

# **ESTRUTURA DE DADOS**

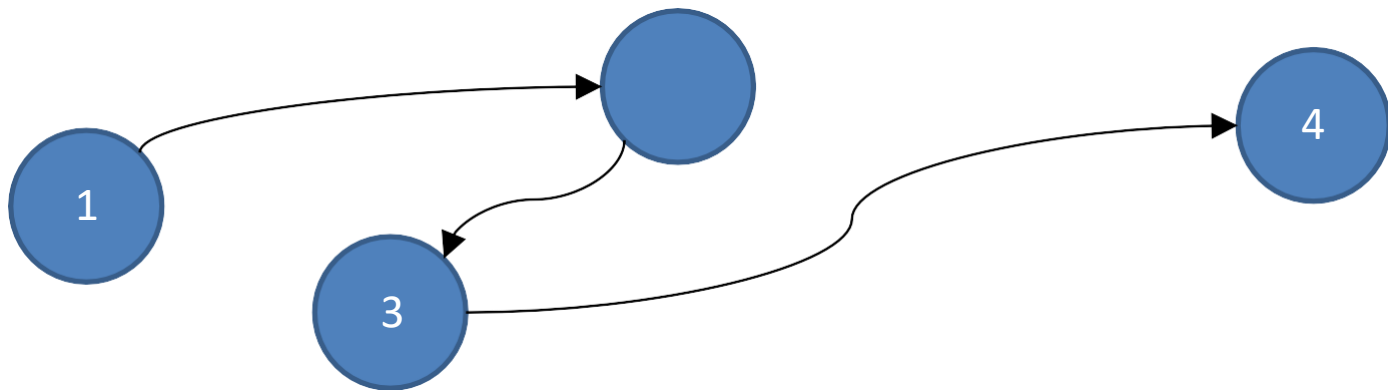
## **LISTAS LINEARES ENCADEADAS**

Baseado no material de prof. Dr. Daniel Caetano

# Discussão sobre Listas Encadeadas

---

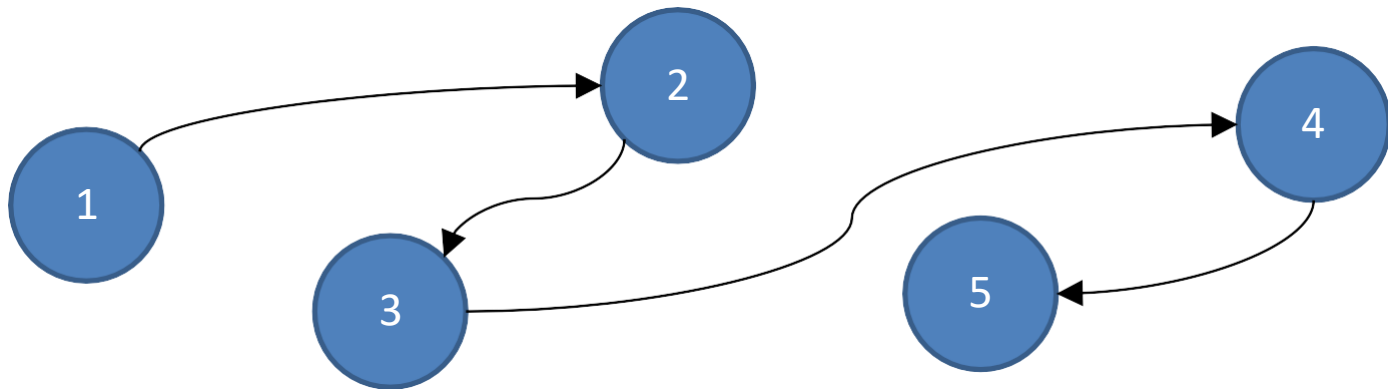
- Lista Encadeada: lista de elementos interligados, como em uma corrente
  - O primeiro elemento aponta para o segundo
  - O segundo aponta para o terceiro
  - O terceiro aponta para o quarto
  - ...



# Discussão sobre Listas Encadeadas

---

- Lista Encadeada: lista de elementos interligados, como em uma corrente
- Para acrescentar outro elemento no fim?
  - Criar o novo elemento
  - Associá-lo à lista (encadeá-lo)



# Como criar um novo nó?

---

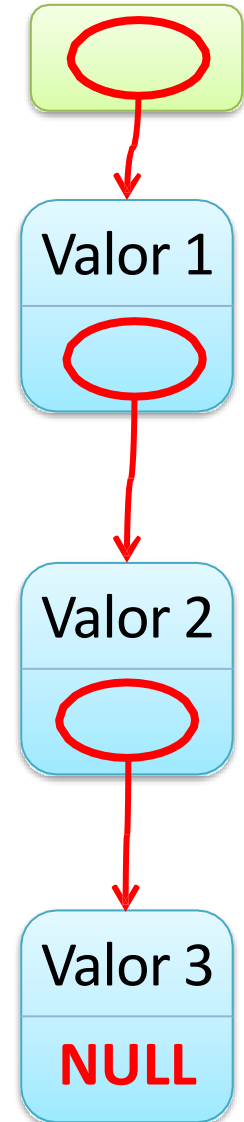
## Alocação Dinâmica de Memória

- Reservar no momento necessário: `malloc`
- Liberar quando não for mais necessário: `free`

