



Instituto de Informática – INF  
Cursos de Bacharelado do INF/UFV (NBC)

## Avaliação

Disciplina:	Algoritmos & Estruturas de Dados – 1	
Professor:	Nádia Félix Felipe da Silva	Nota:
Aluno(a):		Matrícula:
Primeira Avaliação Formal – 2023/2		Data: 30/11/2023

**Atenção:** Você pode resolver qualquer quantidade de questões para obter os 10,0 (dez pontos) desta avaliação.

**Lembre-se:** A resolução das questões devem ser submetidas ao *Sharif Judge System* (SJS) utilizado na disciplina <<https://sharif.inf.ufv.br/nadia>> até às 23h59min do dia 02 de dezembro de 2023 (sábado).

Os “*casos de teste*” deste ambiente o(a) auxiliarão no processo de resolução da avaliação, mas os códigos-fontes poderão ser analisados pelo docente responsável e a nota aferida pelo SJS ser alterada, tanto para *maior* quanto para *menor*.

**Questão 01 (5,0 pontos)** Um(a) estudante de programação está estudando o conceito de *recursividade* e, por isso, resolveu desafiar-se durante o período de carnaval de 2022: implementar código recursivo para o seguinte problema:

“Escrever, utilizando a linguagem de programação  $\mathbb{C}$ , um programa que seja capaz de receber como entrada três números naturais ( $n$ ,  $x$  e  $y$ ), necessariamente nesta ordem e numa única linha, e imprimir uma *matriz moldura* de ordem  $n \times n$ , sendo  $x$  o valor da moldura externa e  $y$  o valor da moldura interna [veja a explicação do conceito adiante neste texto].

Sabe-se que os valores obedecem às seguintes regras:

- $1 \leq n \leq 100$ ;
- $1 \leq x, y < 10$  e  $x \neq y$ ;
- $n$  é sempre um número ímpar.

**Observação:**

Uma *matriz moldura* de ordem  $n \times n$ , com valores  $x$  (moldura mais externa) e  $y$  (moldura interna) é uma matriz com a seguinte forma geral:

	1	2	3	...	i	...	(n-2)	(n-1)	n
1	x	x	x	x	x	x	x	x	x
2	x	y	y	y	y	y	y	y	x
3	x	y	x	x	x	x	x	y	x
...	x	y	x	...	...	...	x	y	x
i	x	y	x	...	y	...	x	y	x
...	x	y	x	...	...	...	x	y	x
(n-2)	x	y	x	x	x	x	x	y	x
(n-1)	x	y	y	y	y	y	y	y	x
n	x	x	x	x	x	x	x	x	x

Por exemplo, se tivermos que  $n = 7$ ,  $x = 1$  e  $y = 0$ , então a *matriz moldura* resultante será a mostrada a seguir:

1	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	1	1	1	0	1
1	0	1	0	1	0	1
1	0	1	1	1	0	1
1	0	0	0	0	0	1
1	1	1	1	1	1	1

Por outro lado, se  $n = 5$ ,  $x = 2$  e  $y = 0$ , tem-se:

2	2	2	2	2
2	0	0	0	2
2	0	2	0	2
2	0	0	0	2
2	2	2	2	2

Ou, ainda, se  $n = 11$ ,  $x = 1$  e  $y = 2$ , tem-se que:

1	1	1	1	1	1	1	1	1	1	1
1	2	2	2	2	2	2	2	2	2	1
1	2	1	1	1	1	1	1	1	2	1
1	2	1	2	2	2	2	2	1	2	1
1	2	1	2	1	1	1	2	1	2	1
1	2	1	2	1	2	1	2	1	2	1
1	2	1	2	1	1	1	2	1	2	1
1	2	1	2	2	2	2	2	1	2	1
1	2	1	1	1	1	1	1	1	2	1
1	2	2	2	2	2	2	2	2	2	1
1	1	1	1	1	1	1	1	1	1	1

**Entrada:** Numa única linha, os valores de  $n$ ,  $x$  e  $y$ , nesta ordem, e separados entre si por um único espaço em branco.

