



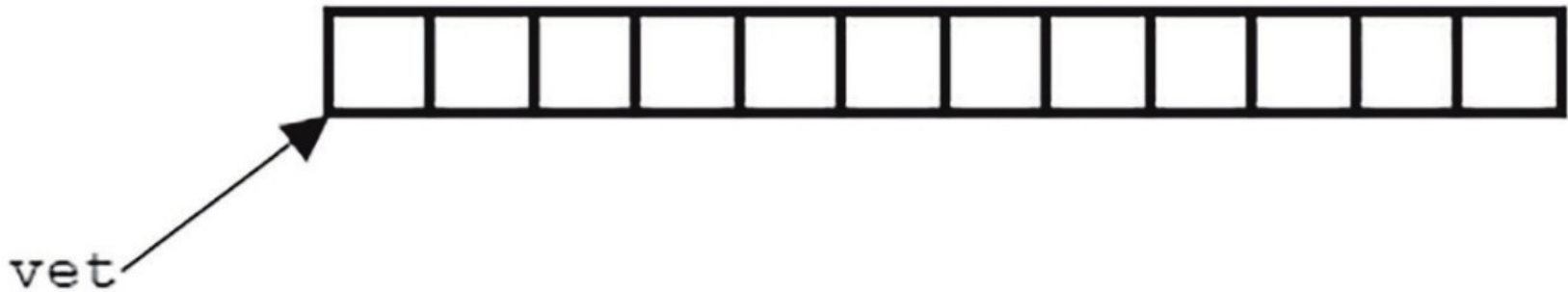
Estrutura de Dados - I

Listas Ligadas

Prof. MSc. Rafael Staiger Bressan
rafael.bressan@unicesumar.edu.br

Definição e Elementos de Listas Ligadas

- Segundo Celes (2004), a utilização do vetor para representar um conjunto de dados é a forma primitiva de representação, em que podemos definir um tamanho máximo de elementos a ser utilizados nesse vetor.





Definição e Elementos de Listas Ligadas

- O uso do vetor, ao ser declarado, reserva um espaço contíguo na memória para armazenar seus elementos. Assim, é possível acessar qualquer um dos seus elementos a partir do primeiro elemento, por meio de um ponteiro.
- Apesar de um vetor ser uma estrutura que permite o acesso aleatório aos elementos, não é uma estrutura flexível de dados, em razão do tamanho máximo que precisa ser definido.



Definição e Elementos de Listas Ligadas

- Uma lista ligada é uma coleção $L:[a1, a2, \dots, a_n]$, em que $n > 0$.
- Sua propriedade estrutural baseia-se apenas na posição relativa dos elementos, dispostos linearmente.
- É também conhecida como lista encadeada. É composta de um conjunto de dados dispostos por uma sequência de nós, em que a relação de sucessão desses elementos é determinada por um ponteiro que indica a posição do próximo elemento, podendo estar ordenado ou não.

Definição e Elementos de Listas Ligadas



Diferentemente dos vetores, em que o armazenamento é realizado de forma contígua, a lista ligada estabelece a sequência de forma lógica.



Definição e Elementos de Listas Ligadas

- Para trabalharmos com a lista encadeada, definimos um ponto inicial ou um ponteiro para o começo dela.
- A partir daí, podemos inserir elementos, remover ou realizar buscas na lista.



Definição e Elementos de Listas Ligadas

- Procedimentos básicos
 - Criação ou definição da estrutura de uma lista.
 - Inicialização da lista.
 - Inserção com base em um endereço como referência.
 - Alocação de um endereço de nó para inserção na lista.
 - Remoção do nó com base em um endereço como referência.
 - Deslocamento do nó removido da lista.

Definição e Elementos de Listas Ligadas

- Quando uma lista está sem nós, é definida como vazia ou nula, assim o valor do ponteiro para o próximo nó da lista é considerado ponteiro nulo





Definição e Elementos de Listas Ligadas

- Os elementos de uma lista são armazenados em posições sequenciais de memória, sempre que possível e de forma dinâmica, e permitem que a lista seja percorrida em qualquer direção.

Definição e Elementos de Listas Ligadas

- Os elementos de informação de uma lista podem ser do tipo int, char e/ou float. Ao criar uma estrutura de uma lista, definimos também o tipo de dado que será utilizado em sua implementação.

```
1  #include<stdio.h>
2  #include<stdlib.h>
3
4  struct lista {
5      int info;
6      struct lista* prox;
7  };
8  typedef struct lista Lista;
```

Definição e Elementos de Listas Ligadas

- Precisamos inicializar a lista para ser utilizada após sua criação. Para isso, basta criarmos uma função em que inicializamos a lista como nula, pois esta é uma das possíveis formas de inicialização

```
9  
10  Lista* inicializa(void){  
11      return NULL;  
12  }
```