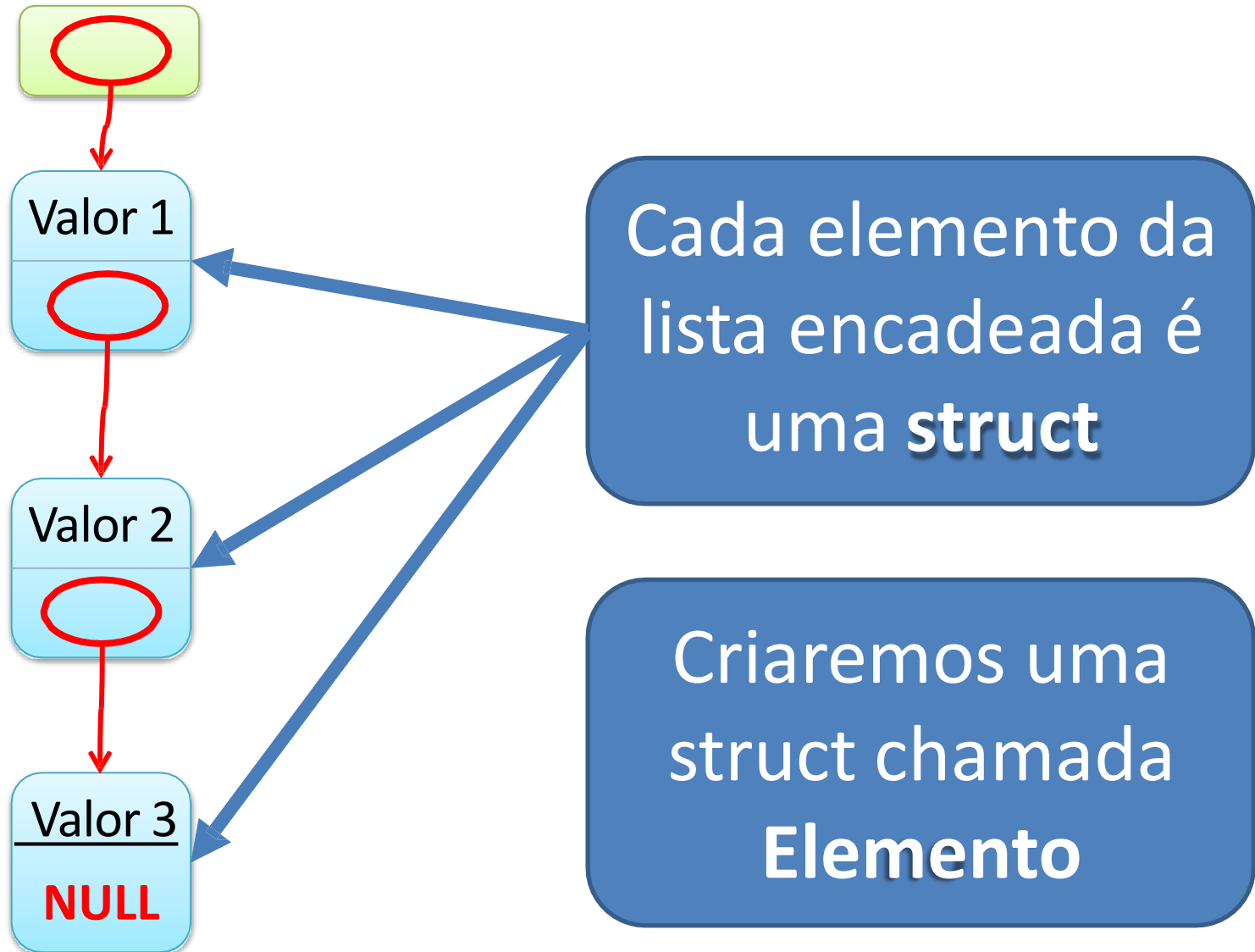
# Representando Listas Encadeadas

---

- Como vimos na dinâmica:
  - Lista encadeada é composta por **nós**
- Um nó tem a função de...
  - Guardar um elemento da lista
- Que outra informação tem o nó?
  - Um ponteiro que indique o próximo nó
- E quando não há um próximo nó?
  - Ajustamos o ponteiro para valer NULL
- Onde começa uma lista encadeada?
  - Precisamos de um ponteiro para isso!

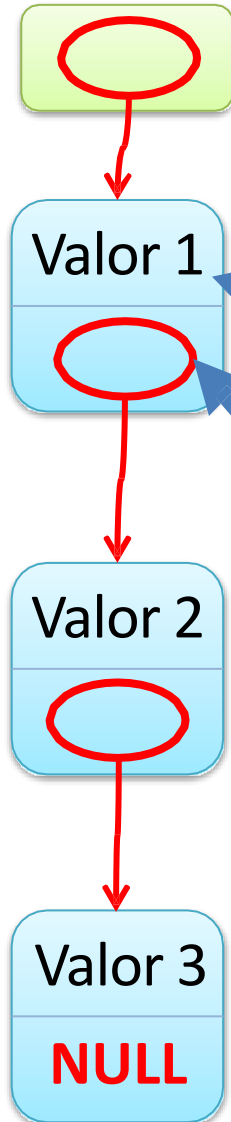


# Representando Listas Encadeadas



# Representando Listas Encadeadas

```
struct elemento {  
    int info;  
    struct elemento* prox;  
};  
typedef struct elemento Elemento;
```

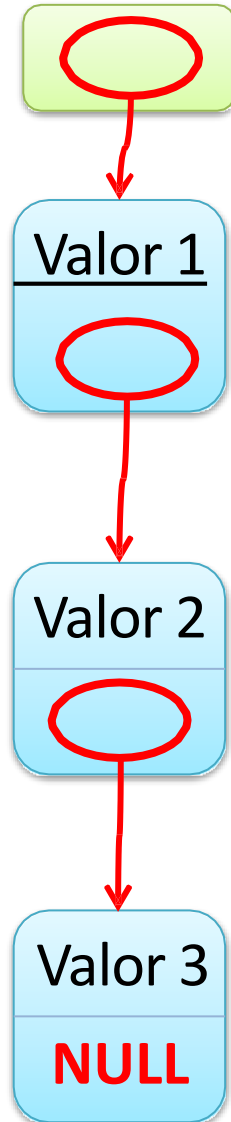


Cada “Elemento” é  
composto de uma  
**informação**

E de um **ponteiro**  
para outro “Elemento”

# Representando Listas Encadeadas

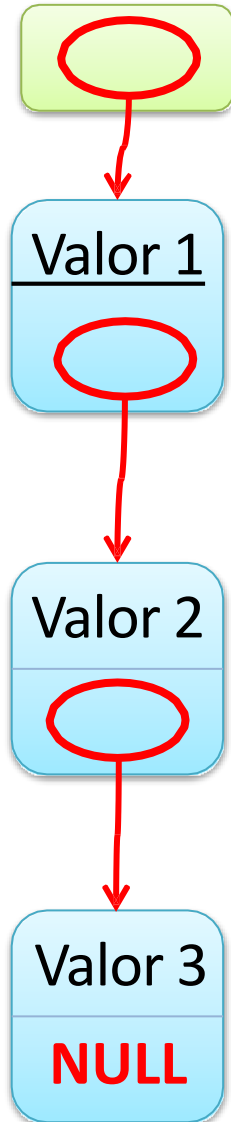
```
struct lista {  
    Elemento * prim;  
};  
typedef struct lista tLista;
```



A lista é  
representada por  
um ponteiro para o  
primeiro Elemento



# Representando Listas Encadeadas

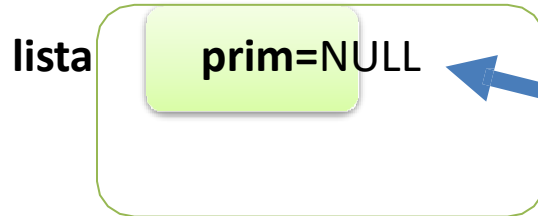


```
struct elemento {  
    int info;  
    struct elemento* prox;  
};  
typedef struct elemento Elemento;
```

```
tLista * lst;
```

