

**Sistemas de Informação  
Algoritmos e Programação II  
Trabalho Prático Final 2020.1**

Esse trabalho prático pode ser feito em equipes de até 3 componentes e os pontos desse trabalho terão a seguinte organização:

- Funcionamento correto do programa (5.0 pontos)
- Vídeo de 7-10 minutos para a apresentação do trabalho (5.0 pontos)

O prazo para a submissão do trabalho é dia 08/07/2021 (quinta-feira) às 23:59.

Um membro do grupo deve submeter o código fonte e arquivos de entrada e saída testados para o programa, compactados no formato ZIP no Sigaa, e enviar um link para o vídeo compartilhado em qualquer sistema de armazenamento em nuvem (Dropbox, Google Drive, MS Onedrive etc.) no campo de comentários da submissão do Sigaa.

### **Instruções**

1. Forme uma equipe e escolha um dentre os dez problemas listados no último item que sua equipe irá tratar. Um membro do grupo deve postar uma mensagem no fórum “Trabalho Final”, para informar os membros da equipe e o problema. A definição de problema por equipe será dada pela ordem de postagem no fórum.

**Cada problema pode ser escolhido por até duas equipes (máximo de duas repetições por problema). Portanto, antes de escolher o problema verifique a lista daqueles já escolhidos.**

2. O problema deve ser implementado na linguagem C. Para padronizar a correção do trabalho prático, o código deve ser executado em terminal do sistema Linux, Windows ou MacOS. A execução do programa será via linha de comando e as entradas e saídas do problema serão via arquivos. Dessa forma a execução do programa deve ser acompanhado de dois nomes de arquivos (parâmetros) separados por espaço. O primeiro arquivo deve ser preparado previamente e contém a entrada do problema conforme o enunciado. O segundo arquivo será gerado pelo programa e deve conter a saída com os resultados do problema, conforme as definições do enunciado. Por exemplo, esse é o comando de execução do programa em sistemas Linux:

*./tp entrada.txt saida.txt*

Nesse exemplo, *entrada.txt* e *saida.txt* são, respectivamente, os nomes dos arquivos de entrada e saída requeridos como o primeiro e segundo parâmetros. Esses arquivos podem ter qualquer nome, o importante é que o primeiro seja a entrada e o segundo seja a saída do problema.

Os arquivos de entrada e saída mencionados devem seguir rigorosamente o formato descrito nos enunciados dos problemas.

3. O vídeo de apresentação deve ter duração entre 7 e 10 minutos **com a participação de todos os membros da equipe**. A seguinte organização deve ser considerada:

a) introdução sobre problema: explicar o problema;

b) explicação da solução: explicar de forma objetiva, como a equipe tratou o problema, especificamente a modularização, tipos abstratos de dados e algoritmos utilizados, preferencialmente com diagramas de fluxos ou visão geral de algoritmos em português (não mostre códigos em C na apresentação!);

c) o custo da solução adotada: explicar a variação do número de iterações em estruturas de repetições para o tamanho da entrada. Nesse sentido, gráficos, tabelas e equações matemáticas podem ser utilizados, assim como foram vistos nos conteúdos e listas de atividades sobre algoritmos de ordenação e busca.

Qualquer ferramenta para gravação de vídeo aulas ou vídeo conferências pode ser utilizada. Por exemplo, OBS, Power Point, Google Meet, Zoom, Skype dentre outros. Manuais e instruções sobre essas ferramentas são vastas em blogs da Internet.

## Lista de Problemas

### Problema 1:

Alguém deixou o quadro de medalhas das olimpíadas fora de ordem. Seu programa deve colocá-lo na ordem correta. A ordem dos países no quadro de medalhas é dada pelo número de medalhas de ouro. Se há empate em medalhas de ouro, a nação que tiver mais medalhas de prata fica a frente. Havendo empate em medalhas de ouro e prata, fica mais bem colocado o país com mais medalhas de bronze. Se dois ou mais países empatarem nos três tipos de medalhas, seu programa deve mostrá-los em ordem alfabética.

