

LISTAS SEQUENCIAIS

Prof. Alberto Costa Neto

TIPOS DE DADOS

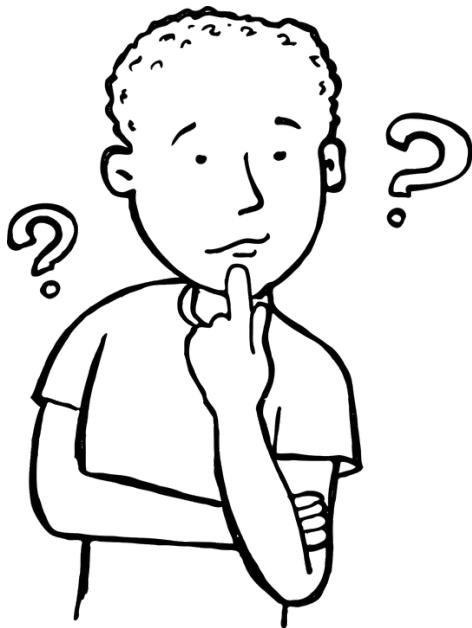
- Tipos de dados básicos
 - Inteiro, ponto flutuante, caractere.
- Tipos de dados estruturados
 - Arranjo e registro.
- Tipos de dados definidos pelo usuário
 - Tipos de dados estruturados
 - Construídos hierarquicamente através de componentes de tipos
 - Básicos
 - Estruturados
 - Definidos pelo usuário

TIPOS ABSTRATOS DE DADOS (TAD)

- ED's capazes de representar tipos de dados não previstos
- Dividem-se em duas partes:
 - Dados
 - Operações
- Característica principal
 - Separação entre conceito e implementação

NOSSO PRIMEIRO TAD: LISTA

Quais operações teria uma lista?



NOSSO PRIMEIRO TAD: LISTA

Quais operações teria uma lista?

- Inserir
- Remover
- Alterar
- Buscar
- Tamanho
- Vazia
- Cheia
- Limpar
- ...



LISTA SEQUENCIAL

- Uma opção de implementação para o TAD Lista.
- Utiliza uma estrutura sequencial, como um arranjo (array) ou arquivo.
- Itens da ED armazenados em posições contíguas de memória.

NÓ DA LISTA

- Cada item da lista é comumente chamado de **Nó**
- É possível que o Nó seja:
 - Um **tipo de dados primitivo**: número ou caractere, por exemplo.
 - Um **tipo de dados composto heterogêneo** (struct, record, classe).

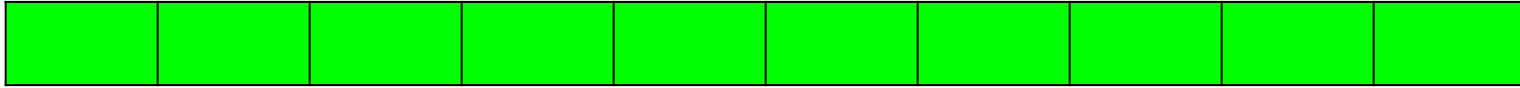
INSERÇÃO

OPERAÇÃO DE INSERÇÃO

- A inserção normalmente é feita na **primeira posição livre** do arranjo, geralmente começando da posição 0 (zero).
- Deve-se controlar a quantidade de itens (nós) na lista, ou seja, seu **tamanho**.
- Como o arranjo tem um tamanho fixo (**capacidade**), é possível que fique cheio em algum momento.

OPERAÇÃO DE INSERÇÃO

Inicialmente o arranjo está vazio (Tamanho = 0)



OPERAÇÃO DE INSERÇÃO

Inicialmente o arranjo está vazio (Tamanho = 0)

--	--	--	--	--	--	--	--	--	--

Inserindo um item (nó) com valor 10 (Tamanho = 1)

10									
----	--	--	--	--	--	--	--	--	--

Inserindo um item (nó) com valor 5 (Tamanho = 2)

10	5								
----	---	--	--	--	--	--	--	--	--

Inserindo itens com valores 2,3,4,6,7,8,9,1 (Tamanho = 10)

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---

OPERAÇÃO DE INSERÇÃO

Quando Tamanho = Capacidade, a Lista está cheia.

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---

O que fazer neste caso?