Definição e Elementos de Listas Ligadas

- Os elementos de ligação em uma lista ligada são os ponteiros. Um ponteiro é um tipo de variável que armazena um endereço de memória e não o conteúdo da posição de memória.
 - **int *ptr;** /* sendo um ponteiro do tipo inteiro*/
 - **float *ptr;** /* sendo um ponteiro do tipo ponto flutuante*/
 - **char *ptr;** /* sendo um ponteiro do tipo caracteres*/



Definição e Elementos de Listas Ligadas

- Para sabermos o endereço da memória reservada à variável, utiliza-se o operador & com o nome de uma variável, enquanto o operador *(asterisco), utilizado com a variável do tipo ponteiro, acessa o conteúdo armazenado do endereço de memória.
 - **int x = 10; /*variável*/**
 - **int *p; /*ponteiro*/**
 - **p = &x; /*ponteiro p aponta para o endereço da variável x*/**



Definição e Elementos de Listas Ligadas

- Em listas, além do uso de ponteiros, utilizamos também as alocações dinâmicas de memória, que são porções de memórias para utilização das listas.
- Para compreendermos como funciona um ponteiro em uma lista, precisamos entender a função **malloc()**, *Memory Allocation* ou Alocação de Memória. É a responsável pela reserva de espaços na memória principal. Tem como finalidade alocar uma faixa de bytes consecutivos na memória do computador e retornar o endereço dessa faixa ao sistema.



Definição e Elementos de Listas Ligadas

- O trecho do código a seguir apresenta um exemplo de utilização da função `malloc()` e do ponteiro:
 - `char *pnt;`
 - `pnt = malloc (2); /* Aloca 2 bytes na memória */`
 - `scanf ("%c", pnt);`



Definição e Elementos de Listas Ligadas

- Em uma lista, precisamos alocar o tipo de dado no qual foram declarados seus elementos e, por esse tipo de dados ocupar vários bytes na memória, precisaremos utilizar a função **sizeof()**, que permite informar quantos bytes o tipo de elemento criado terá.
 - <https://www.youtube.com/watch?v=rf5BHtVYDIk>



Definição e Elementos de Listas Ligadas

```
#include <stdio.h>
#include <stdlib.h>

int main() {

    int *p;
    p=(int *) malloc(sizeof(int));

    if (!p) {
        printf("Erro de memoria insuficiente");
    }else{
        printf("Memoria alocada com sucesso");
    }

    return 0;
}
```



Definição e Elementos de Listas Ligadas

- Podemos classificar as listas de duas formas. Uma delas é a lista com cabeçalho, em que o conteúdo existente no primeiro elemento é irrelevante, pois serve apenas como marcador do seu início.
- Esse primeiro elemento permanece sempre como ponto inicial na memória, independentemente se a lista está com valores ou não.



Definição e Elementos de Listas Ligadas

- Assim, podemos considerar que start é o endereço inicial da nossa lista. Para determinar que a lista está vazia, consideramos: $\text{start} \Rightarrow \text{prox} == \text{NULL}$.
 - `celula *start;`
 - `start = malloc (sizeof (celula));`
 - `start \Rightarrow prox = NULL;`



Definição e Elementos de Listas Ligadas

- Outra classificação de uma lista é chamada de lista sem cabeçalho, em que o conteúdo do primeiro elemento tem a mesma importância que os demais.
- Assim, a lista é considerada vazia se o primeiro elemento é NULL. A criação desse tipo de lista vazia pode ser definida por `start = NULL`.



Operações com Listas Ligadas

- Inserção / remoção de um elemento na lista.
 - Início da lista.
 - Fim da lista.
 - Meio da lista e/ou local desejado.
- Percorrer a lista ligada
 - Exibir todos os elementos.
 - Procurar um elemento.
 -



Atividade Prática

- Desenvolva um programa em C para manipulação de uma lista encadeada com os métodos apresentados no slide 21.
 - Inserção (Início | Fim | Meio)
 - Remoção (Início | Fim | Meio)
 - Percorrer (Exibir todos os elementos | Procurar um elemento)

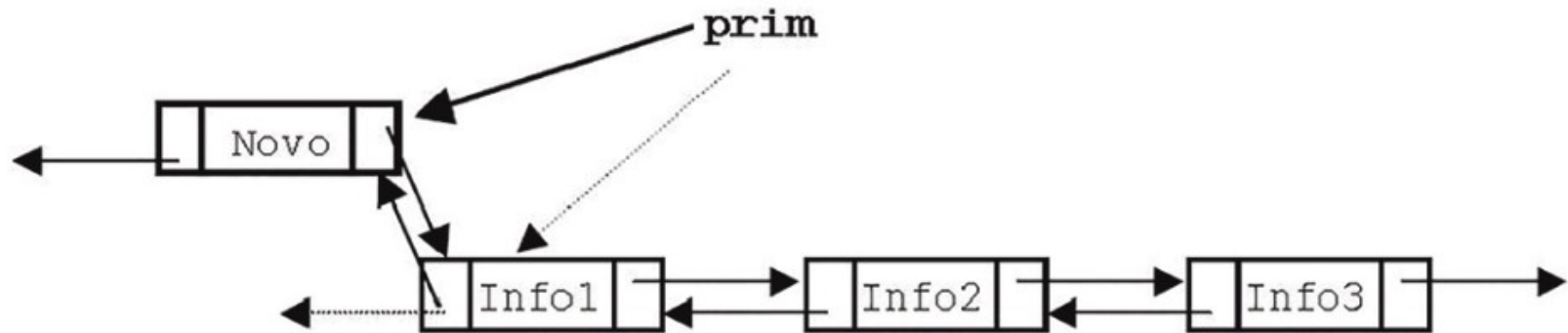
Listas Duplamente Ligadas



```
4 struct lista {  
5     int info;  
6     struct lista* ant;  
7     struct lista* prox;  
8 };  
9 typedef struct lista Lista;  
10
```