Entrada: A entrada é dada pelo número de países participantes  $N$  ( $0 \leq N \leq 500$ ) seguido pela lista dos países, com suas medalhas de ouro  $O$  ( $0 \leq O \leq 10000$ ), prata  $P$  ( $0 \leq P \leq 10000$ ) e bronze  $B$  ( $0 \leq B \leq 10000$ ).

Saída: A saída deve ser a lista de países, com suas medalhas de ouro, prata e bronze, na ordem correta do quadro de medalhas, com as nações mais premiadas aparecendo primeiro.

Exemplo de Entrada	Exemplo de Saída
8 Belgica 2 2 2 Brasil 7 6 6 Franca 10 18 14 Italia 8 12 8 Australia 8 11 10 Colombia 3 2 3 Suica 3 2 2 Tailandia 2 2 2	Franca 10 18 14 Italia 8 12 8 Australia 8 11 10 Brasil 7 6 6 Colombia 3 2 3 Suica 3 2 2 Belgica 2 2 2 Tailandia 2 2 2

### Problema 2:

Tia Joana é uma respeitada professora e tem vários alunos. Em sua última aula, ela prometeu que iria sortear um aluno para ganhar um bônus especial na nota final: ela colocou  $N$  pedaços de papel numerados de 1 a  $N$  em um saquinho e sortear um determinado número  $K$ ; o aluno premiado foi o  $K$ -ésimo aluno na lista de chamada. O problema é que a Tia Joana esqueceu o diário de classe, então ela não tem como saber qual número corresponde a qual aluno. Ela sabe os nomes de todos os alunos, e que os números deles, de 1 até  $N$ , são atribuídos de acordo com a ordem alfabética, mas os alunos

dela estão muito ansiosos e querem logo saber quem foi o vencedor. Dado os nomes dos alunos da Tia Joana e o número sorteado, determine o nome do aluno que deve receber o bônus.

Entrada: A primeira linha contém dois inteiros  $N$  e  $K$  separados por um espaço em branco ( $1 \leq K \leq N \leq 100$ ). Cada uma das  $N$  linhas seguintes contém uma cadeia de caracteres de tamanho mínimo 1 e máximo 20 representando os nomes dos alunos. Os nomes são compostos apenas por letras minúsculas de 'a' a 'z'.

Saída: Seu programa deve imprimir uma única linha, contendo o nome do aluno que deve receber o bônus.

Exemplos de Entrada	Exemplos de Saída
5 1 maria joao carlos vanessa jose	carlos

### Problema 3:

Raju e Meena adoram jogar um jogo diferente com pequenas peças de mármore, chamados Marbles. Eles têm um monte destas peças com números escritos neles. No início, Raju colocaria estes pequenos mármore um após outro em ordem ascendente de números escritos neles. Então Meena gostaria de pedir a Raju para encontrar o primeiro mármore com um certo número. Ele deveria contar 1...2...3. Raju ganha um ponto por cada resposta correta e Meena ganha um ponto se Raju falha. Depois de um número fixo de tentativas, o jogo termina e o jogador com o máximo de pontos vence. Hoje é sua chance de jogar com Raju. Sendo um/a cara esperto/a, você tem em seu favor o computador. Mas não subestime Meena, ela escreveu um programa para monitorar quanto tempo você levará para dar todas as respostas. Portanto, agora escreva o programa, que ajudará você em seu desafio com Raju.

Entrada: A entrada contém vários casos de teste, mas o total de casos é menor do que 65. Cada caso de teste inicia com dois inteiros:  $N$  que é o número de mármore e  $Q$  que é o número de consultas que Meena deseja fazer. As próximas  $N$  linhas conterão os números escritos em cada um dos  $N$  mármore. Os números destes mármore não têm qualquer ordem em particular. As seguintes  $Q$  linhas irão conter  $Q$  consultas. Tenha certeza, nenhum dos números da entrada é maior do que 10000 e nenhum deles é negativo. A entrada é terminada por um caso de teste onde  $N = 0$  e  $Q = 0$ .

