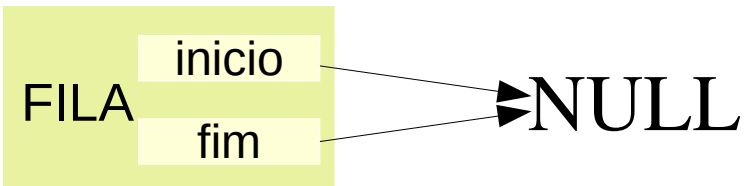
Exercício 13 – 0.Criar fila

Class fila:

Int tamanho;
noh* inicio = null;
noh* fim = null;

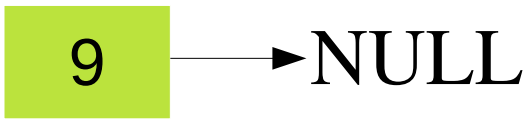
Class noh:

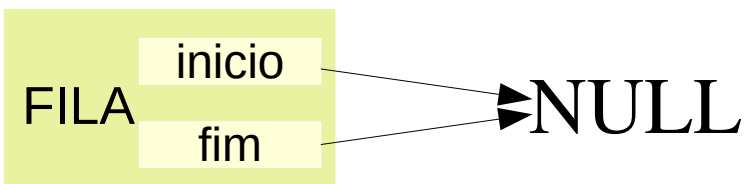
Int dado
noh* próximo;



Tamanho = 0

1. Enfileirar elementos 9,11,2





enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

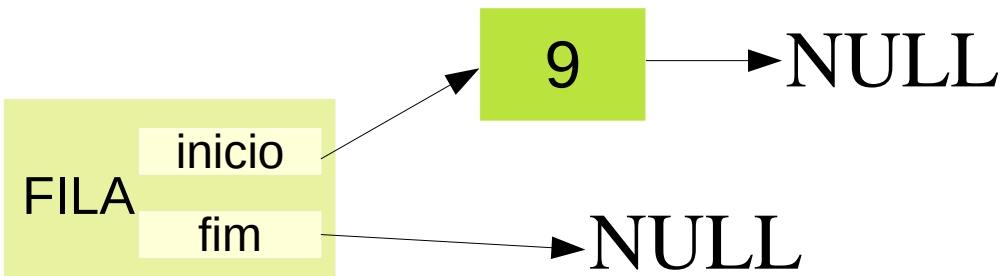
Tamanho = 0

1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 0

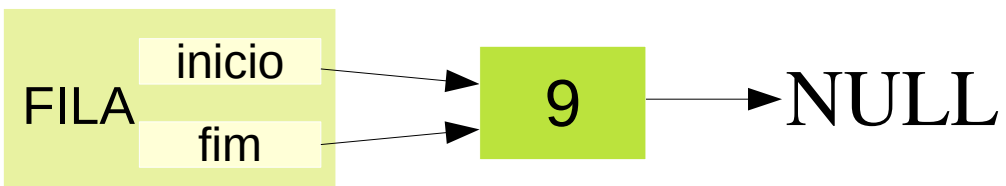


1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 0

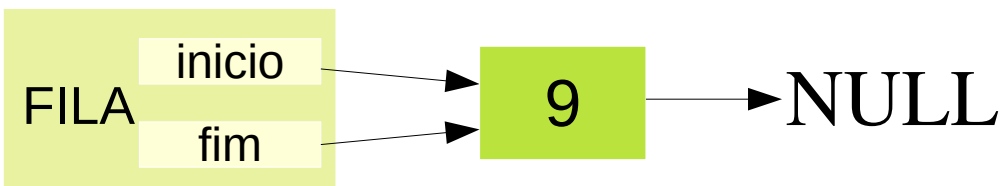


1. Enfileirar elementos 9,11,2

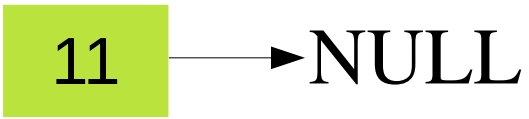
enqueue(valor):

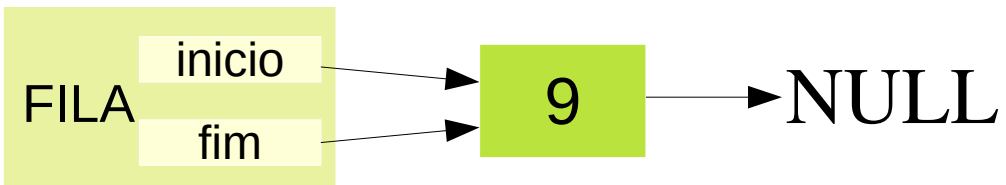
```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1



1. Enfileirar elementos 9,11,2





enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1

1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

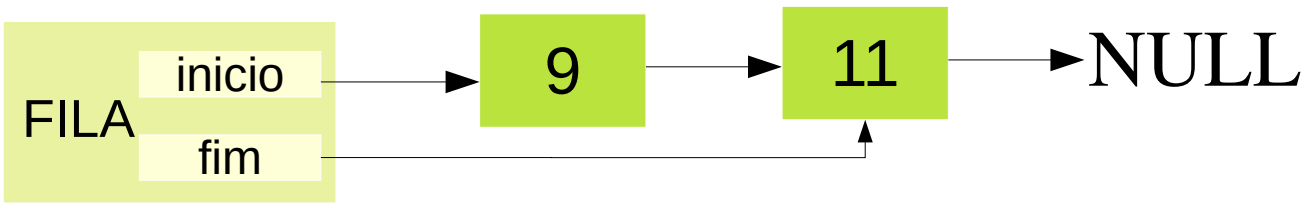
Tamanho = 1

1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1

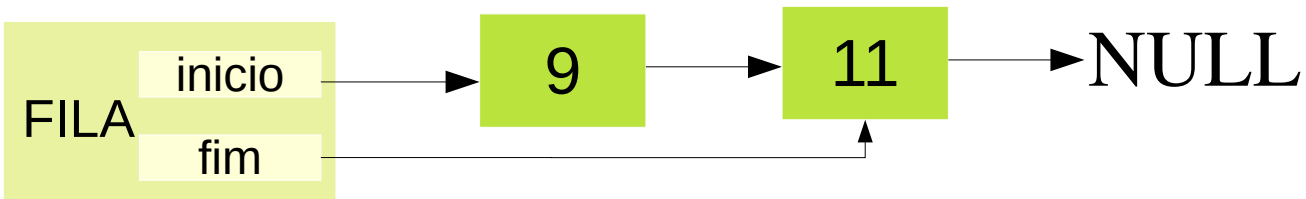


1. Enfileirar elementos 9,11,2

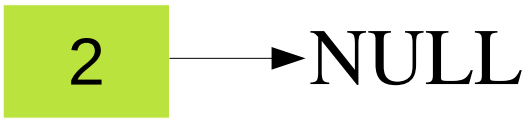
enqueue(valor):

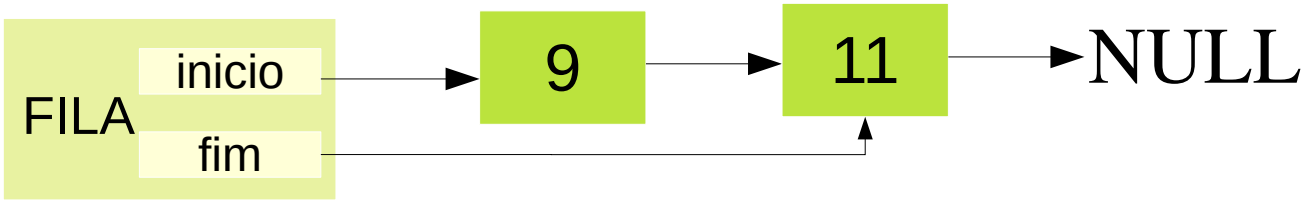
```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 2



