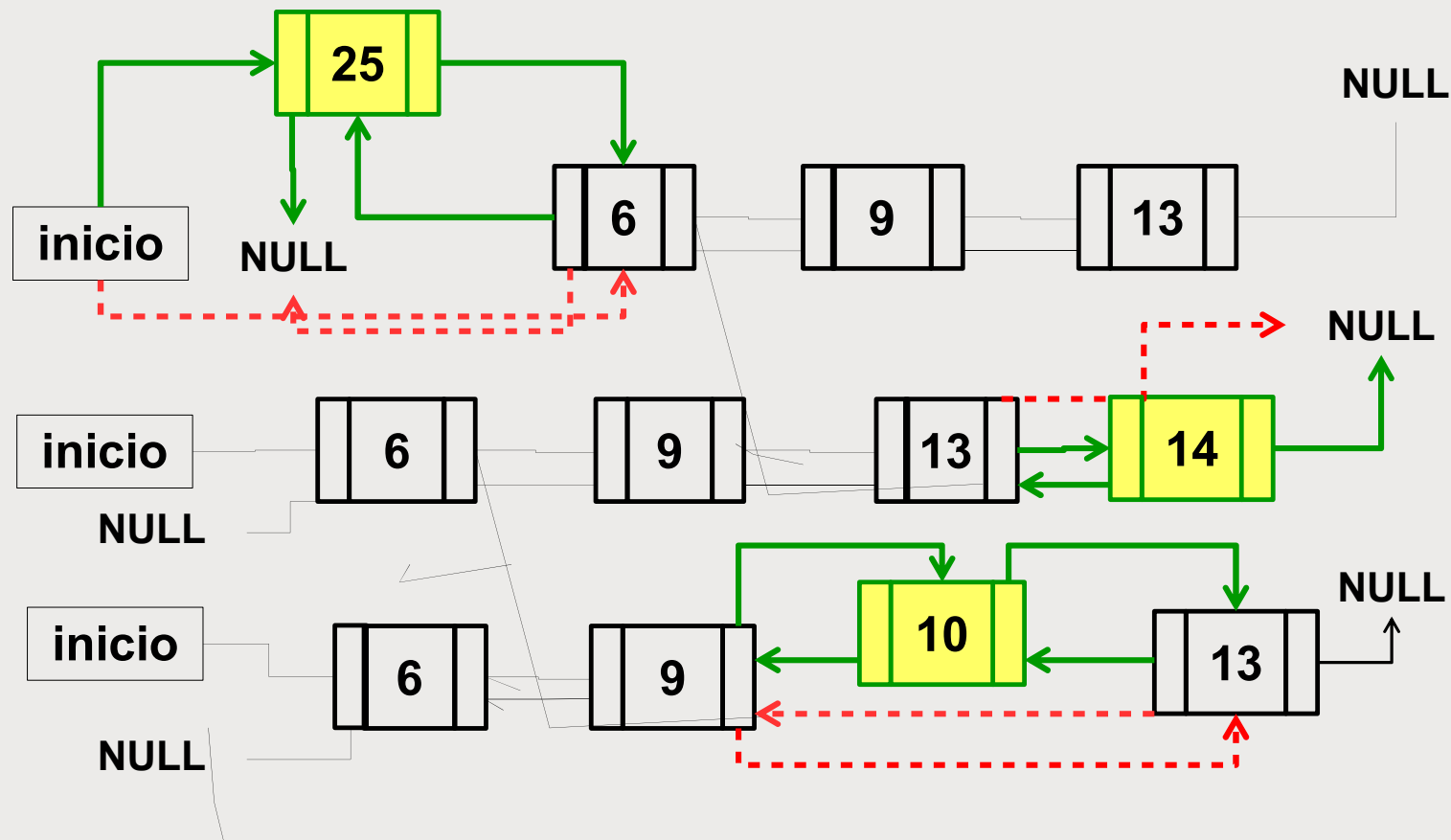


# LISTA DINÂMICA DUPLAMENTE ENCADEADA Inserção



# LISTA DINÂMICA DUPLAMENTE ENCADEADA

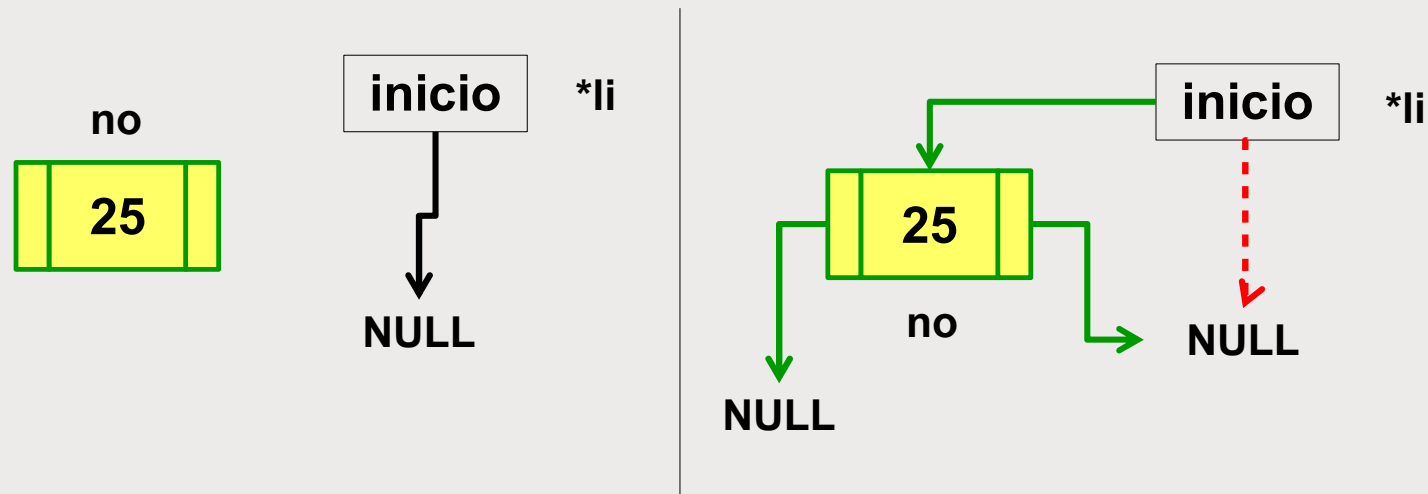