# **INICIALIZANDO LISTAS ENCADEADAS E INSERINDO NÓS**

# Listas Encadeadas: Inicializando

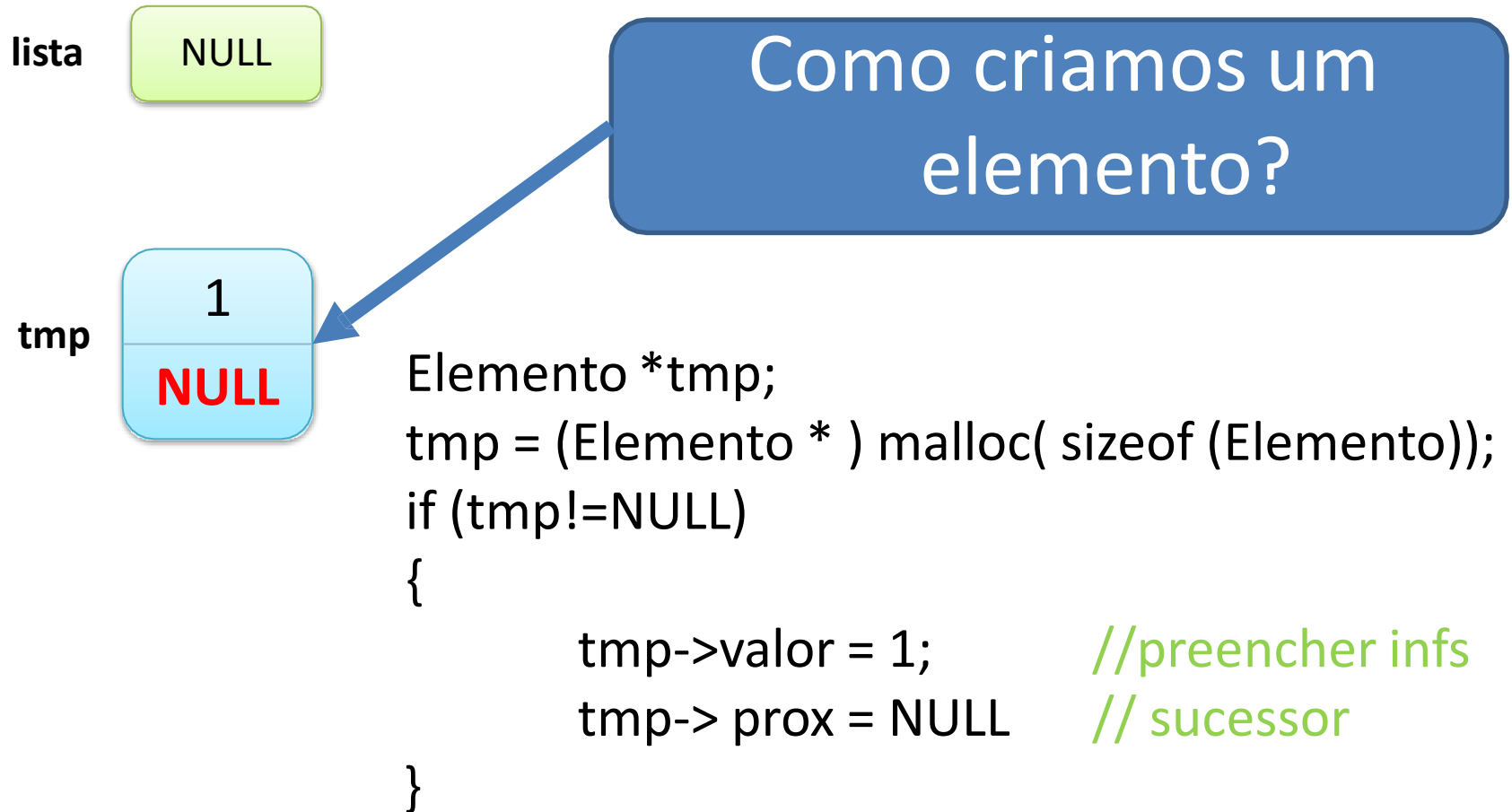


Quando criamos uma lista, ela está vazia

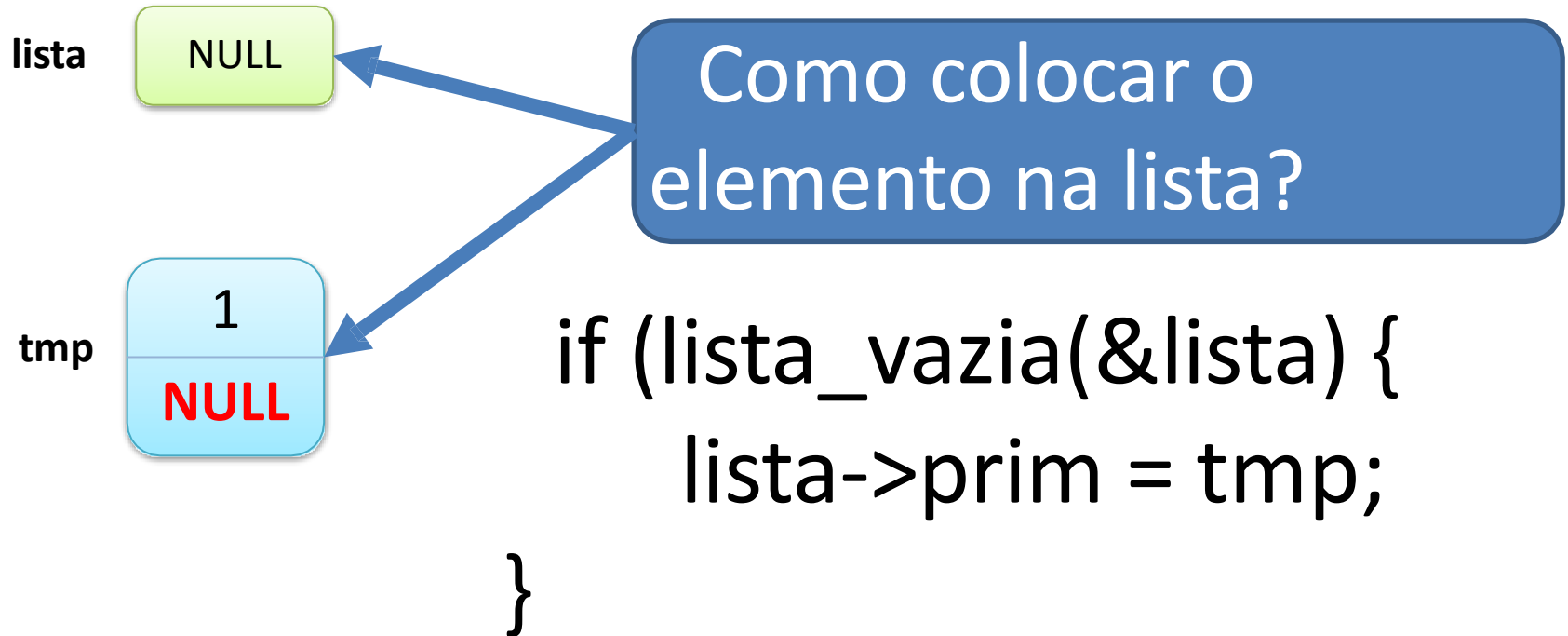
```
tLista lista;  
lista_cria(&lista);
```

```
void lista_cria(tLista * lst) {  
    lst->prim=NULL  
}
```