1. Enfileirar elementos 9,11,2





enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

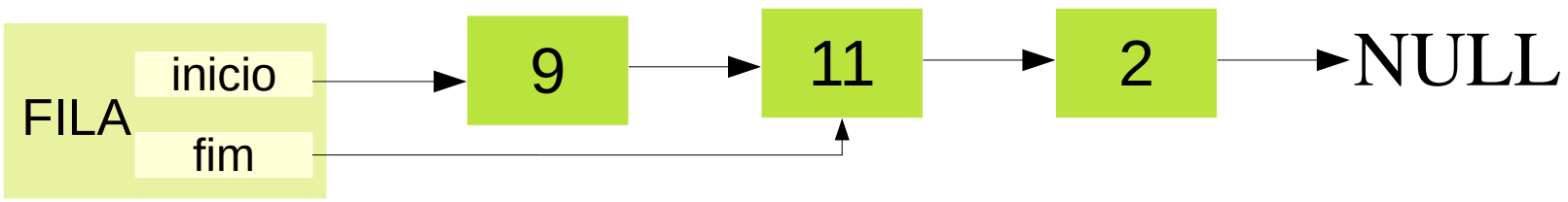
Tamanho = 2

1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 2

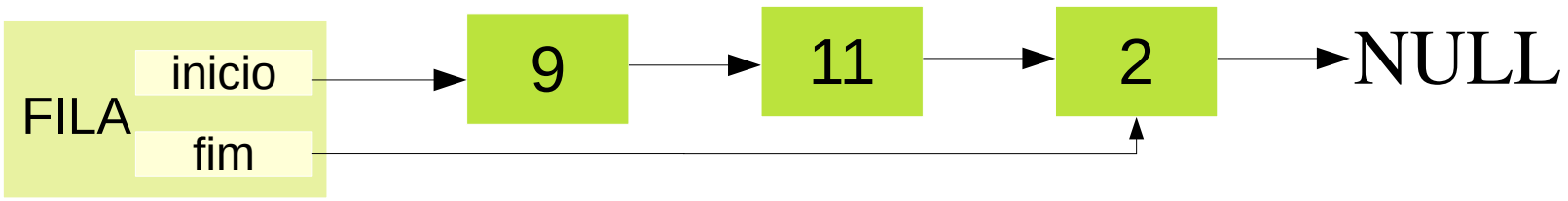


1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 2

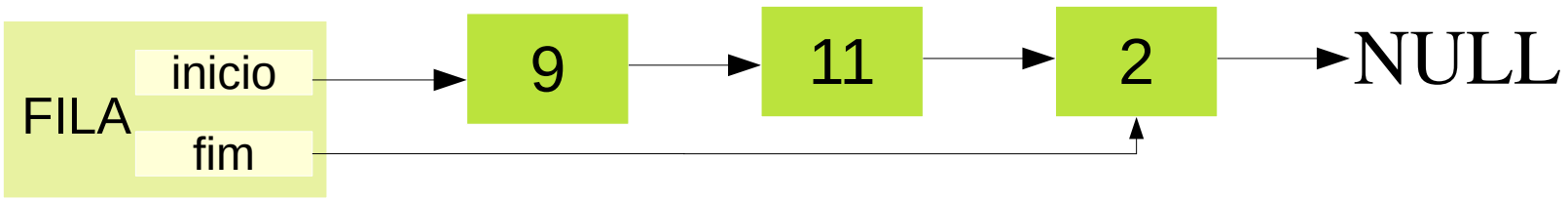


