

Estrutura de Dados

Aula 4 : Listas

Prof. MSc. Fausto Sampaio

https://github.com/Fausto14/estrutura_de_dados

Centro Universitário UniFanor - Wyden

12 de novembro de 2019

1 Listas

- Introdução
- Conceito

2 Operações

- Operações Básicas

3 Lista Sequencial Estática

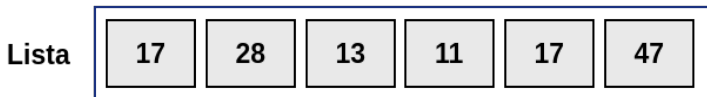
- Definição
- Vantagens
- Desvantagens
- Quando utilizar
- Implementação

4 Referências

Listas

- Lista é algo muito comum para as pessoas;
- Trata-se de um relação finita de itens, todos eles contidos dentro de um mesmo tema;
- Vários são os exemplos possíveis de listas:
 - itens em estoque em uma empresa;
 - dias da semana;
 - lista de compras do supermercado;
 - convidados de uma festa;
 - etc.

- Na computação, uma lista é uma estrutura de dados linear utilizada para armazenar e organizar dados em um computador;
- Uma estrutura do tipo lista é uma sequência de elementos do mesmo tipo;
- Seus elementos possuem estrutura interna abstraída, ou seja, sua complexidade é arbitrária e não afeta o seu funcionamento.
- Além disso, uma lista pode possuir elementos repetidos, assim como ser ordenada ou não, dependendo da aplicação.



- Como nas listas que conhecemos, a estrutura do tipo lista pode possuir N ($N \geq 0$) elementos ou itens.
- Se $N = 0$, dizemos que a lista está vazia.
- Aplicações:
 - Cadastro de Funcionários;
 - Itens em estoque;
 - Cartas de baralho;
 - etc

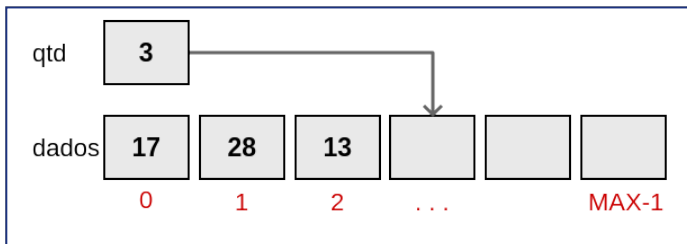
Operações

- Em uma Lista podemos realizar as seguintes operações básicas:
 - criação da Lista;
 - inserção de um elemento;
 - exclusão de um elemento;
 - acesso a um elemento;
 - sobrepor um elemento;
 - imprimir ou listar os elementos;
 - destruição da lista;
 - etc
- Essas operações dependem do tipo de alocação de memória usada:
 - estática;
 - dinâmica;

Alocação Estática

- O espaço de memória é alocado no momento da compilação;
- Exige a definição do número máximo de elementos da Lista;
- Acesso sequencial: elementos consecutivos na memória;

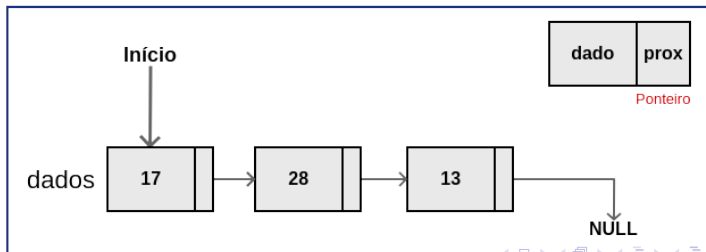
Lista - Estática



Alocação Dinâmica

- O espaço de memória é alocado em tempo de execução;
- A Lista cresce à medida que novos elementos são armazenados, e diminui à medida que elementos são removidos;
- Acesso encadeado: cada elemento pode estar em uma área distinta da memória;
- Para acessar um elemento, é preciso percorrer todos os antecessores na Lista.
- O acesso é através de ponteiros! Não tem índices!

Lista - Dinâmica

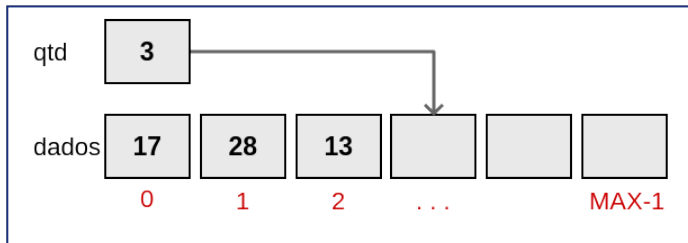


Lista Sequencial Estática

Definição

- **Lista Sequencial Estática** ou **Lista Linear Estática**;
- Tipo de Lista onde o sucessor de um elemento ocupa a posição física seguinte do mesmo;
- Uso de vetores (array).

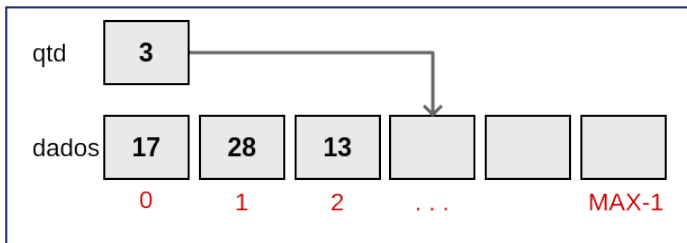
Lista - Estática



Vantagens do uso de vetores

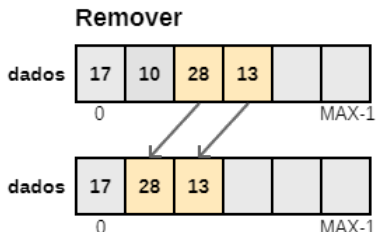
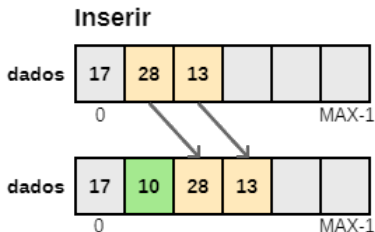
- Acesso rápido e direto aos elementos (índice);
- Tempo constante para acessar um elemento;
- Facilidade em modificar informações.

Lista - Estática



Desvantagens do uso de vetores

- Definição prévia do tamanho do vetor;
- Dificuldade para inserir e remover um elemento entre outros dois: é necessário deslocar os elementos;
- Facilidade em modificar informações.



Quando utilizar

- Listas pequenas;
- Inserção/Remoção apenas no final da lista;
- Tamanho máximo bem definido;
- A busca é operação frequente.

Implementação em C

- Modularização: ListaSequencia.h e ListaSequencial.c

ListaSequencia.h

- os protótipos das funções;
- o tipo de dado armazenado na lista;
- o ponteiro "lista";
- tamanho do vetor usado na lista.

ListaSequencia.c

- os protótipos das funções;
- o tipo de dado "lista";
- implementar as suas funções.

- Ver implementação em C.

Referências

- André Ricardo Backes, CAPÍTULO 5 - Listas, Editor(s): André Ricardo Backes, **Estrutura de Dados Descomplicada em Linguagem C**, Elsevier Editora Ltda., 2016, Pages 77-191, ISBN 9788535285239.