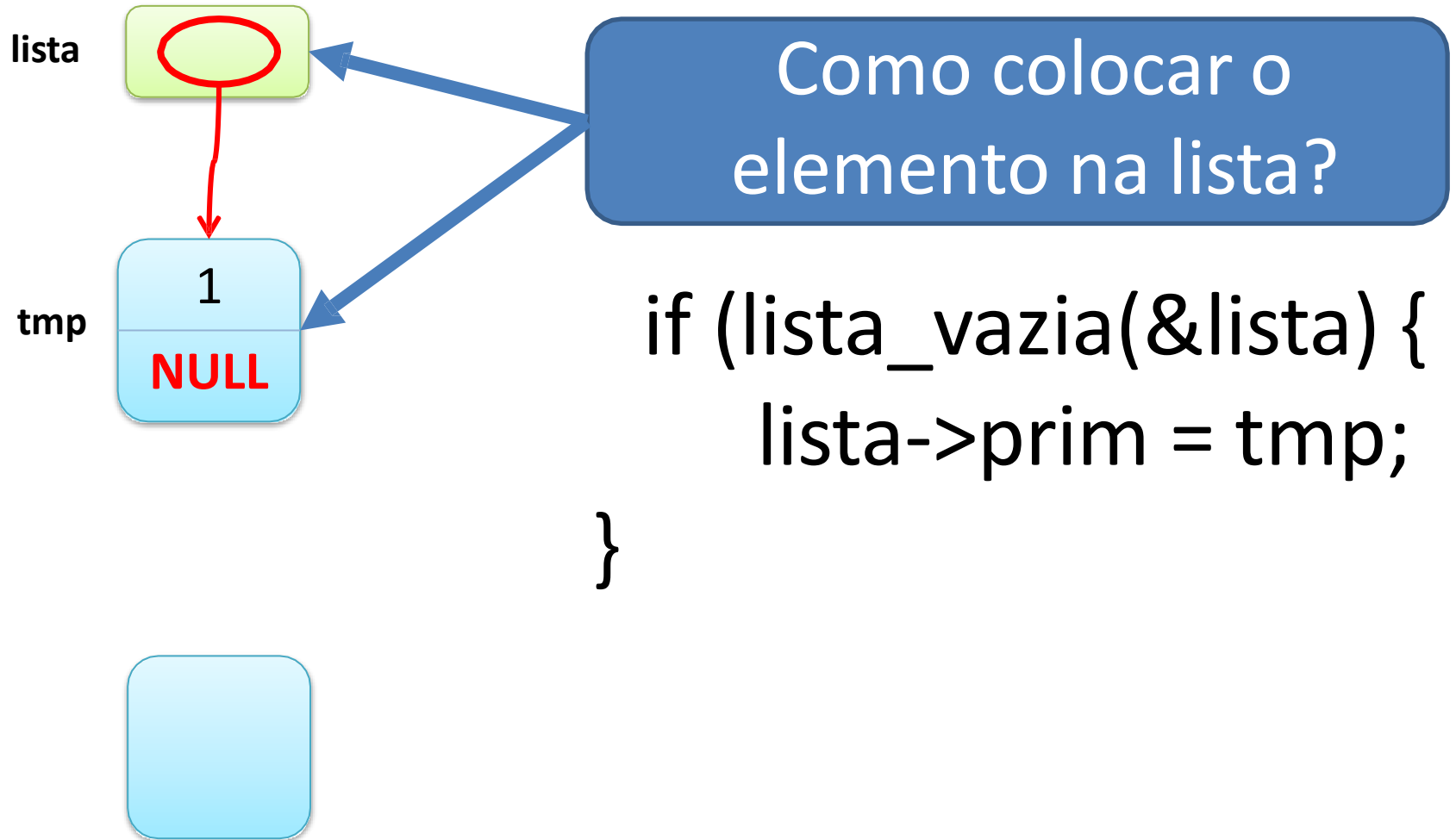
# Listas Encadeadas: Inserção



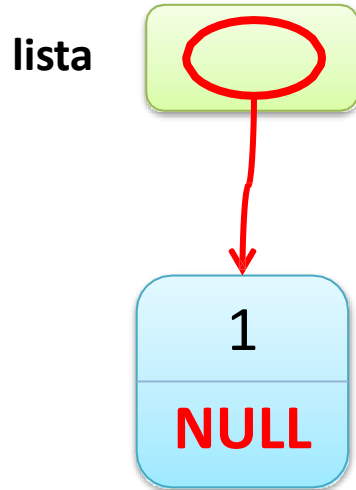
# Listas Encadeadas: Inserção



# Listas Encadeadas: Inserção



# Listas Encadeadas: Inserção



```
if (lista_vazia(&lista) {  
    lista->prim = tmp;  
}
```



E para acrescentar um  
outro elemento?

# Listas Encadeadas: Inserção

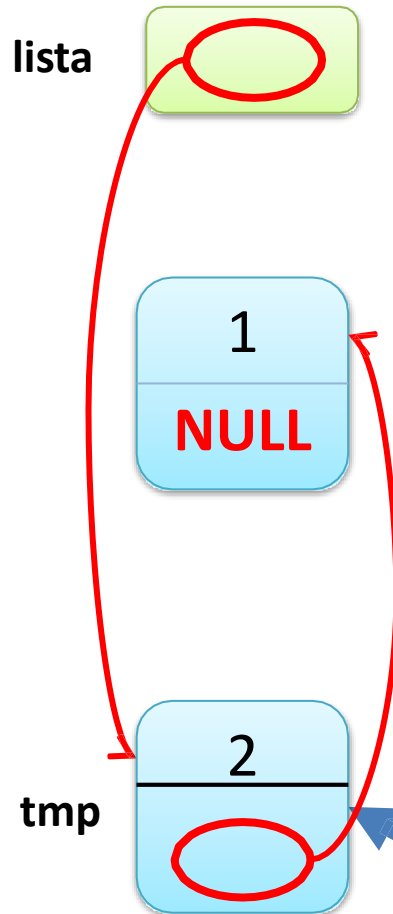


```
if (lista_vazia(&lista) {  
    lista->prim = tmp;  
  
} else {  
    tmp->prox = lista  
}
```

E para acrescentar um  
outro elemento?



