

Universidade Federal de Goiás – UFG
Instituto de Informática – INF
Bacharelados (Núcleo Básico Comum)

Algoritmos e Estruturas de Dados 1 – 2022/1

Lista de Exercícios nº 05 – Algoritmos de Ordenação Interna

Sumário

1	Separando Números Pares de Ímpares	2
2	Insertion - Selection	3
3	Ordenação	4
4	Olimpíadas	5

1 Separando Números Pares de Ímpares

Tales, um menino muito levado, pegou na escola uma caixa repleta de números naturais impressos em cartelas de EVA (*Espuma Vinílica Acetinada*) e derrubou-os sobre o chão da sala de aula. Por estranho que pareça, ao caírem os números formaram uma *fila indiana* de tal maneira que ficaram com seus valores distribuídos aleatoriamente nesta fila.

Sabe-se que na caixa havia $n \in \mathbb{N}^*$ números, com $(1 < n \leq 100)$, mas seus valores são desconhecidos. Sua tarefa é conceber um programa C que seja capaz de ordenar esta fila, segundo as seguintes regras:

- primeiro devem vir todos os números pares, em ordem crescente;
- depois devem vir os números ímpares, em ordem decrescente.

Entrada

A primeira linha de entrada contém o número n , quantidade de números existente na caixa que Tales derrubou.

A segunda linha contém os n números naturais, na ordem em que formaram a fila indiana, sempre separados por um único espaço em branco entre eles.

Saída

A saída deverá ter duas linhas. Na primeira são apresentados os números pares e na segunda os números ímpares, sempre separados por um único espaço em branco entre eles, conforme a ordem definida anteriormente.

Exemplos

Observação: Note que se, como exceção, a saída poderá ter uma única linha, se os números inicialmente fornecidos forem todos pares ou todos ímpares.

Entrada	Saída
10 4 32 34 543 3456 654 567 87 6789 98	4 32 34 98 654 3456 6789 567 543 87

Entrada	Saída
7 2 5 6 51 512 913 375	2 6 512 913 375 51 5

Entrada	Saída
8 6 2 8 12 202 304 18 10	2 6 8 10 12 18 202 304

2 Insertion - Selection

Escreva um programa C que, a partir de um vetor de números naturais fornecido como entrada, calcule a diferença entre o número de trocas realizadas pelos algoritmos `insertionSort` e `selectionSort`, nesta ordem.

Cada movimentação efetiva de um número no vetor deve ser contabilizada. Os algoritmos devem ser implementados de maneira a realizar o menor número de trocas possível.

Entrada

A primeira entrada é um número natural n , $1 \leq n \leq 1000$, que representa o tamanho do vetor de entrada. A próxima linha contém os elementos do vetor, sempre fornecidos da primeira posição até a última, e separados por um único espaço em branco entre si.

Saída

A saída consiste de uma única linha que contém a diferença entre o número de trocas realizadas pelo `insertionSort` e pelo `selectionSort`, nesta ordem.

Exemplos

Entrada	Saída
20 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52	19

Entrada	Saída
20 52 51 50 49 48 47 46 45 44 43 42 41 40 39 38 37 36 35 34 33	199

Entrada	Saída
10 8 6 4 3 2 1 7 9 5 10	23

3 Ordenação

Escreva um programa C para, a partir de dois números naturais dados, l e k , com $0 < l < k \leq n$, imprima a soma do l -ésimo menor elemento do vetor com o k -ésimo menor elemento do vetor, considerando-se que o referido vetor possui n elementos.

Entrada

A primeira linha da entrada contém o número natural n , que representa o tamanho do vetor, com $1 \leq n \leq 1000$.

A segunda linha da entrada contém os n elementos do vetor, sempre apresentados a partir da primeira posição até a última posição, ou seja, das posições $1, 2, 3, \dots, n$, nesta ordem.

Por fim, a terceira linha da entrada contém os valores de l e k , nesta ordem, e separados por um único espaço em branco.