1. Enfileirar elementos 9,11,2

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 3

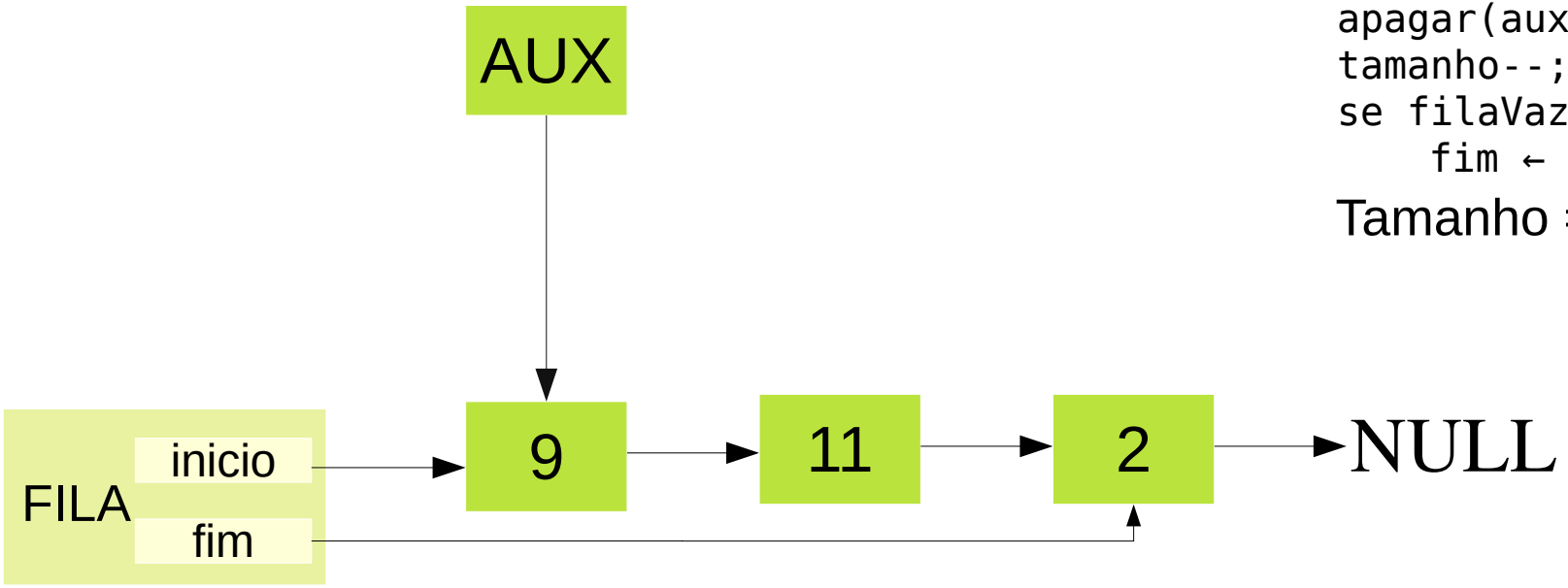


2. Desenfileirar

dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 3

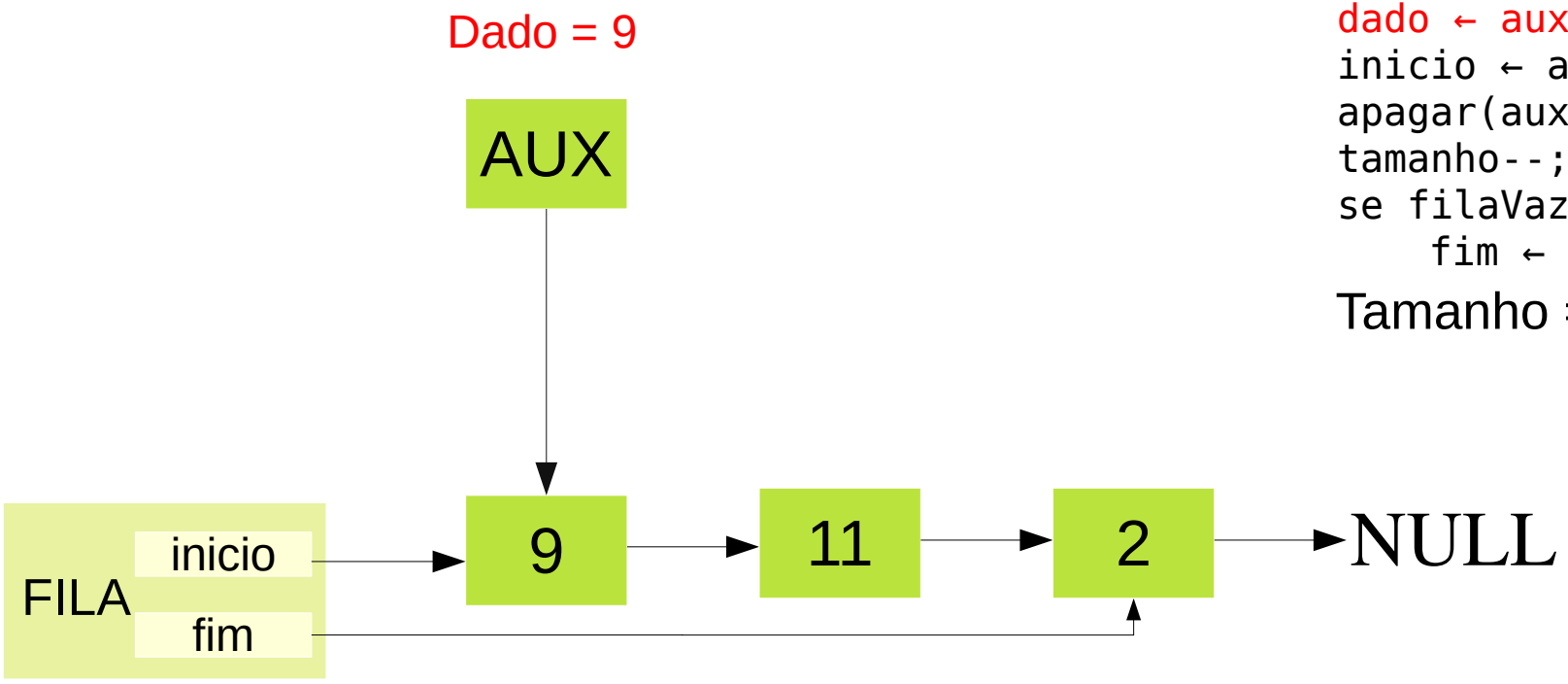


2. Desenfileirar

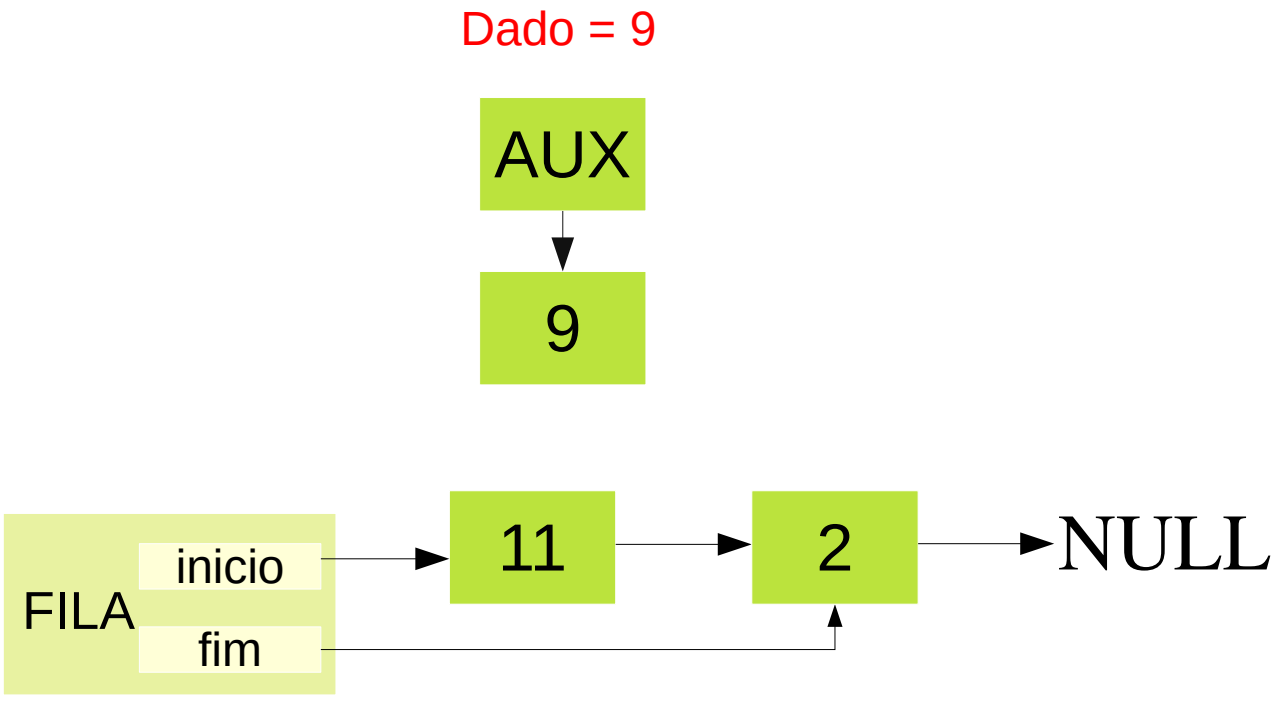
dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 3