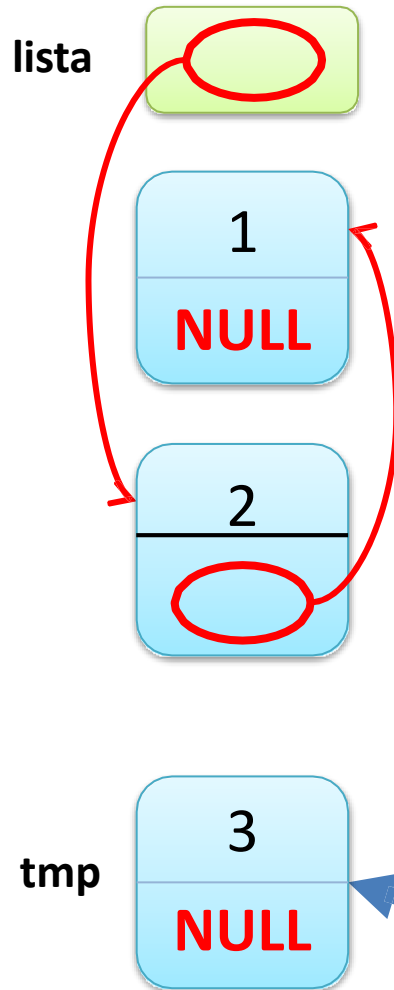
# Listas Encadeadas: Inserção



```
if (lista_vazia(&lista) {  
    lista->prim = tmp;  
} else {  
    tmp->prox = lista  
    lista->prim = tmp;  
}
```

E para acrescentar um  
outro elemento?

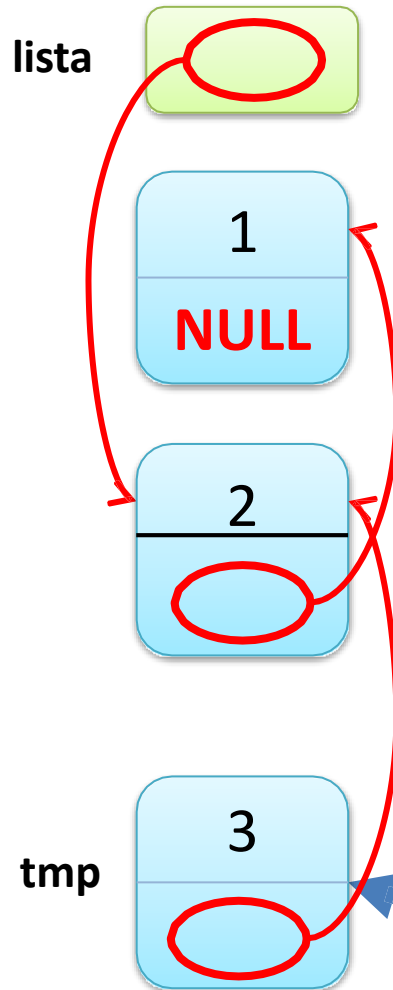
# Listas Encadeadas: Inserção



```
if (lista_vazia(&lista) {  
    lista->prim = tmp;  
} else {  
    tmp->prox = lista  
    lista->prim = tmp;  
}
```

E para acrescentar um  
outro elemento?

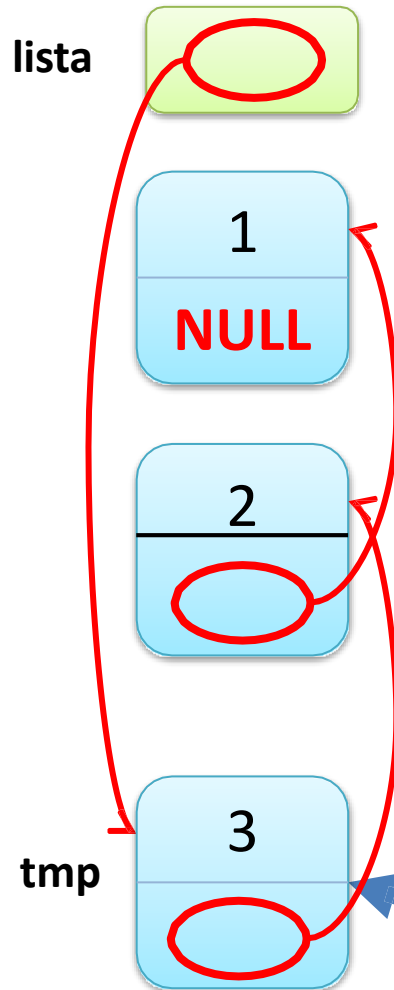
# Listas Encadeadas: Inserção



```
if (lista_vazia(&lista) {  
    lista->prim = tmp;  
} else {  
    tmp->prox = lista  
    lista->prim = tmp;  
}
```

E para acrescentar um  
outro elemento?

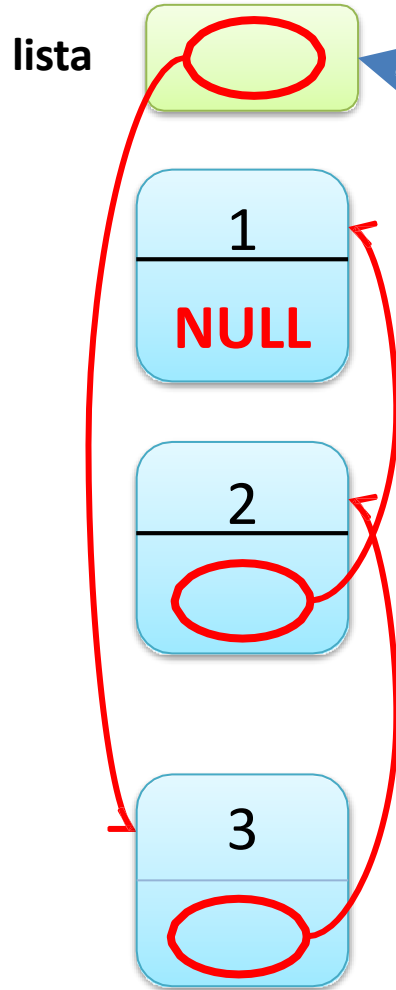
# Listas Encadeadas: Inserção



```
if (lista_vazia(&lista) {  
    lista->prim = tmp;  
} else {  
    tmp->prox = lista  
    lista->prim = tmp;  
}
```

E para acrescentar um  
outro elemento?

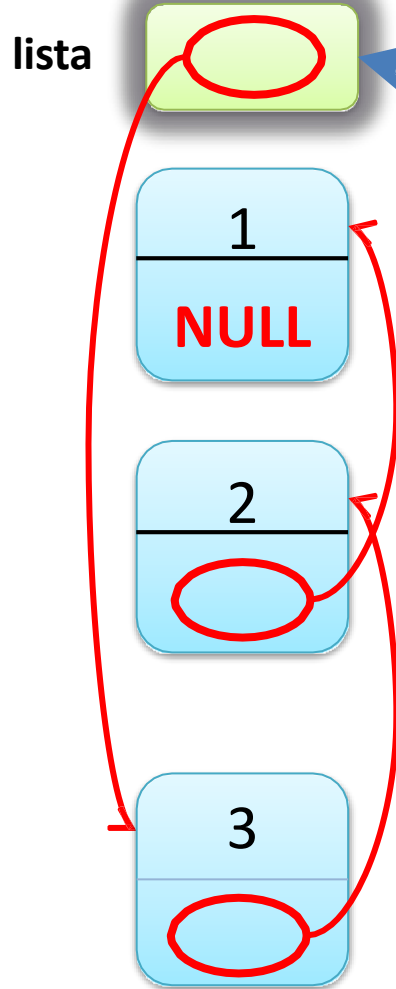
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

