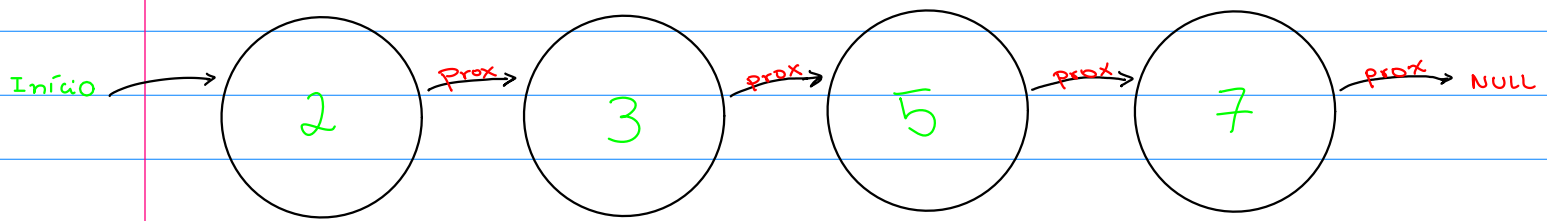


Discente: Paul Henrique Diniz de Lima Alencar.  
Cadeira: Estrutura de dados.

Questão IV - resolução:

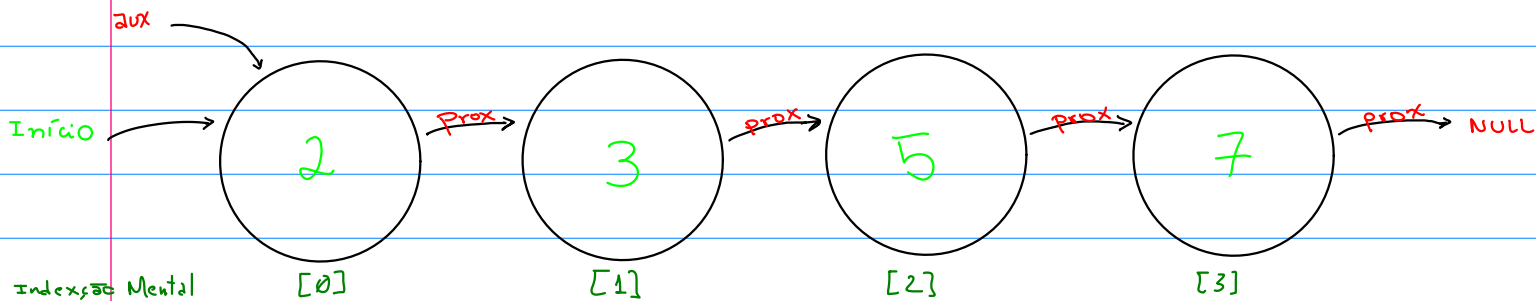


\* Remoção no meio de uma lista:

Passo 1 - precisamos criar um ponteiro aux e fazer com que ele receba o mesmo endereço do ponteiro início.

No `*aux = inicio;`

Graficamente vai ficar algo assim:

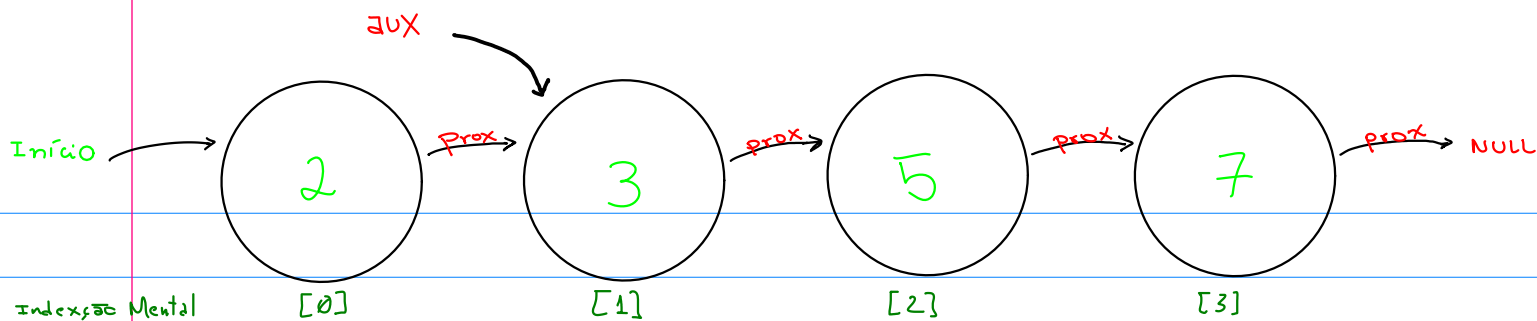


Passo 2 - Precisamos fazer com que o aux percorra até uma posição anterior da qual desejamos remover. Por exemplo, se eu desejo remover o elemento da posição [2], será necessário que meu aux percorra até a posição [1]. Como fazer isso?

