

Programação de Computadores

Profs. Marilton Sanchotene de Aguiar e Giovani Parente Farias

{marilton, giovani.pfarias}@inf.ufpel.edu.br

Cursos de Ciência e Engenharia de Computação
Centro de Desenvolvimento Tecnológico
Universidade Federal de Pelotas





Semana X

Listas Encadeadas



Uma lista encadeada é uma representação de uma sequência de objetos na memória do computador. Cada elemento da sequência é armazenado em uma célula da lista: o primeiro elemento na primeira célula, o segundo na segunda e assim por diante.

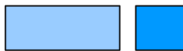


Estrutura de uma Lista Encadeada

Uma lista encadeada (*linked list*) é uma sequência de células; cada célula contém um objeto de algum tipo e o endereço da célula seguinte.

Por exemplo, os objetos armazenados nas células são do tipo `int`. A estrutura de cada célula de uma tal lista pode ser definida assim:

```
struct cel {  
    int  conteudo;  
    struct cel *prox;  
};
```



conteudo prox



- Por conveniência, definiremos o tipo `celula`:

```
typedef struct cel celula;
```

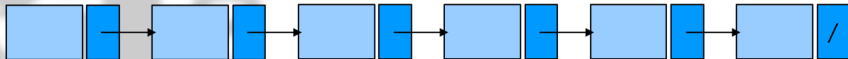
- Uma célula `c` e um ponteiro `p` para uma célula podem ser declarados assim:

```
celula c;  
celula *p;
```



Estrutura de uma Lista Encadeada

- Se c é uma célula então $c.conteudo$ é o conteúdo da célula e $c.prox$ é o endereço da próxima célula.
- Se p é o endereço de uma célula, então $p \rightarrow conteudo$ é o conteúdo da célula e $p \rightarrow prox$ é o endereço da próxima célula.
- Se p é o endereço da última célula da lista então $p \rightarrow prox$ é NULL.



Uma lista encadeada pode ser organizada de duas maneiras diferentes:

Lista com cabeça: O conteúdo da primeira célula é irrelevante servindo apenas para marcar o início da lista. A primeira célula é a cabeça (*head cell*) da lista. A primeira célula está sempre no mesmo lugar na memória, mesmo que a lista fique vazia. Digamos que `ini` é o endereço da primeira célula.

Então `ini→prox = NULL` se, e somente se, a lista está vazia.

Para criar uma lista vazia, basta:

```
celula *ini;  
ini = (celula *)malloc( sizeof (celula));  
ini->prox = NULL;
```



Lista sem cabeça: O conteúdo da primeira célula é tão relevante quanto o das demais. Nesse caso, a lista está vazia se o endereço de sua primeira célula é NULL.

Para criar uma lista vazia basta:

```
celula *ini;  
ini = NULL;
```



Inserção e Remoção em uma Lista

- A função `insere` é utilizada para inserir uma nova célula com conteúdo x entre a posição apontada por p e a posição seguinte em uma lista encadeada.
- Para se remover uma certa célula da lista, a ideia mais óbvia, que é apontar para a célula que se quer remover, claramente não é boa. É melhor apontar para a célula anterior à que se quer remover.
- A função `remove_end` foi desenvolvida com este propósito. Supõe-se que p é o endereço de uma célula de uma lista com cabeça e que se deseja remover a célula apontada por $p \rightarrow \text{prox}$.

Veja

- `exemplo-59.c`

- 1 Modifique o programa `exemplo-48.c`, de modo a alocar dinamicamente, em uma lista encadeada, os dados daquela lista de endereços.