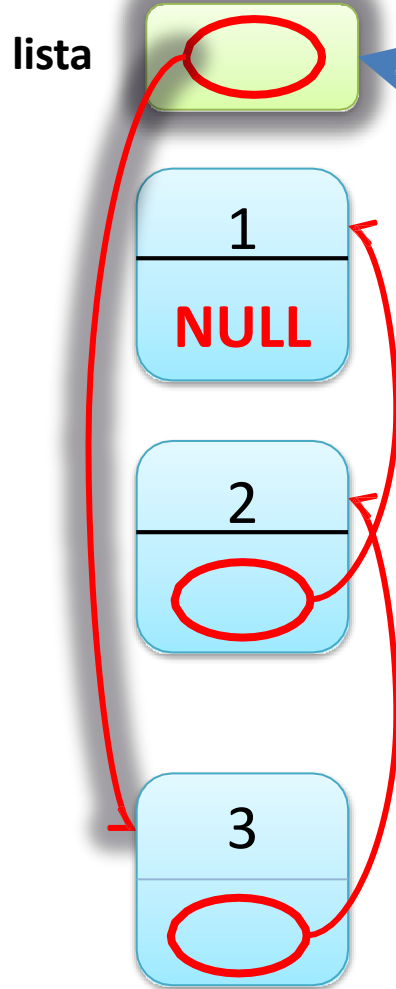
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

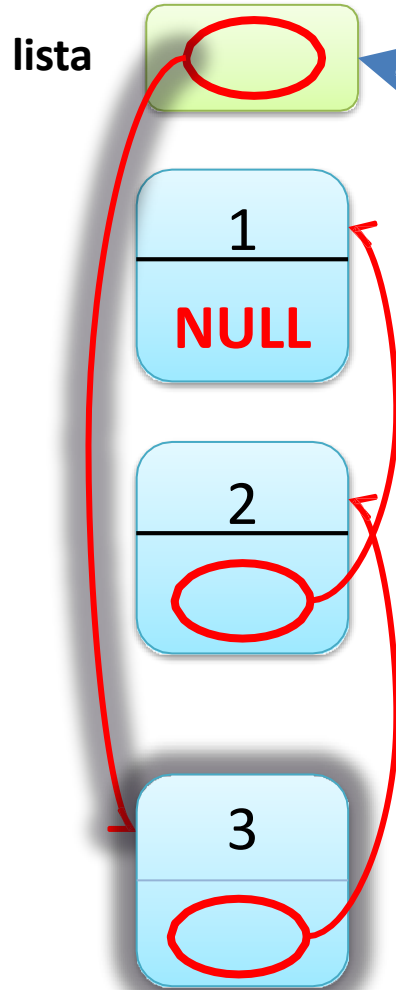
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

# Listas Encadeadas: Listar/Percorrer

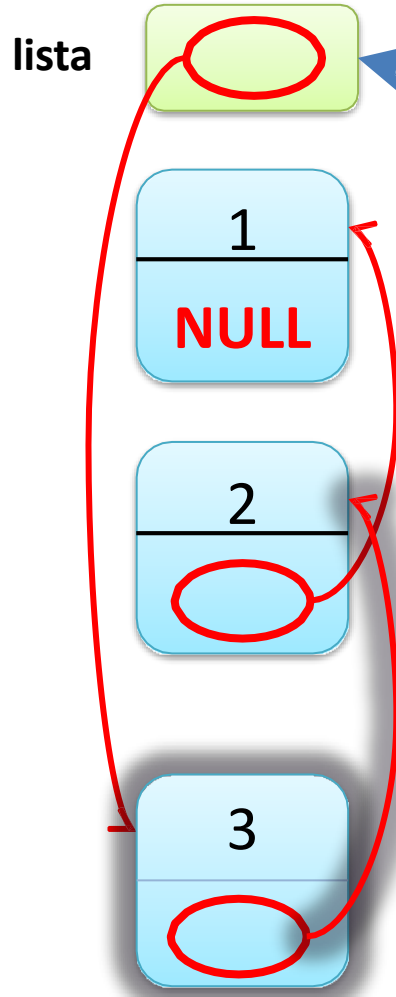


Como percorrer uma lista encadeada?

Basta seguir os ponteiros...



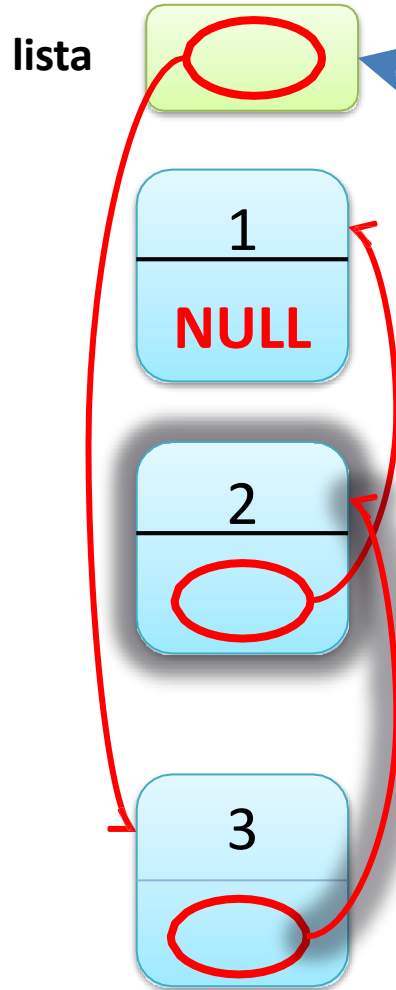
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma  
lista encadeada?

Basta seguir os  
ponteiros...

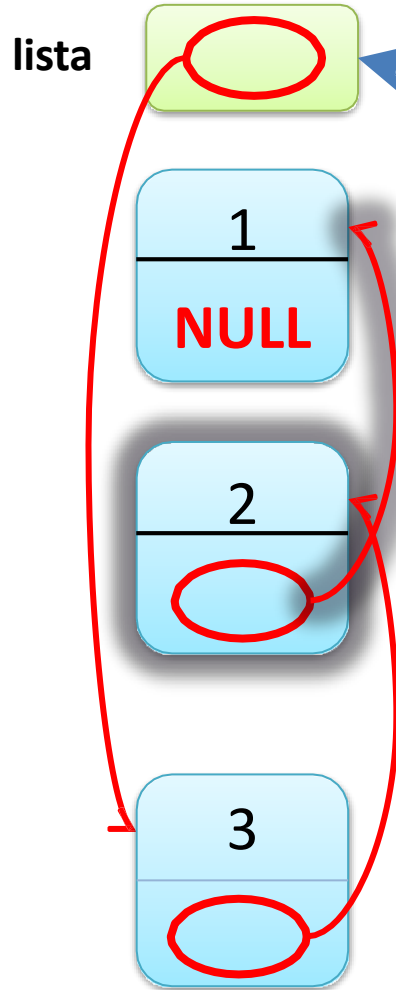
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