Saída

Seu programa deve imprimir, numa única linha, o valor a soma do l -ésimo menor elemento do vetor com o k -ésimo menor elemento do vetor.

Exemplos

Entrada	Saída
9 9 8 7 6 5 4 3 2 1 3 4	7

Entrada	Saída
6 1 2 3 6 4 4 5 2	6

Entrada	Saída
6 1 5 9 7 5 3 5 5	14

Entrada	Saída
5 1 7 5 3 9 4 3	12

Observação: O valor de cada elemento do vetor está no intervalo de 1 a 10000, inclusive extremos.

4 Olimpíadas

O Comitê Olímpico Internacional (COI) está visitando as cidades candidatas a sediar as Olimpíadas de 2032 e, desta vez, Goiânia é uma das cidades concorrentes, mas a competição entre as cidades candidatas está muito acirrada, pois Buenos Aires está sendo considerada uma forte candidata.

O COI tem um conjunto de exigências que devem ser obedecidas pelas cidades candidatas, como: (1) boas *arenas* para os jogos (ginásios, campos de futebol, pistas de atletismo, parque aquático,...); (2) bons alojamentos; (3) um plano para o tráfego de veículos durante os jogos, etc.

Durante sua visita à Goiânia, o COI colocou ainda mais uma exigência: a demonstração da qualidade dos sistemas de informática. Especificamente, o COI quer que a organização local demonstre a sua capacidade em informática produzindo um programa de computador que gere a classificação final dos países, considerando o número de medalhas recebidas pelos atletas de cada país participante da futura olimpíada.

Para ser justo, o COI abriu um concurso que permite que equipes de estudantes de quaisquer faculdades, centros universitários e universidades (públicas e privadas) apresentem seus *programas de computador* para cumprir a tarefa. O programa que melhor resultado obtiver numa *bancada de testes* a ser proposta pelo COI, em momento oportuno, será o escolhido.

Tarefa

Você está na equipe que o INF/UFG designou para *vencer* este concurso.

Sua tarefa é escrever um programa **C** que, dada a informação dos países que receberam medalhas de ouro, prata e bronze em cada modalidade esportiva, gere a lista de classificação dos países na competição.

Nesta tarefa, os países serão identificados por números inteiros: $1, 2, 3, \dots, n$, com $2 \leq n \leq 200$.

O melhor colocado deve ser o país que conseguiu o maior número de medalhas de ouro, havendo empate entre dois ou mais países, o melhor colocado é o país que conseguiu o maior número de medalhas de prata. Novamente havendo empate, o melhor colocado é o país que recebeu o maior número de medalhas de bronze. Por fim, se ainda assim houver empate, o melhor classificado é o que tem o maior número de identificação.

Entrada

A primeira linha da entrada contém dois números naturais n e m , separados por um único espaço em branco, e indicando, respectivamente, o número de países e número de modalidades esportivas envolvidas na competição ($1 \leq m \leq 50$). Os países são identificados por números inteiros de 1 a n .

Cada uma das m linhas seguintes contém três números inteiros O , P e B , separados por um único espaço em branco entre eles, representando os identificadores dos países cujos atletas receberam, respectivamente, medalhas de ouro, prata e bronze.

Assim, se uma das m linhas contém os números 3 2 1, significa que nessa modalidade a medalha de ouro foi ganha pelo país cuja identificação é 3, a de prata pelo país cuja identificação é 2 e a de bronze pelo país cuja identificação é 1.

Uma linha com 10 10 10, significa que o país cuja identificação é 10 recebeu as três medalhas daquela modalidade (ouro, prata e bronze).

Saída

Seu programa deve imprimir, na saída padrão, uma única linha contendo n números, separados por um único espaço em branco entre eles, representando os países na ordem decrescente de classificação, da esquerda para a direita.

Exemplos

Entrada	Saída
2 2 2 1 2 1 2 2	2 1

Entrada	Saída
4 3 3 2 1 4 3 1 4 3 1	4 3 2 1

Entrada	Saída
3 3 3 1 2 2 3 1 1 2 3	3 2 1