Saída: Para cada caso de teste de saída deve haver um número serial do caso de teste. Para cada consulta, escreva uma linha de saída. O formato desta linha dependerá se o número consultado estiver ou não escrito em um dos mármore. Os dois diferentes formatos são descritos abaixo: 'x found at y', se o primeiro marble x foi encontrado na posição y. Posições são numeradas de 1, 2,... a  $N$ . 'x not found', se o marble com o número x não estiver presente.

Exemplo de Entrada	Exemplo de Saída
4 1 2 3 5 1 5	CASE# 1: 5 found at 4 CASE# 2: 2 not found 3 found at 3

5 2	
1	
3	
3	
3	
1	
2	
3	
0 0	

#### Problema 4:

Considerando a entrada de valores inteiros não negativos, ordene estes valores segundo o seguinte critério: primeiro os Pares, depois os Ímpares. Sendo que deverão ser apresentados os pares em ordem crescente e depois os ímpares em ordem decrescente.

Entrada: A primeira linha de entrada contém um único inteiro positivo  $N$  ( $1 < N < 10^5$ ) Este é o número de linhas de entrada que vem logo a seguir. As próximas  $N$  linhas conterão, cada uma delas, um valor inteiro não negativo.

Saída: Apresente todos os valores lidos na entrada segundo a ordem apresentada acima. Cada número deve ser impresso em uma linha, conforme exemplo abaixo.

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

#### Problema 5:

Sua tarefa é escrever um programa que, dada a informação dos países que receberam medalhas de ouro, prata e bronze em cada modalidade, gere a lista de classificação dos países na competição. Nesta tarefa, os países serão identificados por números inteiros. O melhor colocado deve ser o país que conseguiu o maior número de medalhas de ouro. Se houver empate entre países no número de medalhas de ouro, o melhor colocado entre esses é o país que conseguiu o maior número de medalhas de prata. Se houver empate também no número de medalhas de prata, o melhor colocado entre esses é o país que recebeu o maior número de medalhas de bronze. Se ainda assim houver empate entre dois países, o melhor classificado é o que tem o menor número de identificação.

Entrada: A entrada contém um único conjunto de testes. A primeira linha da entrada contém dois números inteiros  $N$  e  $M$ , separados por um espaço em branco, indicando respectivamente o número de países e número de modalidades esportivas envolvidas na competição. 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 espaço em branco, representando os 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 3, a de prata pelo país 2 e a de bronze pelo país 1.

Saída: Seu programa deve imprimir, na saída padrão, uma linha contendo N números, separados por um espaço em branco, representando os países na ordem decrescente de classificação (o primeiro número representa o país que é o primeiro colocado, o segundo número representa o país que é o segundo colocado, e assim por diante).

Exemplo de Entrada	Exemplo de Saída
2 2 2 1 2 1 2 2	2 1
4 3 3 2 1 4 3 1 4 3 1	4 3 2 1
3 3 3 1 2 2 3 1 1 2 3	1 2 3

#### Problema 6:

Joãozinho mora em uma rua que tem N casas. Marquinhos é o melhor amigo dele, mas sempre gosta de pregar peças em Joãozinho. Desta vez, ele pegou os dois brinquedos prediletos de Joãozinho e os escondeu em duas casas distintas da rua. Em compensação, Marquinhos deu uma dica importante para Joãozinho: a soma dos números das casas em que escondei teus brinquedos é igual a K. Além disso, escolhi as casas de tal forma que não existe outro par de casas cuja soma tenha esse mesmo valor. Sabendo disto, encontre qual é o par de casas em que se encontram os brinquedos de Joãozinho. Para auxiliar seu amigo, Marquinhos entregou a Joãozinho uma lista com o número das casas já em ordem crescente (isto é, do menor para o maior número).

Entrada: A primeira linha da entrada contém um número inteiro N, que representa o número de casas que existem na rua. Cada uma das N linhas seguintes contém um número inteiro, representando o número de uma casa. Note que esses N números estão ordenados, do menor para o maior. A última linha da entrada contém um inteiro K, que é a soma dos números das duas casas onde os brinquedos estão escondidos.