**Saída:** A *matriz moldura* conforme apresentada anteriormente, ou seja, com  $n$  linhas

e  $n$  colunas por linha, com elementos da linha sempre separados por um único espaço em branco entre si. As “barras verticais” e a “linhas horizontais” não devem ser impressas, apenas os números da matriz.

**Observação:** Para ser considerada completamente válida, a solução tem que utilizar o conceito de *recursividade*. A resolução correta mas sem o uso de *recursividade* implicará na atribuição de metade de nota, ou seja, 2,5 (dois pontos e meio).

**Questão 02 (5,0 pontos).** Considere que é dado um vetor  $A$  contendo  $n$  números naturais, com  $n \in \mathbb{N}^*$  e  $n \leq 10^4$ .

Construa um programa  $\mathbb{C}$  que seja capaz de determinar, e imprimir, qual é o número de pares  $(i, j)$ , com  $i < j$ , cujo valor  $(A[i] + A[j])$  é máximo.

**Entrada:** Na primeira linha, tem-se o número de casos de teste,  $t \in \mathbb{N}^*$  e  $t \leq 10$ .

Cada par de linhas seguinte representa um caso de teste, sendo que a primeira linha do par contém o número  $n$ , ou seja, o número de elementos do vetor  $A$  e a segunda linha do par contém os valores dos elementos separados, entre si, por um único espaço em branco e fornecidos do primeiro elemento ao último elemento (1º, 2º, 3º, ...,  $n$ -ésimo).

**Saída:** Contém, em cada linha, o número de pares distintos que atendem à regra estabelecida anteriormente.

**Observação:** Sabe-se que os elementos do vetor  $A$  são menores ou iguais a  $10^4$ .

**Exemplo 01:**

ENTRADA	SAÍDA
3	1
4	2
1 2 3 4	10
3	
1 2 1	
5	
1 1 1 1 1	

**Explicação** para as saídas geradas:

**1o teste :** Os pares possíveis são os seguintes: (1,2); (1,3); (1,4); (2,3); (2,4) e (3,4). Somando-se os respectivos valores de  $A[i]$  e  $A[j]$  tem-se, respectivamente: 3, 4, 5, 5, 6 e 7. A soma máxima tem valor igual a 7, e corresponde à soma de  $A[3]$  com  $A[4]$ . Portanto, tem-se um único valor máximo e, por isso, a saída é igual a 1;

**2o teste :** Os pares possíveis são os seguintes: (1,2); (1,3) e (2,3). Somando-se os respectivos valores de  $A[i]$  e  $A[j]$  tem-se, respectivamente: 3, 2 e 3. A soma máxima tem valor igual a 3, e corresponde à soma de  $A[1]$  com  $A[2]$  e de  $A[2]$  com  $A[3]$ . Portanto, o valor máximo é alcançado duas vezes e, por isso, a saída é igual 2.

**3o teste :** Os pares possíveis são os seguintes: (1,2); (1,3); (1,4); (1,5); (2,3); (2,4); (2,5); (3,4); (3,5) e (4,5). Somando-se os respectivos valores de  $A[i]$  e  $A[j]$  tem-se que todas as somas resultam em 2. Logo a soma máxima tem valor igual a 2 e ocorre 10 vezes. Por isso, a saída é igual a 10.

**Exemplo 02:**

ENTRADA	SAÍDA
2	1
4	1
4 3 2 1	
6	
1 2 3 3 2 1	

**Exemplo 03:**

ENTRADA	SAÍDA
2	28
8	21
1 1 1 1 1 1 1 1	
7	
100 100 100 100 100 100 100	

**Questão 03 (5,0 pontos).** Considere um vetor de números naturais  $A$ , cujo tamanho é expresso por  $n \in \mathbb{N}^*$ . Os números naturais são distribuídos em  $A$  sem qualquer ordem aparente, ou seja, parecem ser totalmente aleatórios, inclusive com a possibilidade de repetição de valores.