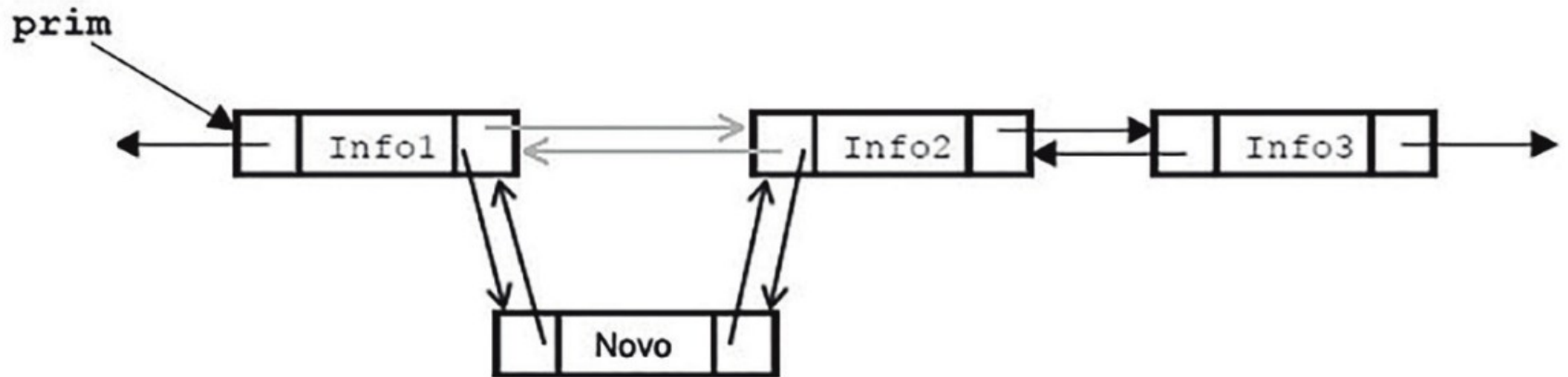
Listas Duplamente Ligadas

Adicionar elementos à lista duplamente ligada



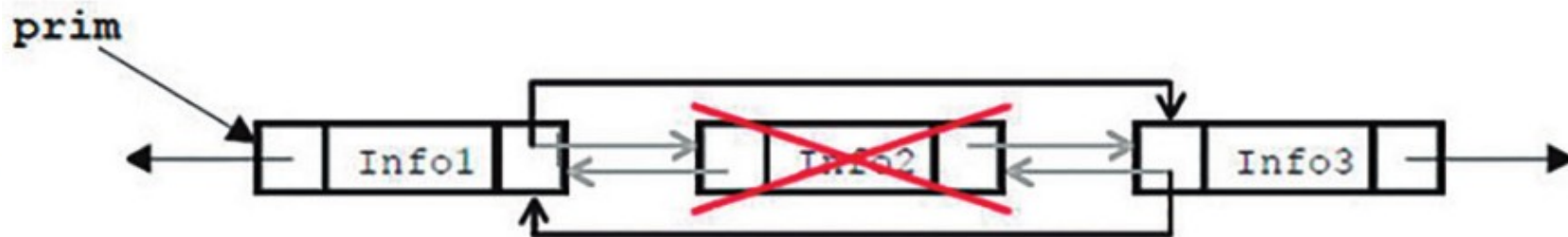
Listas Duplamente Ligadas

Inserindo no meio da lista



Listas Duplamente Ligadas

Remoção de um elemento da lista





Operações com Listas Duplamente Ligadas

- Inserção / remoção de um elemento na lista.
 - Início da lista.
 - Fim da lista.
 - Meio da lista e/ou local desejado.
- Percorrer a lista ligada
 - Exibir todos os elementos.
 - Procurar um elemento.
 -



Atividade Prática

- Desenvolva um programa em C para manipulação de uma lista duplamente encadeada com os métodos apresentados no slide 27.
 - Inserção (Início | Fim | Meio)
 - Remoção (Início | Fim | Meio)
 - Percorrer (Exibir todos os elementos | Procurar um elemento)