

Estrutura de Dados

Prof. Orlando Saraiva Júnior
orlando.saraiva@unesp.br

""A program is like a poem:
you cannot write a poem
without writing it."

E. W. Dijkstra

Estrutura de Dados

Objetivo da aula

Conhecer Listas

- Tipos de Listas

Listas em C++

Listas em Python

Uma **lista** é um conjunto de itens. Existem vários tipos de implementação para listas como estruturas de dados:

- Listas Lineares
- Listas ordenadas
- Listas ligadas
- Listas duplamente ligadas
- Listas FIFO, ou filas.
- Listas LIFO, ou pilhas .

Uma **lista linear** é uma sequência de itens $L:[x_1, x_2, \dots, x_n]$, cujas propriedades estruturais dependem apenas da posição relativa de seus itens.

Se $n=0$, dizemos que a lista L é vazia, do contrário, valem as seguintes propriedades:

- X_1 é o primeiro item de L ;
- X_n é o último item de L ;
- X_i é precedido por x_{i-1} e seguido por x_{i+1} em L , para $1 < i < n$;

Uma **lista ordenada** $L:[a_1, a_2, \dots, a_n]$ é uma lista de dados auto ajustável, no sentido de sempre se manter ordenada após a inserção ou remoção.

É uma lista linear tal que, sendo $n > 1$, temos:

$a_1 \leq a_k$, para qualquer $1 < k < n$;

$a_k \leq a_n$, para qualquer $1 \leq k < n$;

$a_{k-1} \leq a_k \leq a_{k+1}$, para qualquer $1 < k < n$;

Se L é uma lista ordenada, podemos garantir que nenhum elemento em L é inferior a a_1 ou superior a a_n . Além disto, tomado um elemento qualquer no meio da lista, nenhum elemento à sua esquerda o supera e nenhum elemento à sua direita é inferior a ele.

Formalmente, uma lista é um tipo de dado abstrato (ADT) tal que uma instância L (de *list*) suporta as seguintes funcionalidades:

$L.\text{ins}(\text{elem}) \rightarrow$ Adiciona o elemento elem a lista L de forma ordenada;

$L.\text{rem}(\text{elem}) \rightarrow$ Remove o elemento elem da lista L ;

$L.\text{find}(\text{elem}) \rightarrow$ Encontrar o elemento elem na lista L ;

Entendendo Lista Ordenada

Operação	Valor de retorno	Conteúdo da lista
L.ins(5)	-	[5]
L.ins(3)	-	[3, 5]
L.ins(7)	-	[3, 5, 7]
L.ins(9)	-	[3, 5, 7 ,9]
L.ins(1)	-	[1, 3, 5, 7 ,9]
L.find(3)	2	[1, 3, 5, 7 ,9]
L.rem(1)	1	[3, 5, 7 ,9]
L.rem(5)	5	[3, 7 ,9]
L.rem(7)	7	[3, 9]
L.ins(2)	-	[2, 3, 9]
L.ins(8)	-	[2, 3, 8, 9]
L.ins(10)	-	[2, 3, 8, 9, 10]
L.ins(1)	-	[1, 2, 3, 8, 9, 10]

Uma **lista encadeada** é uma estrutura que mantém uma coleção de itens em ordem linear sem, no entanto, exigir que eles ocupem posições consecutivas na memória.

Para manter a ordem lógica dos itens, que podem estar espalhados por toda a memória, a lista encadeada usa blocos de memória, também denominados **nós**. Cada bloco é composto por dois campos, sendo um para guardar o item (**campo dado**) e outro para guardar o endereço do bloco que contém o próximo item (**campo de ligação**).

Hora do código

Lista Linear Sequencial

Lista Linear Sequencial

Código: lista_linear_sequencial.cpp

Funções implementadas:

void inicializarLista(LISTA* lista)

void exibirLista(LISTA* lista)

int buscaSequencial(LISTA* lista, REGISTRO registro)

bool inserirElemLista(LISTA* lista, REGISTRO registro, int i)

bool excluirElemLista(LISTA* lista, REGISTRO registro)

Lista Linear Ordenada

Lista Linear Ordenada

Código: lista_linear_ordenada.cpp

Funções implementadas:

void inicializarLista(LISTA* lista)

Int tamanho(LISTA* l)

void exibirLista(LISTA* lista)

int buscaBinaria(LISTA* lista, REGISTRO registro)

bool inserirElemListaOrd(LISTA* lista, REGISTRO reg)

bool excluirElemListaOrd(LISTA* lista, REGISTRO registro)

Benchmark

Busca Sequencial x Busca Binária



Código: lista_benchmark.cpp

Qual o método de busca é mais eficiente: busca binária ou busca sequencial ?

Lista Ligada

Lista Ligada

Código: lista_ligada.cpp

Neste código, um nó é composto por um inteiro (campo de dado) e um ponteiro para outro nó (campo de ligação).

As buscas são feitas de forma sequencial.

Implementando lista com Python

Python possui ***list*** como um dos tipos *built-in* para tipos sequencia.

Na documentação oficial da linguagem, pode-se observar os métodos prontos para uso de lista.

<https://docs.python.org/3/library/stdtypes.html?highlight=list>

Dúvidas

Prof. Orlando Saraiva Júnior
orlando.saraiva@unesp.br

Fechamento

Projeto “Banco de Dados FATEC Araras”



Com base no código **lista_linear_ordenada.cpp**, crie um mini banco de dados contendo os campos:

- id (campo índice)
- Nome
- RA
- Curso (Sistemas para Internet | outros)

Espera-se funções

- Para adicionar e remover aluno de forma ordenada.
- Uma busca sequencial por RA
- Uma forma de gravar dados e ler em um arquivo-texto