

The background is a deep blue gradient with a subtle pattern of white dots. Overlaid on this are several white geometric elements: a large circular scale on the left with degree markings from 150 to 260, and several concentric circles with arrows indicating clockwise rotation, some solid and some dashed.

# PROGRAMAÇÃO ORIENTADA A OBJETOS

PROF. EDSON MORENO

BASEADO NO MATERIAL ELABORADO PELO PROF. JÚLIO MACHADO

# TADS E ARRAY LISTS



# ARRANJOS/ARRAYS

- Arranjos/Arrays são estruturas básicas de qualquer linguagem de programação
  - Correspondem a uma área contígua de memória que pode ser usada para armazenar uma coleção de elementos do mesmo tipo
  - Todas as posições de um arranjo podem ser acessadas através de um índice (tanto para leitura como para alteração)
  - Exemplo em Java:

```
int[] arr = new int[10];  
arr[3] = 4;  
System.out.println(arr[3]);
```

# TIPOS ABSTRATOS DE DADOS (1)

Um tipo abstrato de dados (TAD) é um tipo que pode ser definido por suas operações



Uma “lista” pode ser entendida como um tipo abstrato de dados. Suas operações poderiam ser:

Inserir um elemento no final da lista	Inserir um elemento numa posição qualquer da lista	Retornar a quantidade de elementos na lista	Iterar sobre a lista	Recuperar o i-ésimo elemento da lista	etc
---------------------------------------	--	---	----------------------	---------------------------------------	-----

## TIPOS ABSTRATOS DE DADOS (2)



Tipos abstratos de dados podem ser implementados em Java usando classes.



São implementados usando os tipos básicos da linguagem (incluindo arrays) mas escondem a implementação (encapsulamento) deixando visíveis apenas as operações públicas



O uso de TADs evita que tenhamos de reimplementar o controle sobre o tipo abstrato de dados toda vez que queremos explorar esse conceito.

# EXEMPLO: TAD LISTAINT

- A coleção ListaInt representa uma lista de números inteiros de capacidade limitada
- Ver código-fonte



# O ARRAYLIST DE JAVA



A classe “ArrayList” é uma das implementações oferecidas pela API de coleções de Java para o TAD “lista”.



Implementa todas as operações básicas deste TAD (inserção, consulta, remoção, etc)



Consulte a documentação da API para obter a lista completa de métodos em <https://docs.oracle.com/en/java/javase/18/docs/api/index.html>

# GENÉRICOS

A classe ArrayList é “genérica”.

Isso significa que podemos indicar o tipo de dados da lista através de um parâmetro de tipo. Exemplos:

- `ArrayList<Integer> lstInt = new ArrayList<>();`
- `ArrayList<Veiculo> lstVeic = new ArrayList<>();`
- `ArrayList<String> lstStr = new ArrayList<>();`

Tipos genéricos serão estudados mais adiante em maiores detalhes.