2. Desenfileirar

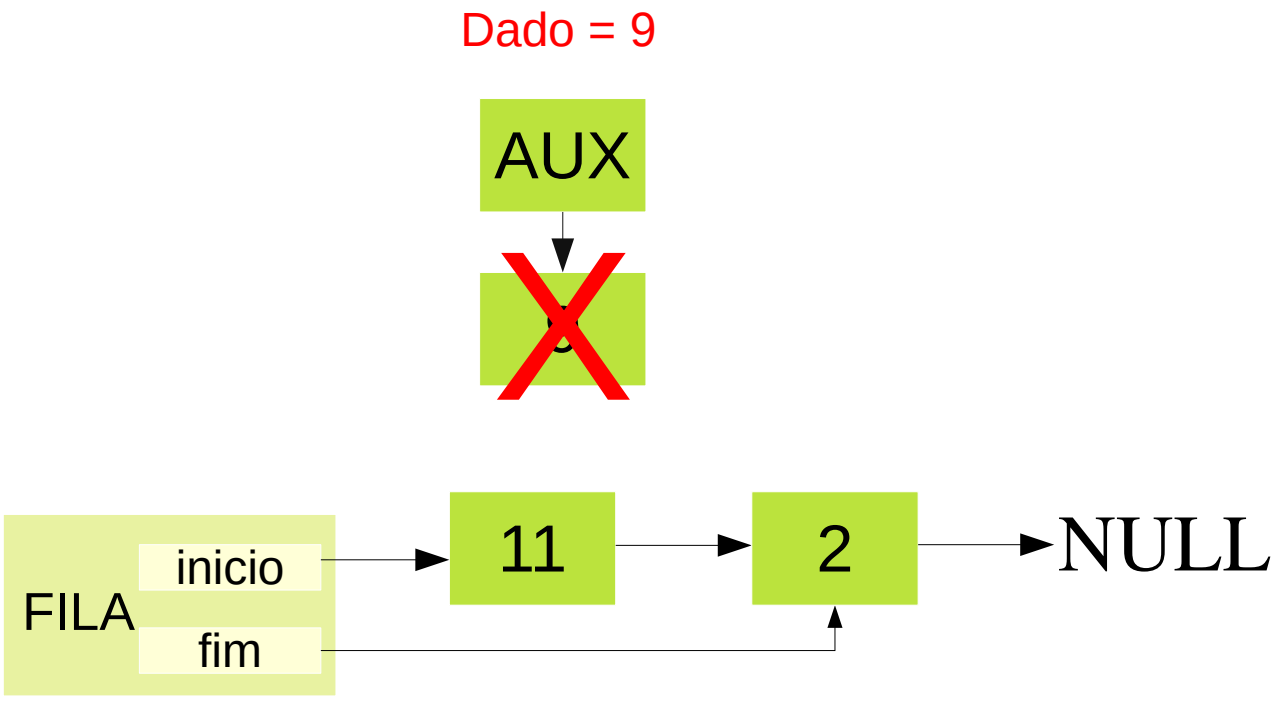


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 3

2. Desenfileirar



dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

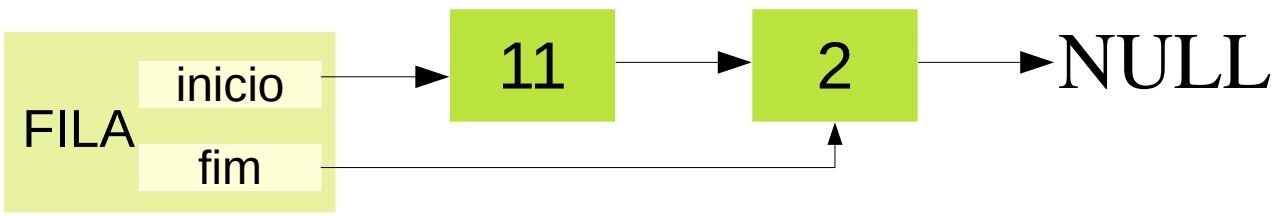
Tamanho = 3

2. Desenfileirar

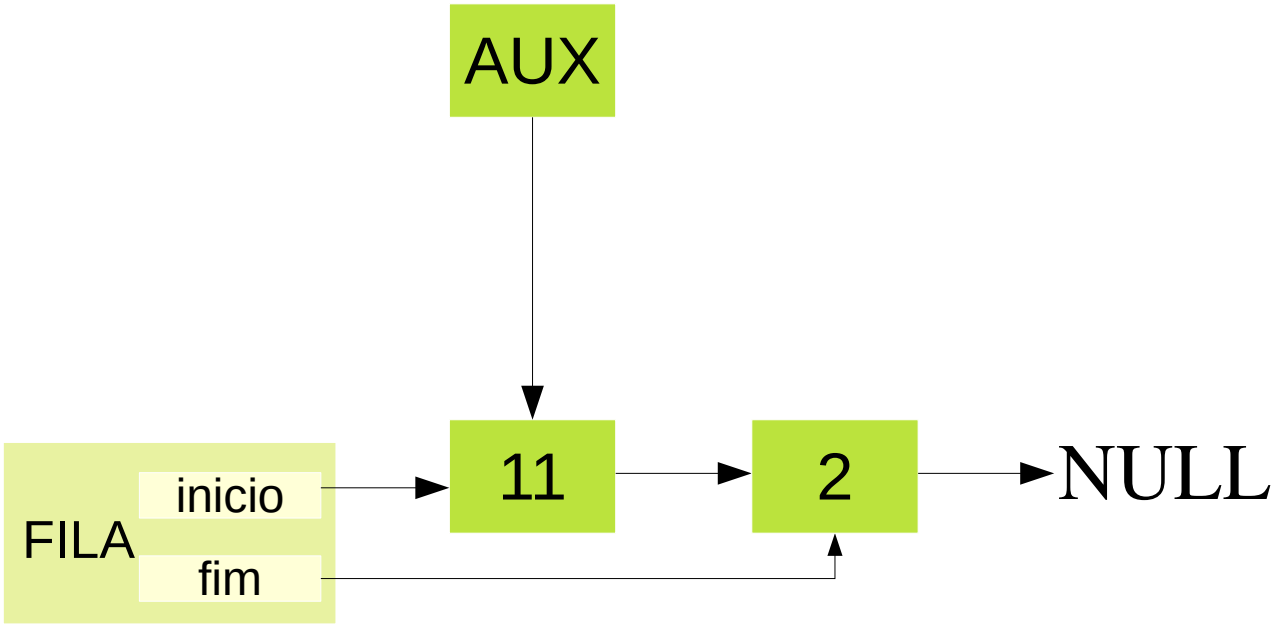
dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 2



