

Lista Encadeada

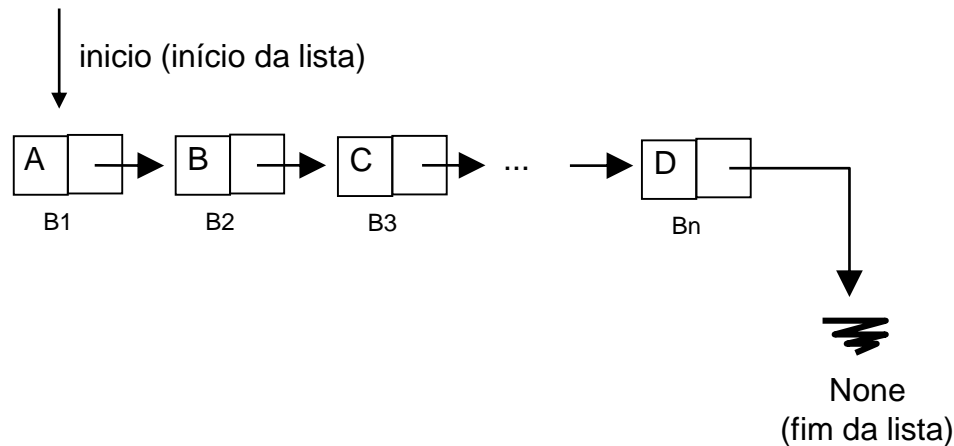
Lista Encadeada

Uma lista é uma estrutura que armazena elementos de forma alinhada, ou seja, com elementos dispostos um após o outro, como em uma lista de nomes, peças, valores, pessoas, compras, etc. Cada item da lista é chamado nó.

Lista Encadeada

Uma lista encadeada L , com n nós $B_1, B_2 \dots B_n$ é definida pelas seguintes características:

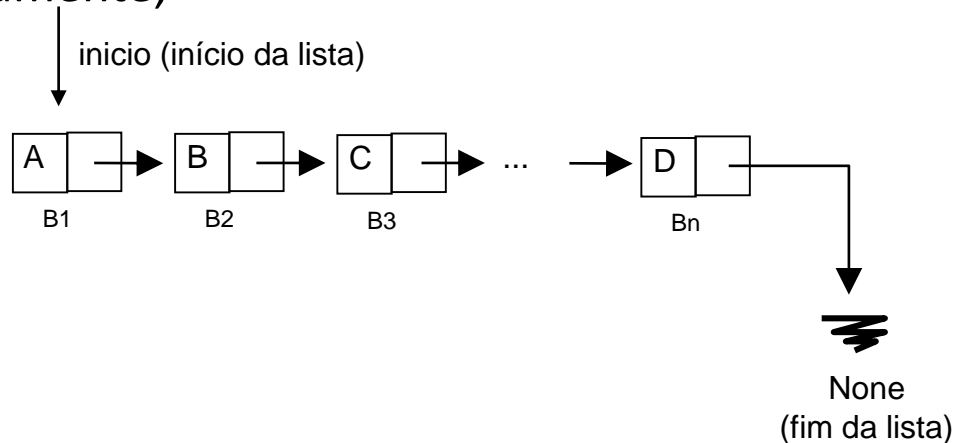
Cada nó B_i da lista, tem dois campos: **informação a ser armazenada, e a indicação do próximo elemento da lista**. Na figura, a informação é representada pelos caracteres A, B, C e D. A indicação do próximo elemento é indicada pela seta (um ponteiro para o próximo elemento). Os nós não estão necessariamente em seqüência física, na memória;



Lista Encadeada

O acesso aos elementos da lista ocorre através de um ponteiro para o início da lista (o primeiro elemento); Na figura, o início da lista é indicado por inicio.

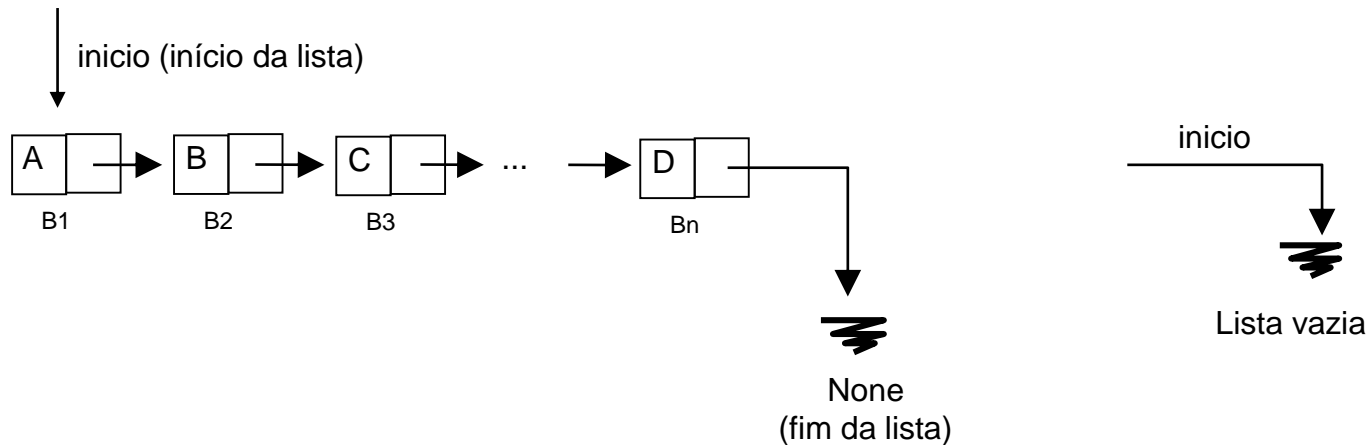
O acesso aos demais elementos ocorre sempre a partir do primeiro elemento (inicio), a seguir através da indicação de quem é o próximo na sequência. Não é possível ter acesso direto ao elemento C, no nó B3, por exemplo. Para acessar o elemento C, no nó B3, é preciso acessar o primeiro elemento através de inicio, depois ir para o próximo elemento através do ponteiro, e depois para o próximo novamente;



Lista Encadeada

O último nó da lista aponta para um None. É essa a indicação de que o nó B_n guarda o último elemento do conjunto;

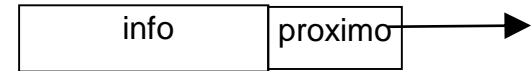
A lista vazia (sem elementos) é representada pelo apontador de início da lista (início) apontando para None.



