



ESTRUTURAS DE DADOS

EXERCÍCIOS DE IMPLEMENTAÇÃO, CONCEITO E ALGORITMOS

LISTA DE EXERCÍCIOS

ENGENHARIA DE COMPUTAÇÃO

TIPOS ABSTRATOS DE DADOS

- 1 Implemente um TAD **Ponto** que represente um ponto no plano cartesiano com coordenadas (x, y) .

O TAD deve fornecer as seguintes operações:

- Criar um ponto com coordenadas x e y
- Obter as coordenadas x e y
- Calcular a distância do ponto até a origem $(0, 0)$
- Calcular a distância entre dois pontos
- Exibir o ponto no formato “ (x, y) ”

Este exercício introduz conceitos fundamentais de TAD: encapsulamento de dados e operações associadas.

Exemplo de uso

```
Ponto p1 = criarPonto(3, 4);  
Ponto p2 = criarPonto(0, 0);
```

```
Distância até origem: 5.00  
Distância entre p1 e p2: 5.00  
Coordenadas: (3, 4)
```

- ② Implemente um TAD **Data** que represente uma data com dia, mês e ano.

O TAD deve fornecer as seguintes operações:

- Criar uma data (com validação de data válida)
- Verificar se o ano é bissexto
- Comparar duas datas (anterior, igual ou posterior)
- Calcular a diferença em dias entre duas datas
- Adicionar N dias a uma data
- Exibir a data no formato “DD/MM/AAAA”

Este exercício trabalha validação, lógica de calendário e operações aritméticas.

Exemplo de uso

```
Data d1 = criarData(15, 3, 2024);  
Data d2 = criarData(20, 3, 2024);
```

Data válida: true

Bissexto: true

Comparação: d1 é anterior a d2

Diferença: 5 dias

d1 + 10 dias: 25/03/2024

- 3 Implemente um TAD Conjunto que represente um conjunto matemático de inteiros.

O TAD deve fornecer as seguintes operações:

- Criar conjunto vazio
- Inserir elemento (sem repetições)
- Remover elemento
- Verificar se elemento pertence ao conjunto
- Obter tamanho do conjunto
- União de dois conjuntos
- Interseção de dois conjuntos
- Diferença entre dois conjuntos
- Verificar se um conjunto é subconjunto de outro
- Exibir elementos do conjunto

Este exercício trabalha estruturas dinâmicas e operações matemáticas sobre coleções.

Exemplo de uso

Conjunto A = {1, 2, 3, 4}

Conjunto B = {3, 4, 5, 6}

$A \cup B = \{1, 2, 3, 4, 5, 6\}$

$A \cap B = \{3, 4\}$

$A - B = \{1, 2\}$

$3 \in A$: true

$A \subseteq B$: false

- 4 Implemente um TAD **MatrizEsparsa** que represente uma matriz esparsa (com muitos elementos zero) de forma eficiente.

O TAD deve armazenar apenas elementos não-nulos usando lista encadeada e fornecer:

- Criar matriz esparsa com dimensões $M \times N$
- Inserir elemento em posição (i, j)
- Acessar elemento em posição (i, j)
- Somar duas matrizes esparsas
- Multiplicar duas matrizes esparsas
- Transpor a matriz
- Exibir matriz completa (incluindo zeros)

Este exercício avançado trabalha eficiência de memória, estruturas encadeadas e algoritmos de matriz.

Exemplo de uso

```
MatrizEsparsa M1(5, 5);  
M1.inserir(0, 0, 10);  
M1.inserir(2, 3, 20);  
M1.inserir(4, 4, 30);
```

Elementos armazenados: 3 (de 25 possíveis)

```
M1[2][3] = 20  
M1[1][1] = 0
```

Matriz transposta:

```
10  0  0  0  0  
 0  0  0  0  0  
 0  0  0 20  0  
 0  0  0  0  0  
 0  0  0  0 30
```

LISTAS E ARRAYS

FILAS

EXERCÍCIOS DE ENADE

Ano	#	Curso	Conteúdo
2011	22	Computação	Lista Encadeada com sentinela
2011	23	Computação	Filas e Pilhas
2011	24	Computação	Arrays e busca binária
2011	25	Computação	Árvores Binárias de Busca
2011	27	Computação	Grafos: Matriz vs Lista de Adjacência
2014	10	Ciência da Computação	TAD e encapsulamento
2014	11	Ciência da Computação	Análise de custo: Listas vs Arrays
2014	12	Ciência da Computação	Fila de Prioridade e Heap
2014	13	Ciência da Computação	Árvores Balanceadas
2014	14	Ciência da Computação	Ordenação: QuickSort, MergeSort, HeapSort
2014	D03	Ciência da Computação	Backtracking
2014	D04	Ciência da Computação	Grafos: Algoritmo de Dijkstra
2017	22	Ciência da Computação	Arrays em Algoritmos Gulosos
2017	33	Ciência da Computação	Complexidade do MergeSort
2019	09	Engenharia da Computação	MergeSort com Arrays
2019	31	Engenharia da Computação	Filas em Sistemas Operacionais
2021	09	Ciência da Computação	Filas, Listas e Pilhas
2021	20	Ciência da Computação	Busca em Vetores

Ano	#	Curso	Conteúdo
2021	23	Ciência da Computação	Árvore Binária de Busca
2021	32	Ciência da Computação	Ordenação: comparações
2021	34	Ciência da Computação	Grafos e Dijkstra
2021	D05	Ciência da Computação	Heap Binário em Vetor
2023	18	Engenharia da Computação	Vetores de Tamanho Dinâmico
2023	33	Engenharia da Computação	Filas e Pilhas em Automação
2023	D02	Engenharia da Computação	QuickSort e consolidação de dados

EXERCÍCIOS SABOR ENADE