3. Desenfileirar

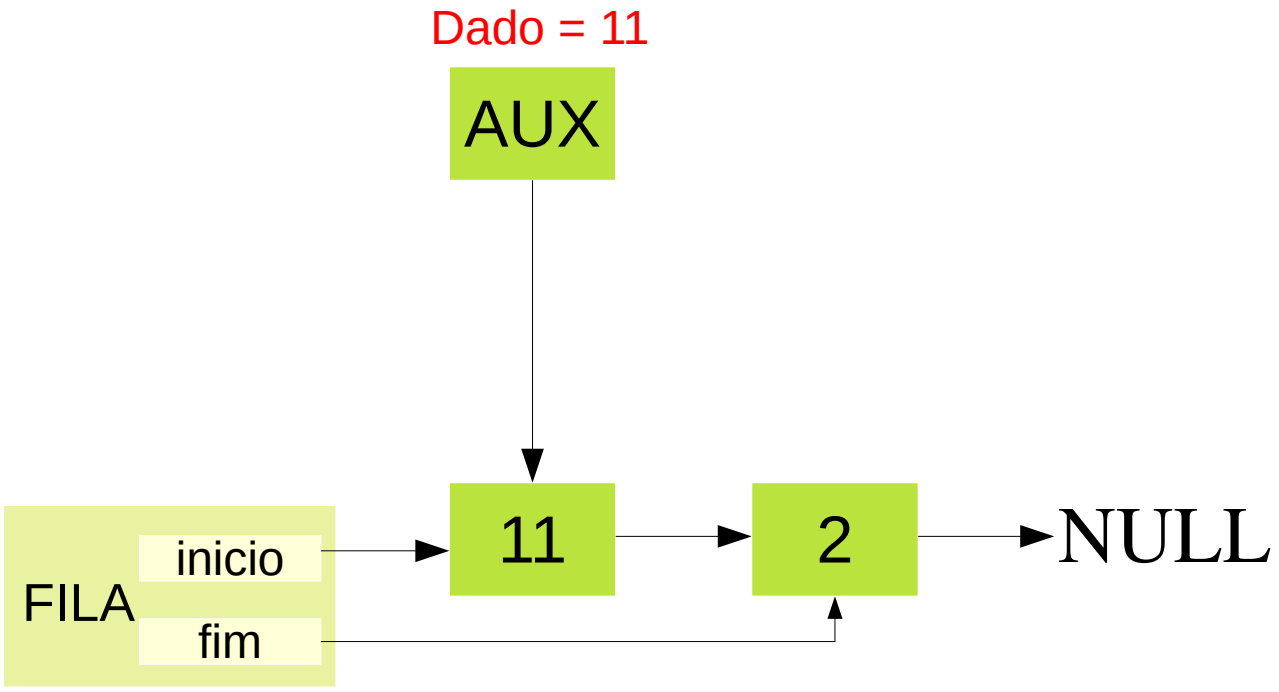


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 2

3. Desenfileirar

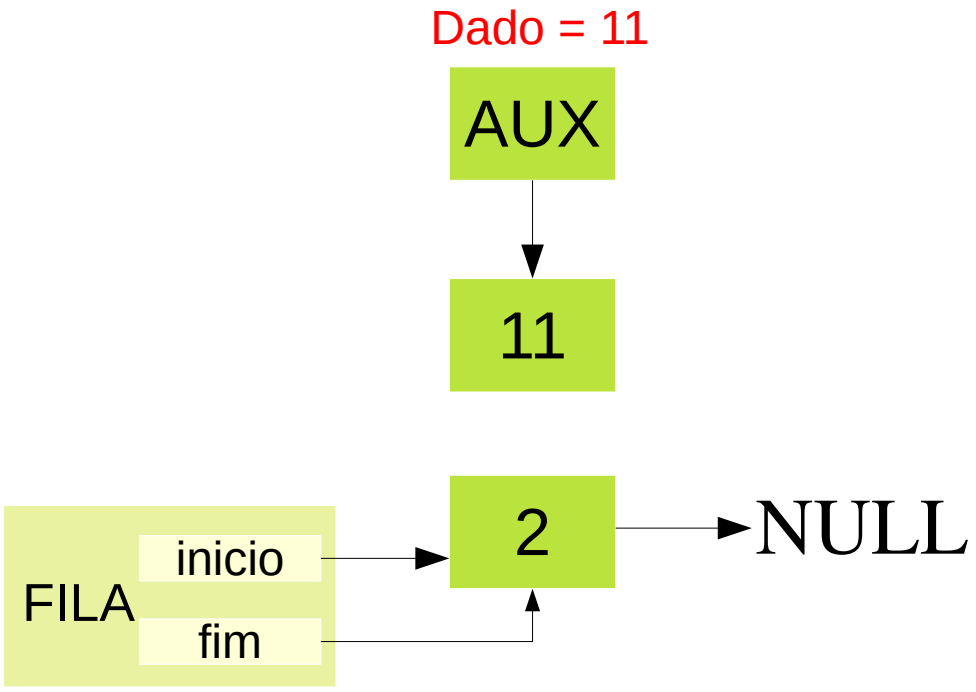


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 2

3. Desenfileirar

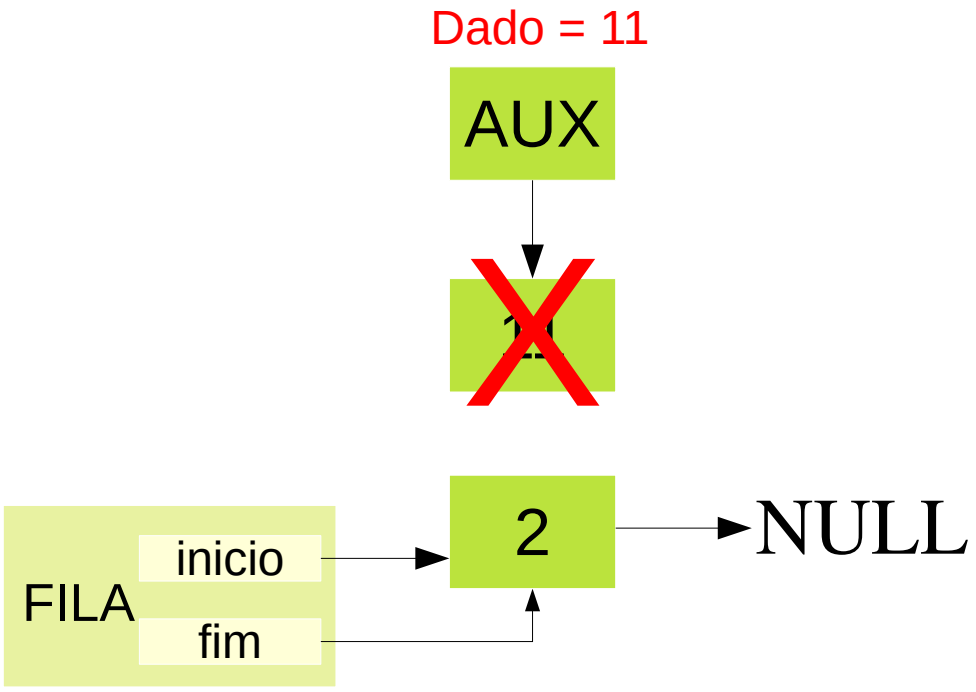


