

Semana 3

Aula: https://www.youtube.com/watch?v=iBoWPFDQC_I&list=PL_JAaU8k6DQXxJ_HL_kiy8_jXzGs6e6EH&index=6&ab_channel=UNIVESP

Lista Linear Sequencial

Nesse tipo de lista cada elemento tem um sucessor e um antecessor, com a exceção do primeiro elemento que não tem um antecessor e nem o ultimo que não possui um sucessor

⇒ Essa estrutura segue o mesmo princípio quando trata-se de ordem lógica e ordem física, ou seja, a ordem lógica (armazenagem na memória) e a ordem física (index no array) são respeitadas.

Exemplo:

O elemento com index 2 vem antes do elemento 3 fisicamente (index $2 < 3$), assim como quando comparados os elementos de memória podemos ver que o elemento 2 também é antecessor ao 3.

Modos de busca:

- **Modo padrão:** Utiliza um for para verificar se o array já está no seu ultimo elemento e também verifica se o elemento atual da iteração é igual ao elemento buscado
- **Busca com sentinela:** Nesse caso, nós reduzimos 1 verificação (a de verificar se já estamos na ultima iteração do Array a todo momento) pois, no começo da busca, criamos um ultimo endereço colocando o valor que o usuario quer encontrar. Assim, a busca irá retornar o elemento encontrado. Bastando fazer somente uma verificação após as iterações: verificar se esse elemento encontrado é valido ou não (se for a sentinela não é válido)
- **Busca Binária:** Busca binária só pode ser utilizada se as chaves estiverem ordenadas. Entretanto, é a busca mais eficiente.

Exemplo de busca binária:

Busca binária

```
int buscaBinaria(LISTA* l, TIPOCHAVE ch) {
    int esq, dir, meio;
    esq = 0;
    dir = l->nroElem-1;
    while(esq <= dir) {
        meio = ((esq + dir) / 2);
        if(l->A[meio].chave == ch) return meio;
        else {
            if(l->A[meio].chave < ch) esq = meio + 1;
            else dir = meio - 1;
        }
    }
}
```