

Projeto e Análise de Algoritmos

Primeiro Trabalho Prático (Primeira Versão)

Data Limite para Entrega: 14/12/2024

2º Semestre 2024 - DC-UFSCar

1 Introdução

No primeiro Trabalho Prático (TP01) será solicitada a entrega de programas que solucionem os problemas apresentados nas próximas seções. Os programas entregues devem seguir os itens abaixo:

- Os programas devem ser feitos individualmente e plágio não será tolerado;
- Os programas devem ser entregues no beecrowd (<https://www.beecloud.com.br>) com código em linguagem C, contendo um cabeçalho com as informações do estudante (nome, curso, RA);
- Cada estudante deve se cadastrar no beecrowd (<https://www.beecloud.com.br>), acessar o beecrowd academic (<https://www.beecloud.com.br/judge/pt/disciplines>) e se matricular na disciplina “PAA 2024s2 Turmas A, B e C” usando Chave CyWtic8 e ID 013672;
- Na linguagem C, compile seus programas usando o compilador GCC com flags -Wall -pedantic -O2 -Wno-unused-result, pois warnings podem impedir o código de funcionar no beecrowd ainda que funcionem no seu computador.
- Importantel: Para todos os estudantes de computação (engenharia ou ciência), gostaria que o trabalho fosse feito em linguagem C, como está especificado no enunciado.
- Importante2: Eu posso liberar submissão em Python para que os estudantes de outros cursos possam usar esta linguagem, já que eles não veem introdução à programação em C.
- Importante3: Os estudantes de outros cursos que não computação, por favor me enviem uma mensagem informando nome, RA e curso, para eu saber se devo criar uma versão da lista de exercícios que aceite soluções em Python.

Notem que os problemas que formam este trabalho prático fazem parte da lista de exercícios “Divisão e Conquista” na página da disciplina no beecrowd.

2 Cálculo de Exponencial

A operação de exponencial corresponde a uma sequência de multiplicações. Neste problema queremos descobrir o valor de um número base elevado a um determinado expoente, sendo que a base é um real positivo e o expoente é um inteiro não negativo.

Entrada

Um número real $B \geq 1$ e um inteiro $N \geq 0$ deverão ser lidos da entrada padrão.

Saída

O valor correspondente a B^N deve ser calculado e deve ser impresso na saída padrão uma linha (terminada em '\n') com o dígito mais significativo de B^N e a ordem de grandeza de B^N na base 10, i.e., o valor do expoente do valor em notação científica.

Exemplos de entradas e saídas correspondentes

Entrada	Saída	Entrada	Saída
2.0 1	2 0	2.0 10	1 3
Entrada	Saída	Entrada	Saída
5.0 0	1 0	1.5 2	2 0

3 Distância Genômica

Dada a sequência de genes de dois indivíduos diferentes, podemos inferir quão distantes eles estão evolutivamente contando o número de mutações gênicas necessárias para transformar o genoma de um indivíduo no do outro. Existem diversos tipos de mutações gênicas, mas uma das mais comuns é a troca da posição de genes adjacentes, ou seja, a inversão de dois genes. Vamos focar apenas neste tipo de mutação, o que nos leva a supor que os dois indivíduos analisados tem os mesmos genes. Para simplificar, vamos numerar os genes dos indivíduos de 0 a $N - 1$, sendo N o número de genes. Seu objetivo é, dadas duas sequências de números inteiros correspondendo aos genomas dos indivíduos 1 e 2, determinar a distância genômica baseada apenas em inversões entre os dois indivíduos.

Entrada

A instância deve ser lida da entrada padrão. A primeira linha informa a quantidade N de genes dos indivíduos comparados, a segunda linha apresenta uma

sequência de N números inteiros (entre 0 e $N - 1$) correspondendo ao genoma do indivíduo 1, e a terceira linha apresenta uma sequência de N números inteiros (entre 0 e $N - 1$) correspondendo ao genoma do indivíduo 2.

Saída

Deve ser impresso na saída padrão o menor número de inversões de genes necessário para transformar o genoma do indivíduo 2 no genoma do indivíduo 1.