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 2

3. Desenfileirar



dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

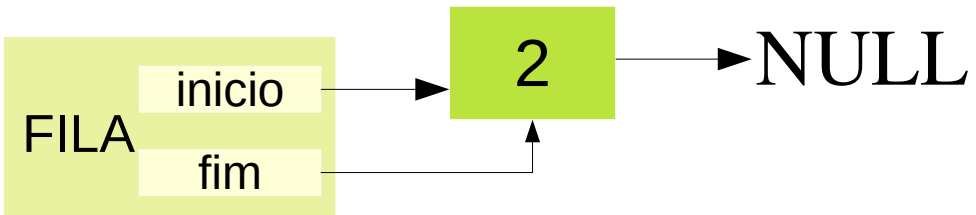
Tamanho = 2

3. Desenfileirar

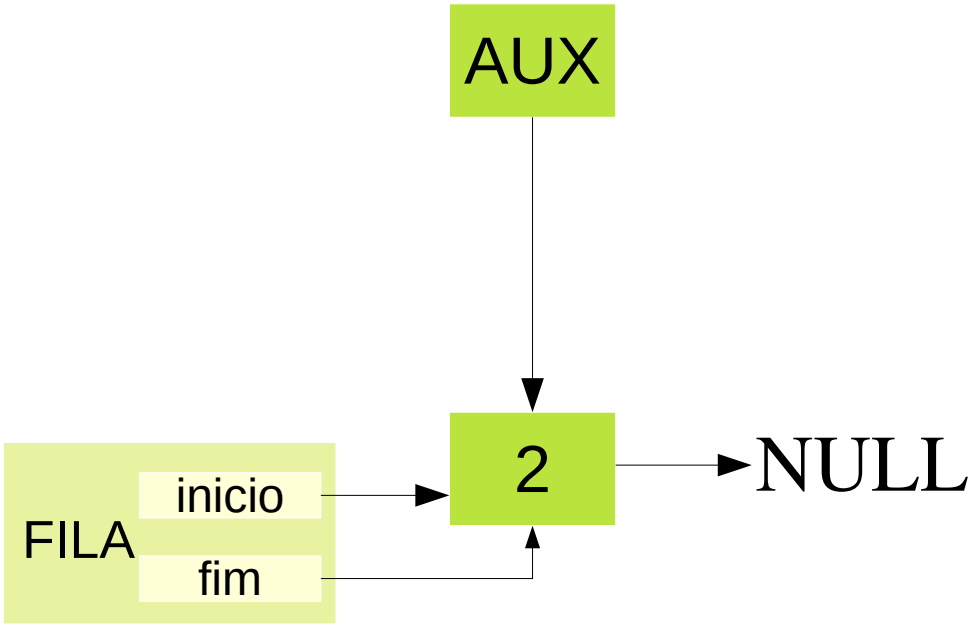
dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1



