

Linguagens de Programação 1

Francisco Sant'Anna

Sala 6020-B

`francisco@ime.uerj.br`

`http://github.com/fsantanna-uerj/LP1`

Provas

- T3: 14/06 – Arquivos/Tipos/String
- P2: 28/06 – Tudo
- PF: 05/07 – Tudo

Listas Encadeadas

Exercício 9.1

- Criar um vetor `vet` de 5 posições
- Ler 5 números e guardá-los em `vet`
- Exibir todos os números de `vet`
- Ler um outro número `I`
- Remover o valor de `vet` no índice `I`
 - Manter o vetor sem buracos
 - As posições vazias devem ser preenchidas com 0
 - Exemplo:

- | 90 | 55 | 15 | 70 | 10 |

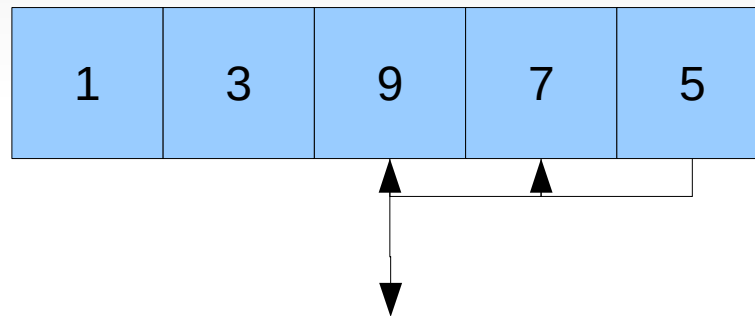
Após remover `I=2`:

- | 90 | 55 | 70 | 10 | 0 |

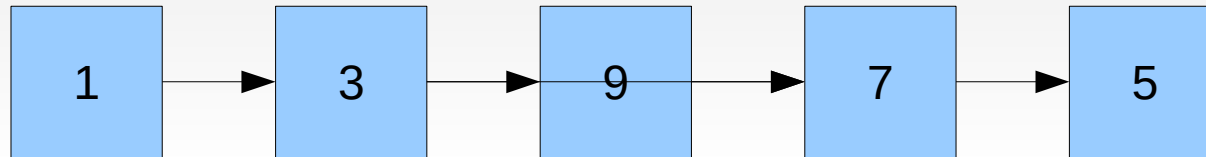
Exercício 9.1 - Problemas?

- Precisamos re-ajustar as posições o tempo todo.
- E se eu precisar de mais um elemento?

Vetor - Rígido



Lista Encadeada - Flexível



```
struct Caixa {  
    int valor;  
    struct Caixa* prox;  
};
```

Exercício 9.2

- [illegible]

Exercício 9.3

- Qual é o `prox` da última caixa?
 - Tem que ser um endereço que “não existe”
 - `NULL == 0`
- Crie uma função que receba um ponteiro para uma caixa e percorra todas as caixas (até o `NULL`), exibindo todos os valores
- `void exhibe (struct Caixa* caixa);`

O que é uma “lista”?

- 10 -> 20 -> 30
- ??? lista = ???;
- O que é uma lista vazia?
- Como é uma função que recebe uma lista?
- Uma lista é um **struct Caixa***