Resposta: utilizar uma estrutura de repetição da seguinte forma:

```
No *aux = inicio;  
for(i=0; i<pos-1; i++){  
    aux = aux->prox;  
}
```

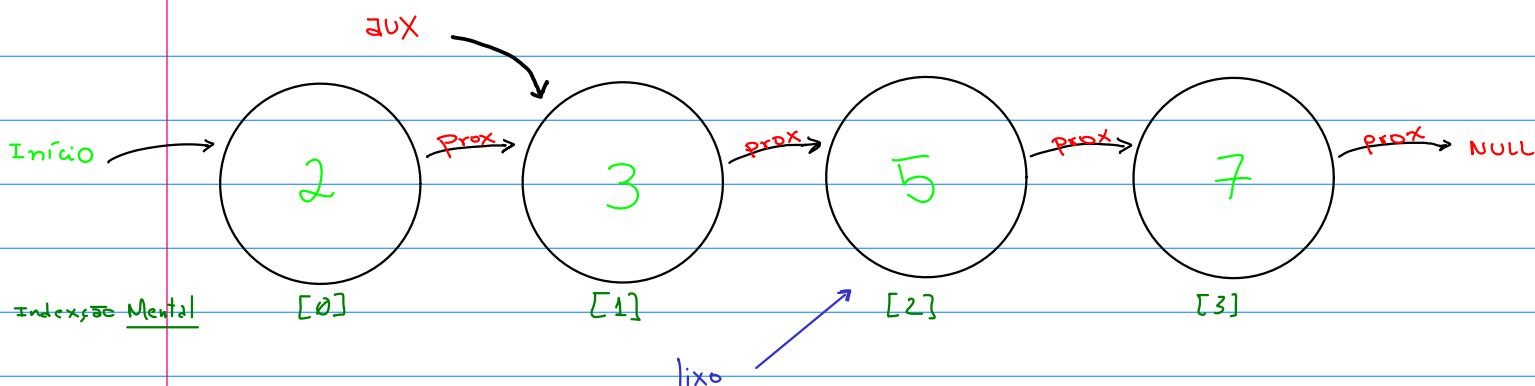
Depois dessas instruções a lista encadeada ficará assim:



**Passo 3** - agora precisamos criar um ponteiro que detenha o endereço do Nó que desejamos remover. Nesse caso queremos remover o Nó que possui o valor 5. Dessa forma, precisamos escrever o seguinte comando:

$\text{lixo} = \text{aux} \rightarrow \text{prox};$

Depois dessa instrução a lista encadeada ficará assim:



**Passo 4** - Já que vamos remover o Nó que contém o valor 5, precisamos fazer com que  $\text{aux} \rightarrow \text{prox}$  aponte para o Nó que possui o valor 7. Assim, precisamos fazer com que  $\text{aux} \rightarrow \text{prox}$ , guarde/armazene/detenha o endereço do Nó na posição [3]. Como fazer isso?

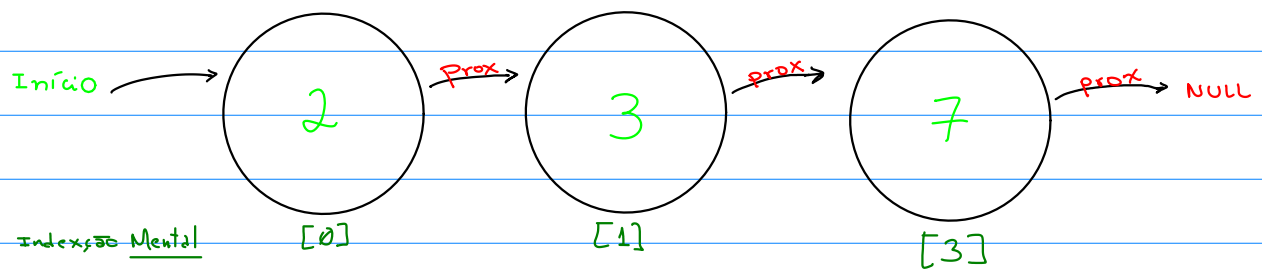
**Resposta:** precisamos escrever a seguinte instrução:

$\text{aux} \rightarrow \text{prox} = \text{aux} \rightarrow \text{prox} \rightarrow \text{prox};$

Essa instrução mantém o encadeamento da lista e além disso o Nó lixo (Nó que desejamos remover) não será mais alcançado. Assim, logicamente o Nó na posição [2], que possui o valor 5 não é mais um elemento de nossa lista. Porém, esse Nó ainda existe em nossa memória (5, 7, 9) e está ocupando espaço, logo o mais interessante agora seria desalocar esse espaço na memória. Para fazer isso basta digitar a seguinte instrução:

$\text{free}(\text{lixo});$

Nossa lista agora está:



## Referências:

FERNANDES, Tatiane et al. *Estrutura de dados - Uma abordagem gráfica do Laboratório de Estrutura de dados*. 1ed. Ceará. 2020.