## Inserção



# LISTA DINÂMICA DUPLAMENTE ENCADEADA

## Inserção

Lista vazia

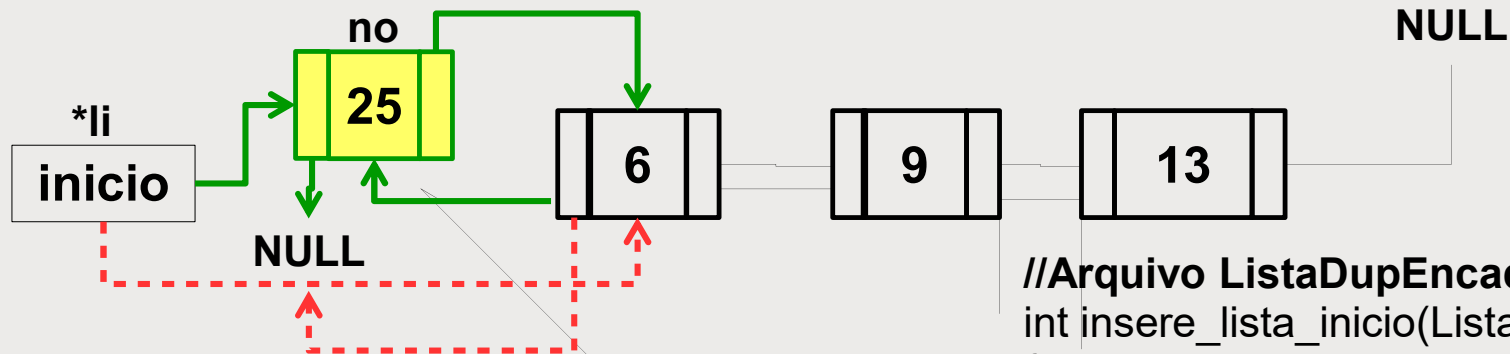


```
no->dados = al;  
no->prox = NULL;  
No->ant = NULL;  
*li = no;
```

