal elaborada”, quando ela não é completa e nem quase completa.

### Exemplo

Entrada	Saída
2 ola, como voce esta hoje hoje fui na feira, e comprei banana, melao e abacates	frase mal elaborada frase quase completa



## Problema E: Quermesse

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2189>

Os alunos do último ano resolveram organizar uma quermesse para arrecadar fundos para a festa de formatura. A festa prometia ser um sucesso, pois o pai de um dos formandos, Teófilo, dono de uma loja de informática, decidiu doar um computador para ser sorteado entre os que comparecessem. Os alunos prepararam barracas de quentão, pipoca, doces, ensaiaram a quadrilha e colocaram à venda ingressos numerados sequencialmente a partir de 1. O número do ingresso serviria para o sorteio do computador. Ficou acertado que Teófilo decidiria o método de sorteio; em princípio o sorteio seria, claro, computadorizado.

O local escolhido para a festa foi o ginásio da escola. A entrada dos participantes foi pela porta principal, que possui uma roleta, onde passa uma pessoa por vez. Na entrada, um funcionário inseriu, em uma lista no computador da escola, o número do ingresso, na ordem de chegada dos participantes. Depois da entrada de todos os participantes, Teófilo começou a trabalhar no computador para preparar o sorteio. Verificando a lista de presentes, notou uma característica notável: havia apenas um caso, em toda a lista, em que o participante que possuía o ingresso numerado com  $i$ , havia sido a  $i$ -ésima pessoa a entrar no ginásio. Teófilo ficou tão encantado com a coincidência que decidiu que o sorteio não seria necessário: esta pessoa seria o ganhador do computador.

Conhecendo a lista de participantes, por ordem de chegada, sua tarefa é determinar o número do ingresso premiado, sabendo que o ganhador é o único participante que tem o número do ingresso igual à sua posição de entrada na festa.

### Entrada

A entrada é composta de vários conjuntos de teste. A primeira linha de um conjunto de teste contém um número inteiro positivo  $N$  ( $0 \leq N \leq 10000$ ) que indica o número de participantes da festa. A linha seguinte contém a sequência, em ordem de entrada, dos  $N$  ingressos das pessoas que participaram da festa. O final da entrada é indicado quando  $N = 0$ . Para cada conjunto de teste da entrada haverá um único ganhador.

### Saída

Para cada conjunto de teste da entrada seu programa deve produzir três linhas. A primeira linha identifica o conjunto de teste, no formato "Teste  $n$ ", onde  $n$  é numerado a partir de 1. A segunda 5 linha deve conter o número do ingresso do ganhador, conforme determinado pelo seu programa. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

**Exemplo**

Entrada	Saída
4 4 5 3 1 10 9 8 7 6 1 4 3 2 12 10 0	Teste 1 3  Teste 2 10

## Problema F: Saldo de Gols

<https://www.urionlinejudge.com.br/judge/pt/problems/view/2191>

Hipólito é um torcedor fanático. Coleciona flâmulas, bandeiras, recortes de jornal, figurinhas de jogadores, camisetas e tudo o mais que se refira a seu time preferido. Quando ganhou um computador de presente em uma festa, resolveu montar um banco de dados com os resultados de todos os jogos de seu time ocorridos desde a sua fundação, em 1911. Depois de inseridos os dados, Hipólito começou a ficar curioso sobre estatísticas de desempenho do time. Por exemplo, ele deseja saber qual foi o período em que o seu time acumulou o maior saldo de gols. Como Hipólito tem o computador há muito pouco tempo, não sabe programar muito bem, e precisa de sua ajuda.

É dada uma lista, numerada sequencialmente a partir de 1, com os resultados de todos os jogos do time (primeira partida: 3 x 0, segunda partida: 1 x 2, terceira partida: 0 x 5 ...). Sua tarefa é escrever um programa que determine em qual período o time conseguiu acumular o maior saldo de gols. Um período é definido pelos números de sequência de duas partidas, A e B ( $A \leq B \leq N$ ). O saldo de gols acumulado entre A e B é dado pela soma dos gols marcados pelo time em todas as partidas realizadas entre A e B (incluindo as mesmas) menos a soma dos gols marcados pelos times adversários no período. Se houver mais de um período com o mesmo saldo de gols, escolha o maior período (ou seja, o período em que  $B - A$  é maior). Se ainda assim houver mais de uma solução possível, escolha qualquer uma delas como resposta.

### Entrada

Seu programa deve ler vários conjuntos de teste. A primeira linha de um conjunto de teste contém um inteiro não negativo,  $N$  ( $0 \leq N \leq 10000$ ), que indica o número de partidas realizadas pelo time (o valor  $N = 0$  indica o final da entrada). Seguem-se  $N$  linhas, cada uma contendo um par de números inteiros não negativos  $X$  e  $Y$  ( $0 \leq X, Y \leq 50$ ) que representam o resultado da partida:  $X$  são os gols a favor e  $Y$  os gols contra o time de Hipólito. As partidas são numeradas sequencialmente a partir de 1, na ordem em que aparecem na entrada.

### 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 a partir de 1. A segunda linha deve conter um par de inteiros  $I$  e  $J$  que indicam respectivamente a primeira e última partidas do melhor período, conforme determinado pelo seu programa, exceto quando o saldo de gols do melhor período for menor ou igual a zero; neste caso a segunda linha deve conter a expressão “nenhum”. 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
2 3	2 2
7 1	
9	Teste 2
2 2	3 8
0 5	
6 2	
1 4	Teste 3
0 0	nenhum
5 1	
1 5	
6 2	
0 5	
3	
0 2	
0 3	
0 4	
0	