Lista Encadeada

Na lista a inserção e a remoção podem ocorrer no início, meio ou fim da lista.

```
class ListaNo:  
    def __init__(self,info):  
        self.info=info  
        self.proximo=None
```



Lista Encadeada

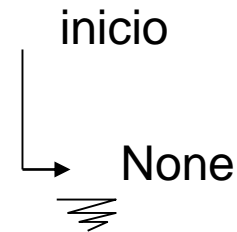
Operações sobre listas

```
class ListaNo:
```

```
    def __init__(self,info):
```

```
        self.info=info
```

```
        self.proximo=None
```



```
class ListaLigada:
```

```
    def __init__(self):
```

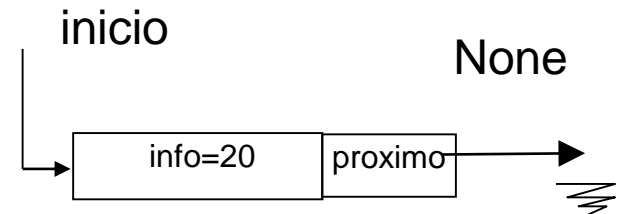
```
        self.inicio=None
```

```
    def inserelInicio(self,info):
```

```
        no=ListaNo(info)
```

```
        no.setProximo(self.inicio)
```

```
        self.inicio=no
```



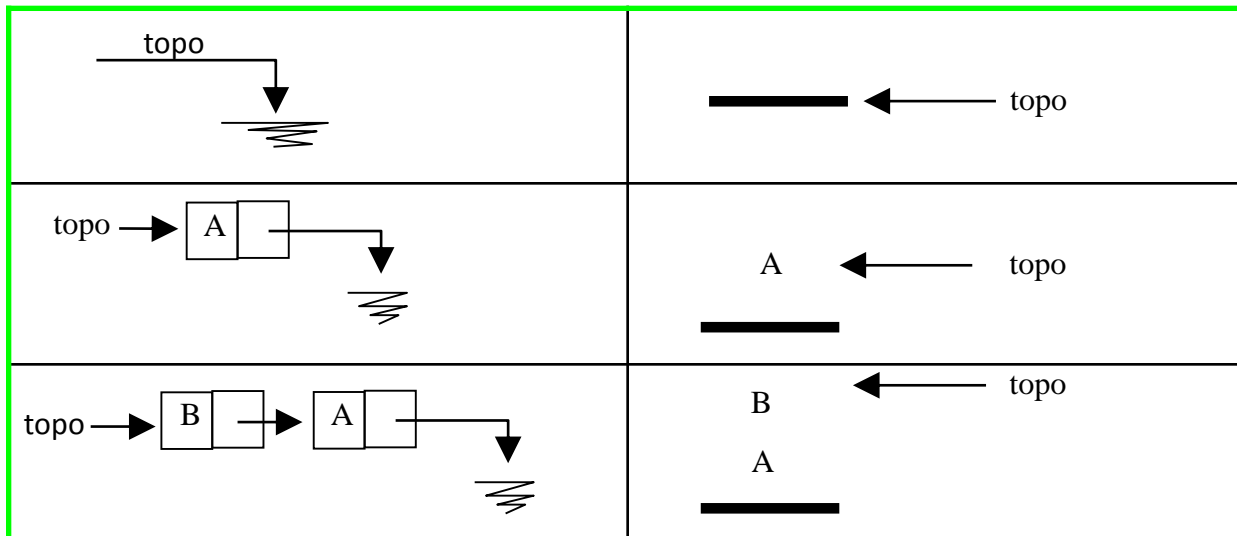
```
lista = ListaLigada()
```

```
lista.inserelInicio(20)
```

Lista Encadeada

Operações sobre listas

Uma Pilha pode ser vista como uma lista onde a remoção e a inserção ocorrem no início.

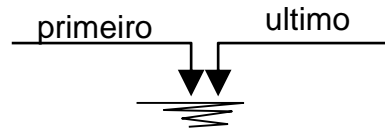


Lista Encadeada

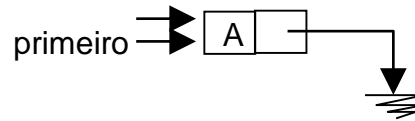
Operações sobre listas

Uma fila pode ser vista como uma lista onde a remoção é feita de um lado e a inserção de outro lado.

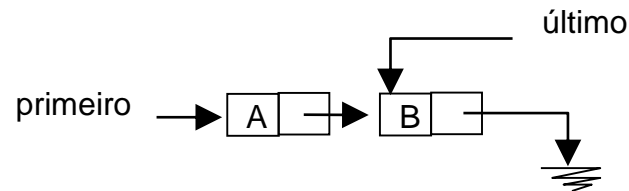
fila vazia



insere A



insere B



retira