# LISTA DINÂMICA DUPLAMENTE ENCADEADA

## Inserção

Início da lista



//Arquivo ListaDupEncadeada.h

```
int insere_lista_inicio(Lista* li, struct aluno al);
```

//main.c

```
Int x = insere_lista_inicio(li, dados_aluno)
```

//Arquivo ListaDupEncadeada.c

```
int insere_lista_inicio(Lista* li, struct aluno al)
{
```

```
    if(li == NULL) return 0;
```

```
    Elem* no = (Elem*) malloc(sizeof(Elem));
```

```
    if(no == NULL) return 0;
```

```
    no->dados = al;
```

```
    no->prox = (*li);
```

```
    no->ant = NULL;
```

```
    If (*li != NULL)
```

```
        (*li)->ant = no;
```

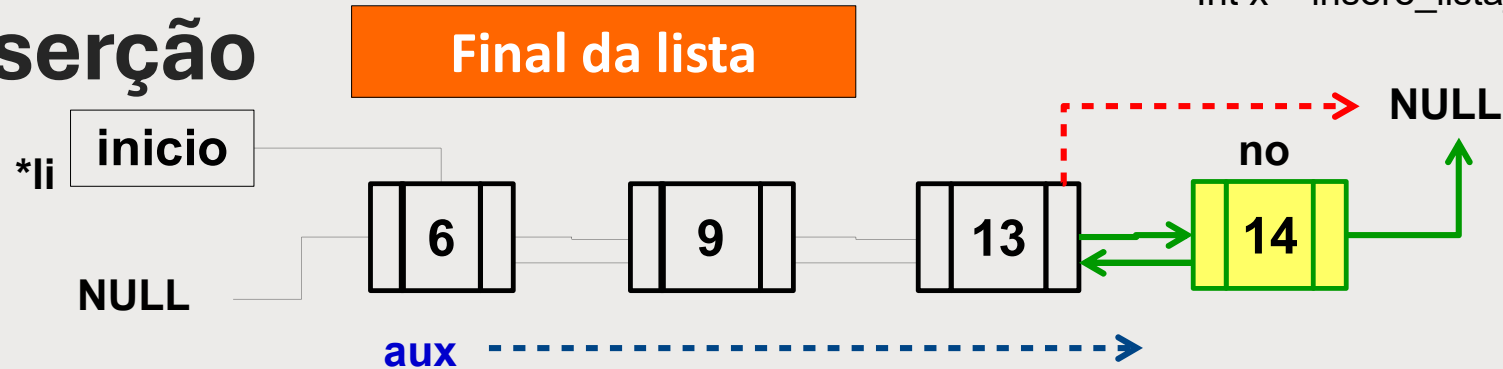
```
    *li = no;
```

```
    return 1;
```

```
}
```

# LISTA DINÂMICA DUPLAMENTE ENCADEADA

## Inserção



### //Arquivo ListaDupEncadeada.c

```
int insere_lista_final(Lista* li, struct aluno al)
{
    if(li == NULL) return 0;
    Elem* no = (Elem*) malloc(sizeof(Elem));
    if(no == NULL) return 0;
    no->dados = al;
    no->prox = NULL;
    if((*li) == NULL){ //lista vazia
        no->ant = NULL;
        *li = no;
    }
```

### //Arquivo ListaDupEncadeada.h

```
int insere_lista_final(Lista* li, struct aluno al);
```

### //main.c

```
Int x = insere_lista_final(li, dados_aluno)
```

### }else{

```
    Elem *aux = *li;
    while(aux->prox != NULL){
        aux = aux->prox;
    }
    aux->prox = no;
    no->ant = aux;
    return 1;
}
```