Saída: Se programa deve imprimir uma única linha, contendo dois inteiros, A e B,  $A < B$ , que representam os números das casas em que estão escondidos os brinquedos. Os dois números devem ser separados por um espaço em branco.

Exemplo de Entrada	Exemplo de Saída
4 1 2 3 5 8	3 5

4	2 3
1	
2	
3	
5	
5	

### Problema 7

Alice e Bob gostam de jogos. E agora eles estão prontos para começar um novo jogo. Eles colocaram  $n$  barras de chocolate em uma linha. Alice começa a comer barras de chocolate uma a uma, da esquerda para a direita, e Bob - da direita para a esquerda. Para cada barra de chocolate é conhecido o tempo necessário para o jogador consumi-la (Alice e Bob comem com a mesma velocidade). Quando o jogador consome uma barra de chocolate, ele imediatamente começa com outra. Não é permitido comer duas barras de chocolate ao mesmo tempo, deixar a barra inacabada e fazer pausas. Se os dois jogadores começarem a comer a mesma barra simultaneamente, Bob deixa isso para Alice como um verdadeiro cavalheiro. Quantas barras cada um dos jogadores vai consumir?

Entrada: A primeira linha contém um inteiro  $n$  ( $1 \leq n \leq 105$ ) - a quantidade de barras na mesa. A segunda linha contém uma sequência  $t_1, t_2, \dots, t_n$  ( $1 \leq t_i \leq 1000$ ), onde  $t_i$  é o tempo (em segundos) necessário para consumir a  $i$ -ésima barra (na ordem da esquerda para a direita).

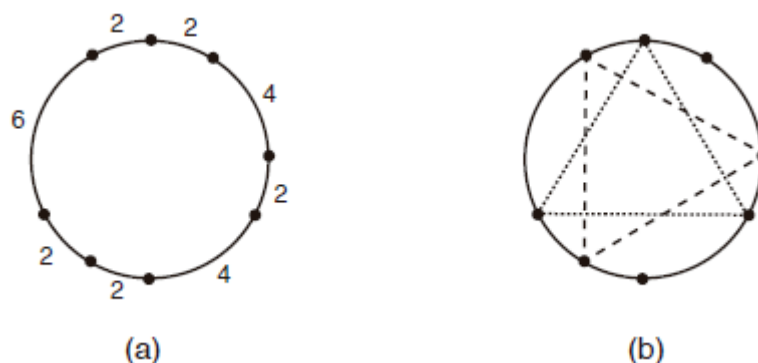
Saída: Imprima dois números "a" e "b", onde "a" é a quantidade de barras consumidas por Alice, e "b" é a quantidade de barras consumidas por Bob.

Exemplo de Entrada	Exemplo de Saída
5 2 9 8 2 7	2 3

### Problema 8

São dados  $N$  pontos em uma circunferência. Você deve escrever um programa que determine quantos triângulos equiláteros distintos podem ser construídos usando esses pontos como vértices.

A figura abaixo ilustra um exemplo; (a) mostra um conjunto de pontos, determinados pelos comprimentos dos arcos de circunferência que têm pontos adjacentes como extremos, e (b) mostra os dois triângulos que podem ser construídos com esses pontos.



Entrada: A entrada contém vários casos de teste. A primeira linha de cada caso de teste contém um número inteiro  $N$  ( $3 \leq N \leq 105$ ), o número de pontos dados. A segunda linha contém  $N$  inteiros  $X_i$  ( $1 \leq X_i \leq 103$ ) para  $1 \leq i \leq N$ , representando os comprimentos dos arcos entre dois pontos

consecutivos na circunferência: para  $1 \leq i \leq (N - 1)$ ,  $X_i$  representa o comprimento do arco entre os pontos  $i$  e  $i + 1$ ;  $X_N$  representa o comprimento do arco entre os pontos  $N$  e  $1$ . O final da entrada é determinado por EOF (fim de arquivo).