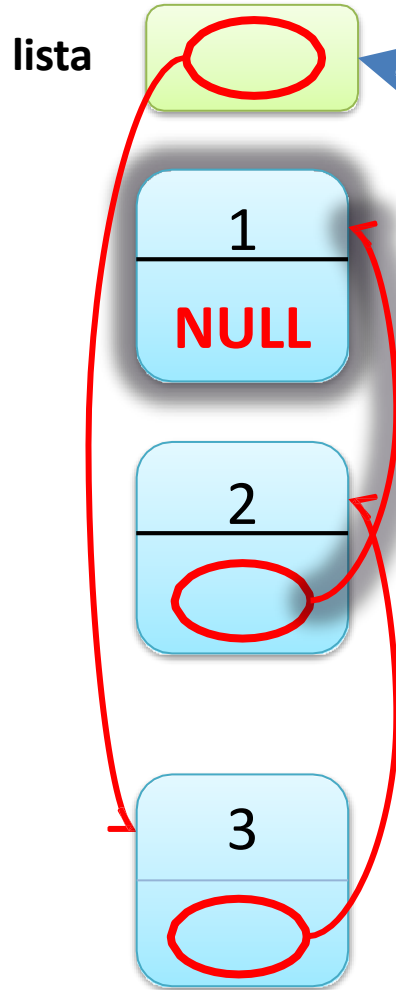
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

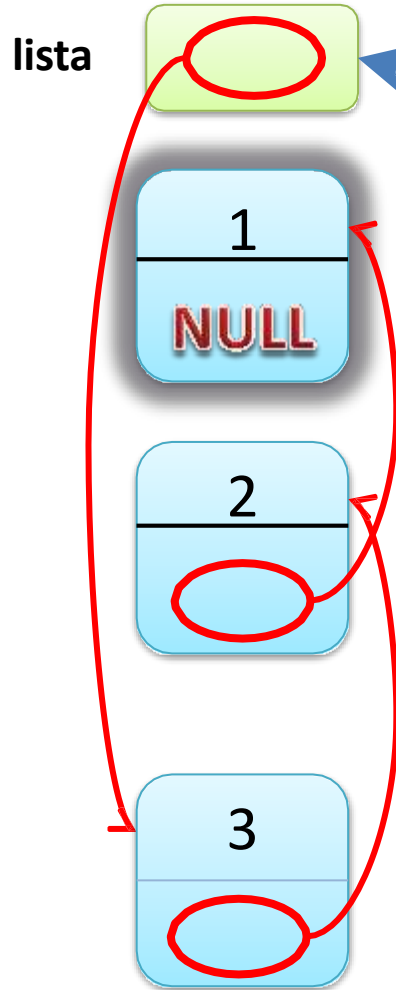
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

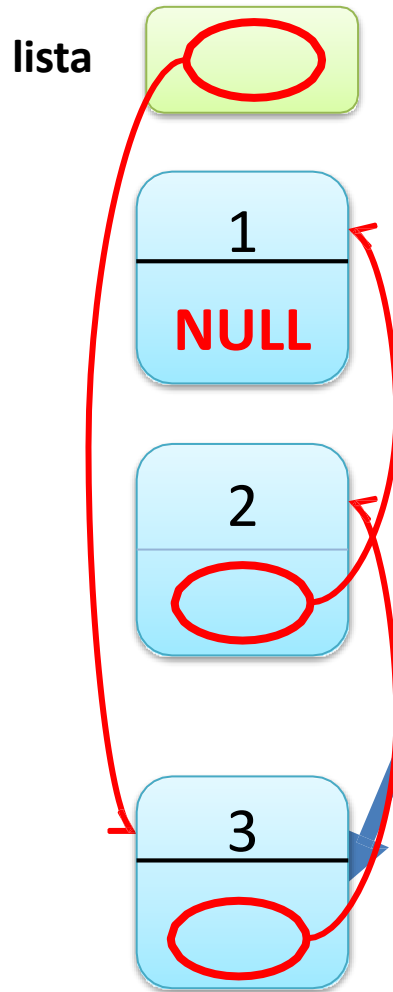
# Listas Encadeadas: Listar/Percorrer



Como percorrer uma lista encadeada?

Basta seguir os ponteiros...

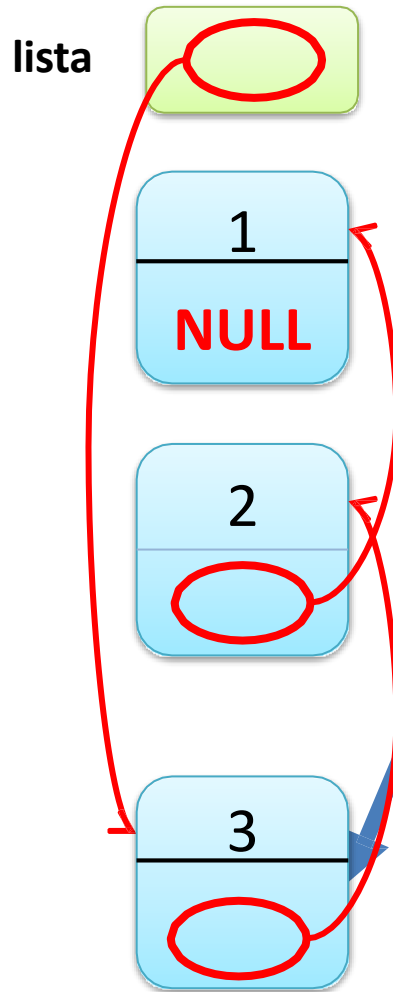
# Listas Encadeadas: Remoção



Como remover o primeiro elemento?

```
if (!lista_vazia(lista )) {  
    tmp = lista->prim;  
    lista->prim= tmp->prox;  
    free( tmp);  
}
```

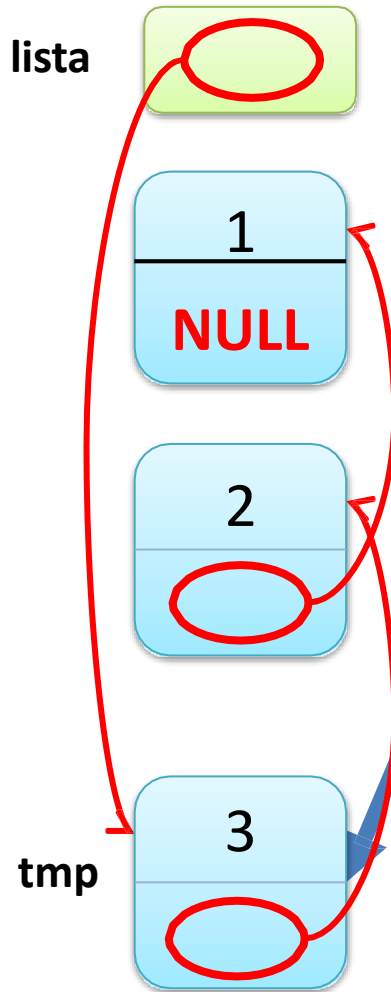
# Listas Encadeadas: Remoção



Como remover o primeiro elemento?

```
if (!lista_vazia(lista )) {  
    tmp = lista->prim;  
    lista->prim = tmp->prox;  
    free( tmp);  
}
```