Cada elemento do  $A$  corresponde à *ordem* (ou seja, posição numa determinada sequência) de um certo número primo, onde 2 é o número primo de ordem 1, 3 é o número primo de ordem 2, 5 é o número primo de ordem 3, e assim sucessivamente.

Um programa de computador, escrito no linguagem  $\mathbb{C}$ , deve ser desenvolvido para imprimir os números primos correspondentes à ordem indicada nos elementos de  $A$ , com a impressão ocorrendo em ordem ascendente em relação aos valores contidos em  $A$ .

**Entrada:** A primeira linha da entrada contém o valor de  $n$ , ( $1 \leq n \leq 100$ ).

A segunda linha contém os valores dos elementos de  $A$ , sempre mostrados do primeiro elemento até o último elemento, onde  $0 < v[i] \leq 10^4$ .

**Saída:** Deve mostrar, na primeira linha, os números primos correspondentes aos elementos de  $A$  em ordem ascendente e, na segunda linha, a soma destes números.

Tabela 1: Questão 02 – Exemplos de entrada/saída.

INPUT	OUTPUT
10 10 2 7 8 5 4 3 1 6 9	2 3 5 7 11 13 17 19 23 29 129
INPUT	OUTPUT
7 35 140 18 150 28 59 15	47 61 107 149 277 809 863 2313
INPUT	OUTPUT
5 295 354 178 444 785	1061 1933 2383 3119 6011 14507
INPUT	OUTPUT
5 787 1032 3720 5000 9	23 6037 8231 34847 48611 97749
INPUT	OUTPUT
5 1 2 3 3 4	2 3 5 5 7 22

**Questão 04 (5,0 pontos)** Ada, uma programadora de jogos digitais educacionais, deseja criar um novo jogo que envolva palavras de diferentes idiomas, como Português e Inglês, por exemplo.

Para isto uma de suas necessidades é ter uma rotina que seja capaz de, dada uma cadeia de caracteres  $S$  a ela, gerar “*todos os possíveis anagramas distintos daquela cadeia*”.

Por exemplo, se a rotina receber a cadeia  $S = \text{CAB}$ , ela deve gerar:

CAB  
CBA  
ACB  
ABC  
BAC  
BCA

Só que Ada tem três restrições para a elaboração da solução:

1. A rotina de geração deve ser, necessariamente, *recursiva*;
2. A impressão das cadeias-resposta deve ser realizada com uma cadeia por linha, sendo estas apresentadas em ordem alfabética crescente.<sup>1</sup>;
3. Não pode haver repetição de cadeias na impressão, ou seja, se houver a geração de cadeias idênticas, apenas uma delas deve ser impressa.

Você, na condição de estagiário(a) da equipe de Ada, foi incumbido elaborar, utilizando a linguagem de programação  $\mathbb{C}$ , esta solução.

**Atenção:**

1. Note que a cadeia  $S$  pode ser repetição de símbolos e, por isso, alguns anagramas serão iguais e devem ser excluídos da impressão. Por exemplo, se  $S = \text{ABA}$  apenas deverão ser impressas as cadeias:

---

<sup>1</sup>Ou seja: a lista anteriormente mostrada, apesar de apresentar todos os possíveis anagramas distintos, não seria aceita, pois não está em ordem alfabética crescente.

AAB

ABA

BAA

2. Considere, para simplificação da solução, que a cadeia de caracteres  $S$  conterá apenas letras maiúsculas, dígitos, sinais de pontuação e o caractere “espaço em branco”. Não conterá, portanto, letras acentuadas, sinais gráficos, etc.

**Entrada:** Numa única linha, a cadeia  $S$ , sabendo-se que  $|S| \leq 256$ .

**Saída:** Todos os possíveis anagramas distintos de  $S$ , com um por linha, na ordem previamente estabelecida.

**Observação:** Para ser considerada completamente válida, a solução tem que utilizar o conceito de *recursividade*. A resolução correta mas sem o uso de *recursividade* implicará na atribuição de metade de nota, ou seja, 2,5 (dois pontos e meio).