OPERAÇÃO DE INSERÇÃO

Quando Tamanho = Capacidade, a Lista está cheia.

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---

O que fazer neste caso?

- Realocar os dados para um arranjo maior (se a linguagem de programação suportar)
- Não aceitar mais inserções
- Criar uma lista maior e copiar os dados



BUSCA

OPERAÇÃO DE BUSCA

- Procurar a posição de um item na lista
- Inicia-se pela posição 0 (zero) e segue-se até:
 - Encontrar um item e encerrar a procura.
 - Chegar ao último item sem encontrar.
- Quando o item procurado é encontrado, seu valor é retornado.
- Este algoritmo é conhecido como **Busca Sequencial**.

ALTERAÇÃO

OPERAÇÃO DE ALTERAÇÃO

- Consiste em alterar o item contido numa dada posição da lista.
- A posição deve ser dentro da faixa de posições preenchidas.
- O acesso à posição é direto, por se tratar de um array.
- O item contido naquela posição da lista é substituído pelo valor do novo item.

REMOÇÃO

OPERAÇÃO DE REMOÇÃO

- Consiste em remover um item da lista.
- A procura se inicia pela posição 0 (zero) e segue até:
 - Encontrar um item igual e encerrar a procura.
 - Chegar ao último item sem encontrar.
- Quando o item procurado é encontrado, seu valor é substituído pelo valor à sua direita (se houver).
 - Esta operação deve ser repetida para todos os elementos à direita.
- O tamanho da lista é decrementado em 1 (um).

OPERAÇÃO DE REMOÇÃO

- Considerando a lista abaixo, ao se tentar remover o 1, o que acontece?

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---



OPERAÇÃO DE REMOÇÃO (CASO SIMPLES)

- Considerando a lista abaixo, ao se tentar remover o 1, o que acontece?

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---

Este é o caso mais simples, pois basta colocar a posição como disponível novamente (Tamanho = 9).

10	5	2	3	4	6	7	8	9	
----	---	---	---	---	---	---	---	---	--

OPERAÇÃO DE REMOÇÃO (CASO COMPLEXO)

- Considerando a lista abaixo, ao se tentar remover o 8, o que acontece?

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---




OPERAÇÃO DE REMOÇÃO (CASO COMPLEXO)

- Considerando a lista abaixo, ao se tentar remover o 8, o que acontece?

10	5	2	3	4	6	7	8	9	1
----	---	---	---	---	---	---	---	---	---


Temos que colocar o 9 no lugar do 8.

10	5	2	3	4	6	7	9	9	1
----	---	---	---	---	---	---	---	---	---



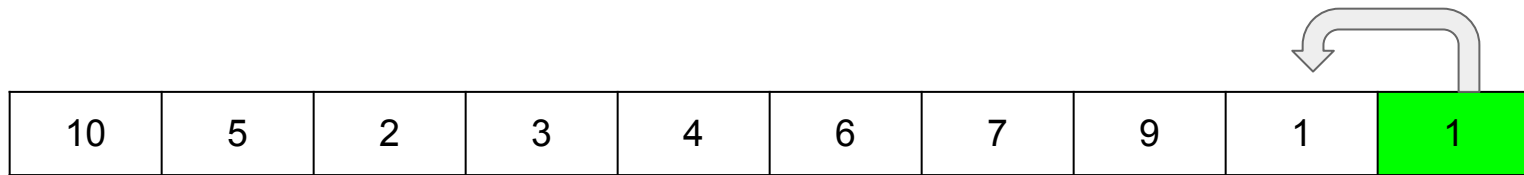
E o mesmo em relação ao 1

10	5	2	3	4	6	7	9	1	1
----	---	---	---	---	---	---	---	---	---



OPERAÇÃO DE REMOÇÃO (CASO COMPLEXO)

- Note que o 1 não é apenas removido "logicamente" do array, ou seja, não deve mais ser acessado.



SUGESTÕES DE ESTUDO

Estruturas de Dados (Nina Edelweiss)

- Capítulo 2
- Seções 3.1 a 3.3

Projeto de Algoritmos com implementações em Java e C++ (Nivio Ziviani)

- Seções 1.1, 1.2, 3.1 e 5.1

Estruturas de dados (Paulo Veloso)

- Seções 5.1 a 5.4 e 9.1