# Listas Encadeadas: Remoção

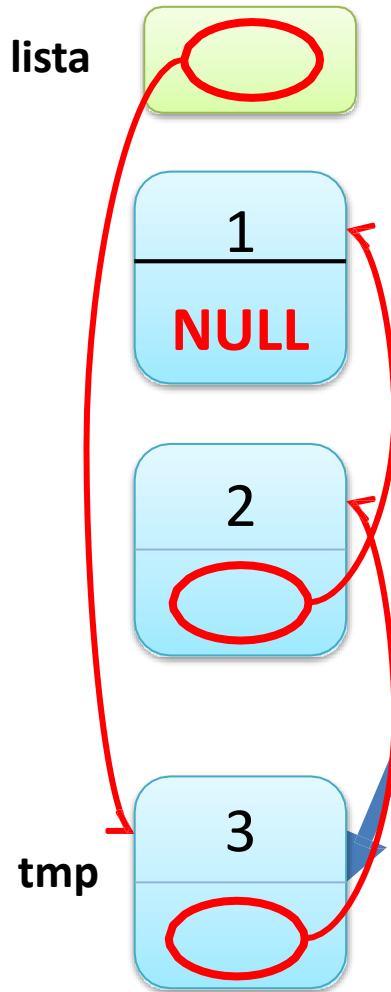


Como remover o primeiro elemento?

```
if (!lista_vazia(lista)) {  
    tmp = lista->prim;  
    lista->prim = tmp->prox;  
    free( tmp);  
}
```



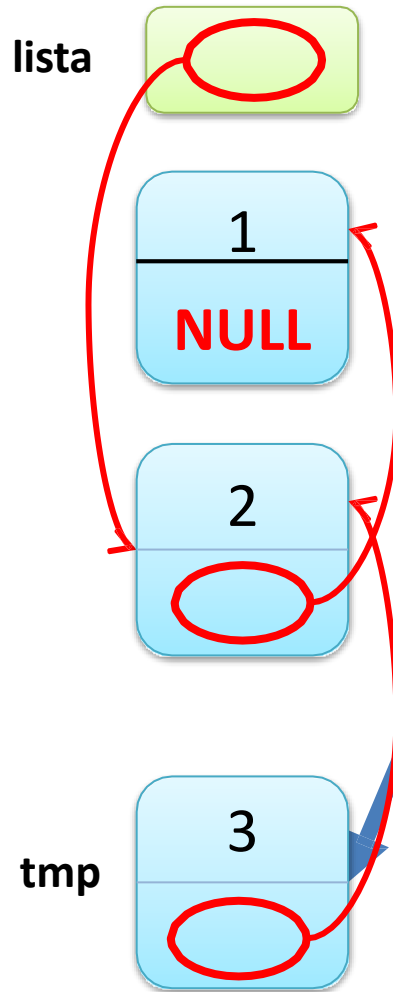
# Listas Encadeadas: Remoção



Como remover o primeiro elemento?

```
if (!lista_vazia(lista )) {  
    tmp = lista->prim;  
    lista->prim= tmp->prox  
    free( tmp);  
}
```

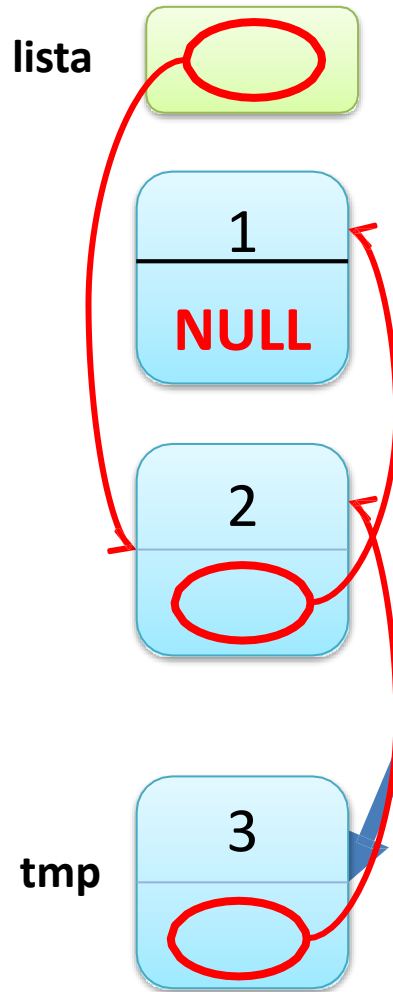
# Listas Encadeadas: Remoção



Como remover o primeiro elemento?

```
if (!lista_vazia(lista )) {  
    tmp = lista->prim;  
    lista->prim = tmp->prox;  
    free( tmp);  
}
```

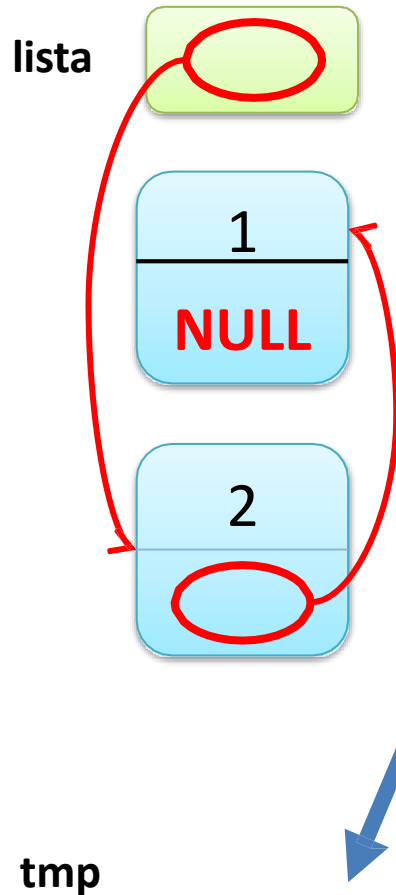
# Listas Encadeadas: Remoção



Como remover o primeiro elemento?

```
if (!lista_vazia(lista )) {  
    tmp = lista->prim;  
    lista->prim= tmp->prox;  
    free( tmp);  
}
```

# Listas Encadeadas: Remoção



Como remover o primeiro elemento?

```
if (!lista_vazia(lista )) {  
    tmp = lista->prim;  
    lista->prim= tmp->prox;  
    free( tmp);  
}
```

# Operações Com Listas Encadeadas

---

- Apresentadas
  - Inicialização
  - Inserção no Início
  - Listar / Percorrer
  - Remover
- A realizar:
  - Buscar
  - Inserir no final
  - Inserir no meio
  - Remover do meio