Exemplos de entradas e saídas correspondentes

Entrada	Saída	Entrada	Saída
5	7	5	0
0 1 2 3 4		0 1 2 3 4	
1 4 3 2 0		0 1 2 3 4	
Entrada	Saída	Entrada	Saída
5	0	5	10
1 4 3 2 0		4 3 2 1 0	
1 4 3 2 0		0 1 2 3 4	

4 Detecção de Outliers

A detecção de outliers é uma atividade muito comum em diversas áreas, como a Estatística e a Mineração de Dados. Um outlier é um objeto cujos valores de seus atributos se difere muito da maioria dos outros objetos observados. Calma, jovem. Já vamos explicar melhor.

Um objeto pode ser qualquer tipo de entidade, descrita por atributos. Atributos são características que descrevem esse objeto. Por exemplo, uma pessoa pode ser considerada um objeto. Seus atributos seriam, entre outros, sua altura, seu peso e sua idade. Uma pessoa com 2,35 metros de altura, por exemplo, pode ser considerada um outlier.

A definição de outlier pode variar de uma área para outra, ou até mesmo de um autor para outro, na literatura. Neste trabalho, utilizaremos uma definição que é muito comum na literatura de Mineração de Dados. Um outlier é aquele objeto cuja distância até o seu k -ésimo vizinho mais próximo é estritamente maior que um limiar L . Ou seja, se a distância para o k -ésimo objeto mais próximo de um objeto A ultrapassar um valor previamente definido L , então A pode ser considerado um outlier. Lembre que o primeiro vizinho mais próximo é o 0-ésimo. A distância entre dois objetos pode ser calculada de várias maneiras. Neste trabalho essas distâncias já foram calculadas e são parte da entrada.

Entrada

Cada caso de teste é iniciado por uma linha contendo um inteiro N ($2 \leq N \leq 10^9$), um inteiro S ($1 \leq S \leq 1000$), um inteiro k ($0 \leq k < N$) e um real positivo L . O total de objetos no conjunto considerado é N . Porém, em função de restrições no tamanho das entradas, só vamos procurar outliers entre os primeiros S objetos desse conjunto. Assim, temos S linhas após a primeira, e cada uma contém N números. Vamos associar um índice i , entre 0 e $S - 1$ a cada linha após a primeira. Especificamente, a i -ésima linha contém a distância do i -ésimo objeto para todos os demais. Assim, associemos um índice j , entre 0 e $N - 1$, a cada valor de uma determinada linha. Dessa forma, $D_{i,j}$ é o j -ésimo valor da linha i e indica a distância de i até j . Na entrada, é garantido que $D_{i,i} = 0$, ou seja, a distância de um objeto a si mesmo é zero.

Saída

A saída de um caso de teste contém apenas uma linha (terminada em ‘\n’) contendo um inteiro, que indica o número de outliers encontrados.

Exemplos de entradas e saídas correspondentes

Entrada	Saída	Entrada	Saída
5 5 0 0.7	2	5 2 1 1.3	2
0.0 1.0 1.5 3.0 2.0		0.0 1.0 1.5 3.0 2.0	
1.0 0.0 2.0 5.0 1.5		1.0 0.0 2.0 5.0 1.5	
2.2 1.7 0.0 2.3 0.7			
1.8 0.3 0.1 0.0 2.4			
1.5 1.6 2.3 0.7 0.0			

Desafio: Outliers Variante Distância Média

Nesta versão do problema anterior, ao invés de receber um limiar L na entrada, vamos calcular a distância média de cada um dos S objetos para o seu k -ésimo vizinho mais próximo, como mostram os seguintes exemplos. A média deve ser arredondada para duas casas decimais.

Entrada	Saída	Entrada	Saída
5 5 0	0.70	5 2 1	1.50
0.0 1.0 1.5 3.0 2.0		0.0 1.0 1.5 3.0 2.0	
1.0 0.0 2.0 5.0 1.5		1.0 0.0 2.0 5.0 1.5	
2.2 1.7 0.0 2.3 0.7			
1.8 0.3 0.1 0.0 2.4			
1.5 1.6 2.3 0.7 0.0			