4. Desenfileirar

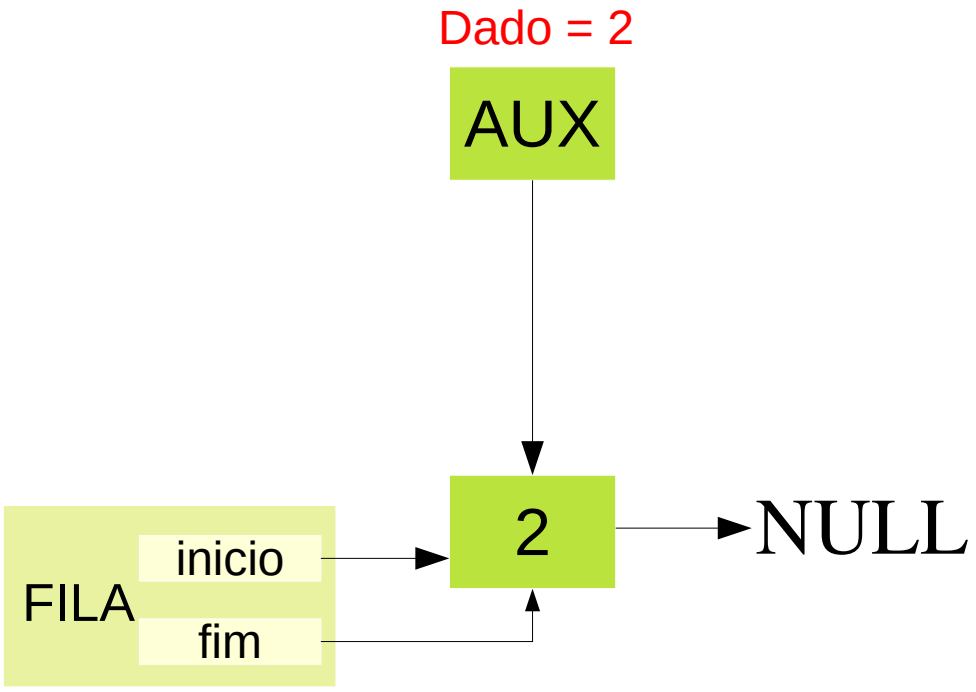


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1

4. Desenfileirar

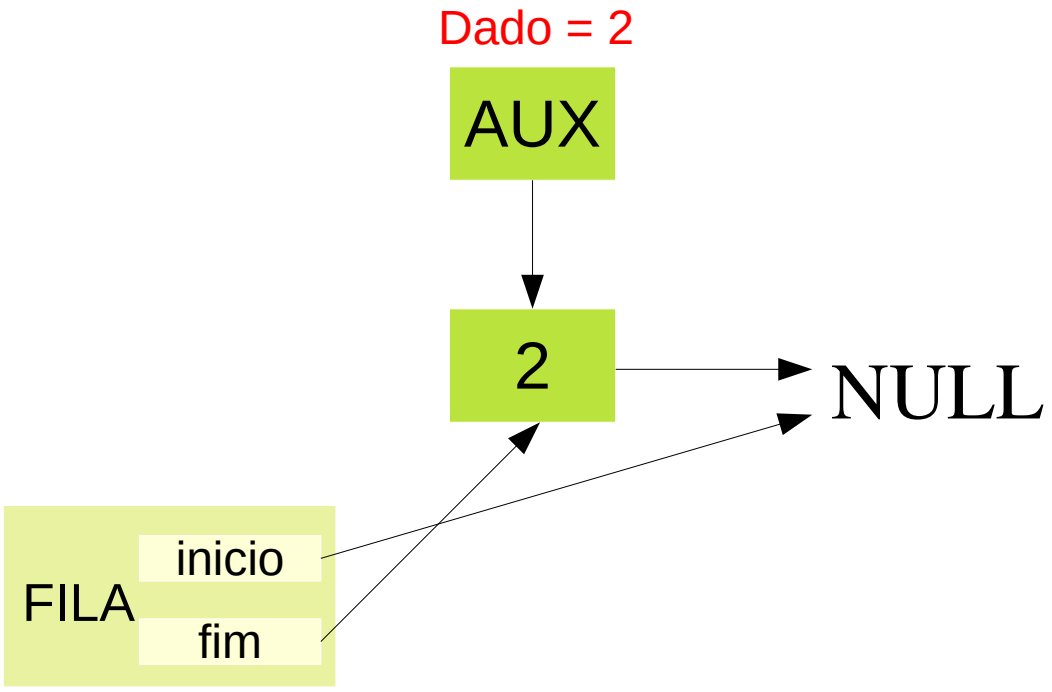


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1

4. Desenfileirar

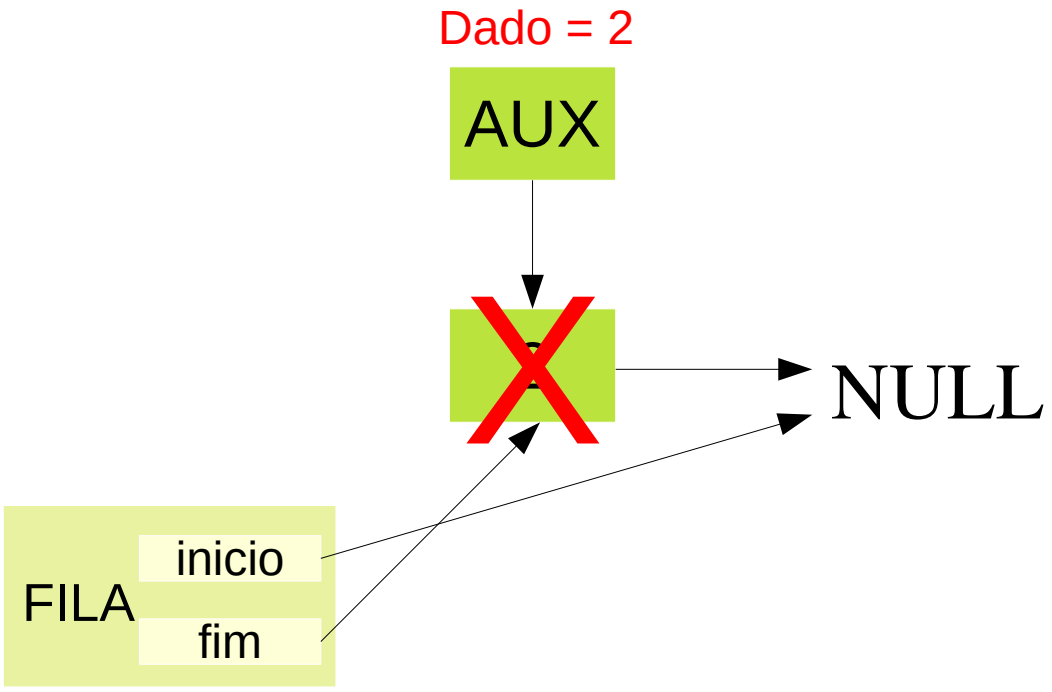


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1

4. Desenfileirar



dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

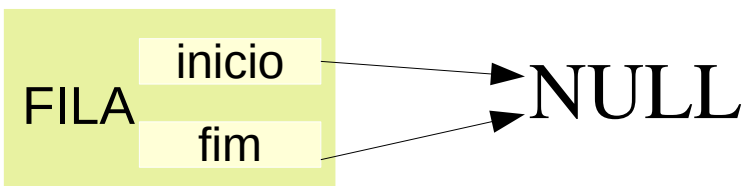
Tamanho = 0

4. Desenfileirar

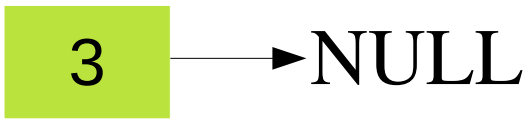
dequeue():

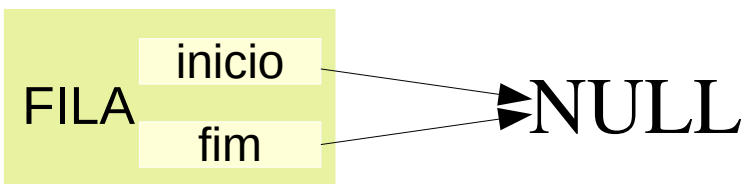