Saída: Seu programa deve produzir uma única linha para cada caso de teste, contendo um único inteiro, o número de triângulos equiláteros distintos que podem ser construídos utilizando os pontos dados como vértices.

Exemplo de Entrada	Exemplo de Saída
8 4 2 4 2 2 6 2 2 6 3 4 2 1 5 3	2 1

## Problema 9

Andrea, Carlos e Marcelo são muito amigos e passam todos os finais de semana à beira da piscina. Enquanto Andrea se bronzeia ao sol, os dois ficam jogando Bolhas. Andrea, uma cientista da computação muito esperta, já disse a eles que não entende por que passam tanto tempo jogando um jogo tão primário. Usando o computador portátil dela, os dois geram um inteiro aleatório  $N$  e uma sequência de inteiros, também aleatória, que é uma permutação de  $1, 2, \dots, N$ . O jogo então começa, cada jogador faz um movimento, e a jogada passa para o outro jogador. Marcelo é sempre o primeiro a começar a jogar. Um movimento de um jogador consiste na escolha de um par de elementos consecutivos da sequência que estejam fora de ordem e em inverter a ordem dos dois elementos. Por exemplo, dada a sequência  $1, 5, 3, 4, 2$ , o jogador pode inverter as posições de  $5$  e  $3$  ou de  $4$  e  $2$ , mas não pode inverter as posições de  $3$  e  $4$ , nem de  $5$  e  $2$ . Continuando com o exemplo, se o jogador decide inverter as posições de  $5$  e  $3$  então a nova sequência será  $1, 3, 5, 4, 2$ . Mais cedo ou mais tarde, a sequência ficará ordenada. Perde o jogador impossibilitado de fazer um movimento. Andrea, com algum desdém, sempre diz que seria mais simples jogar cara ou coroa, com o mesmo efeito. Sua missão, caso decida aceitá-la, é determinar quem ganha o jogo, dada a sequência inicial.

Entrada: A entrada contém vários casos de teste. Os dados de cada caso de teste estão numa única linha, e são inteiros separados por um espaço em branco. Cada linha contém um inteiro  $N$  ( $2 \leq N \leq 10^5$ ), seguido da sequência inicial  $P = (X_1, X_2, \dots, X_N)$  de  $N$  inteiros distintos dois a dois, onde  $1 \leq X_i \leq N$  para  $1 \leq i \leq N$ . O final da entrada é indicado por uma linha que contém apenas o número zero.

Saída: Para cada caso de teste da entrada seu programa deve imprimir uma única linha, com o nome do vencedor, igual a Carlos ou Marcelo., sem espaços em branco.

Exemplo de Entrada	Exemplo de Saída
5 1 5 3 4 2	Marcelo
5 5 1 3 4 2	Carlos
5 1 2 3 4 5	Carlos
6 3 5 2 1 4 6	Carlos
5 5 4 3 2 1	Carlos
6 6 5 4 3 2 1	Marcelo
0	

## Problema 10

Vasya, estudante do ensino médio, ganhou uma string de comprimento  $n$  como presente de aniversário. Esta string consiste apenas nas letras 'a' e 'b'. Vasya denota a beleza da string como o comprimento máximo de uma substring (subsequência consecutiva) consistindo em letras iguais. Vasya não pode alterar mais do que  $k$  caracteres da string original. Qual é a beleza máxima da corda que ele pode alcançar?

Entrada: A primeira linha da entrada contém dois inteiros  $n$  e  $k$  ( $1 \leq n \leq 100\,000$ ,  $0 \leq k \leq n$ ) - o comprimento da string e o número máximo de caracteres a serem alterados. A segunda linha contém a string, consistindo apenas nas letras 'a' e 'b'.

Saída: Imprima o único inteiro - a beleza máxima da string que Vasya pode alcançar alterando não mais do que  $k$  caracteres.

Exemplo de Entrada	Exemplo de Saída
4 2 Abba	4
8 1 Aabaabaa	5