```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 0



5. Enfileirar elemento 3



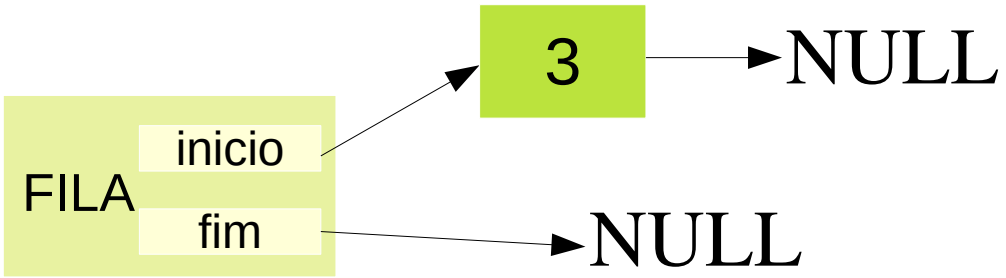


enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 0

5. Enfileirar elemento 3



enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

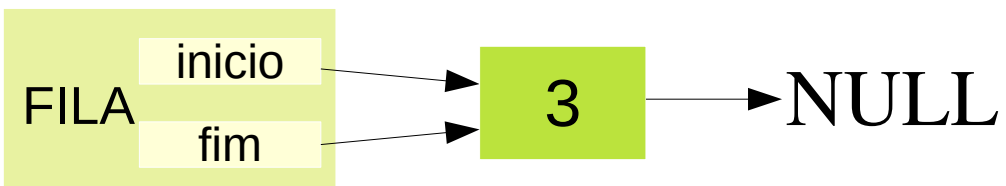
Tamanho = 0

5. Enfileirar elemento 3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 0

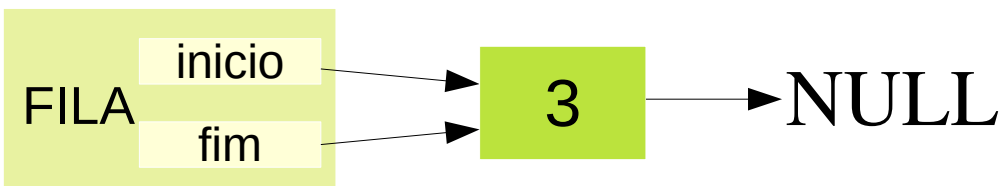


5. Enfileirar elemento 3

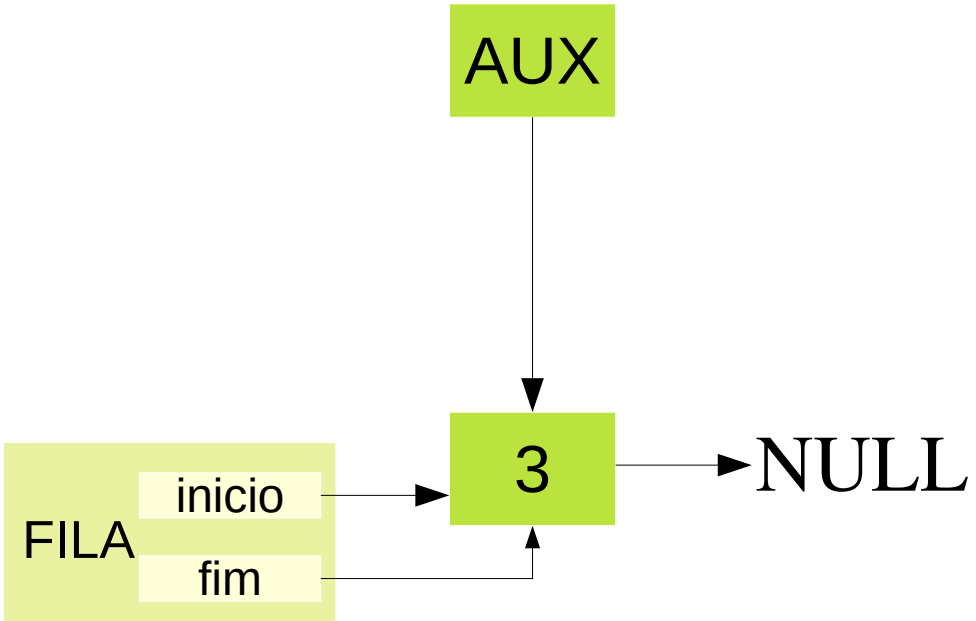
enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1



6. Desenfileirar

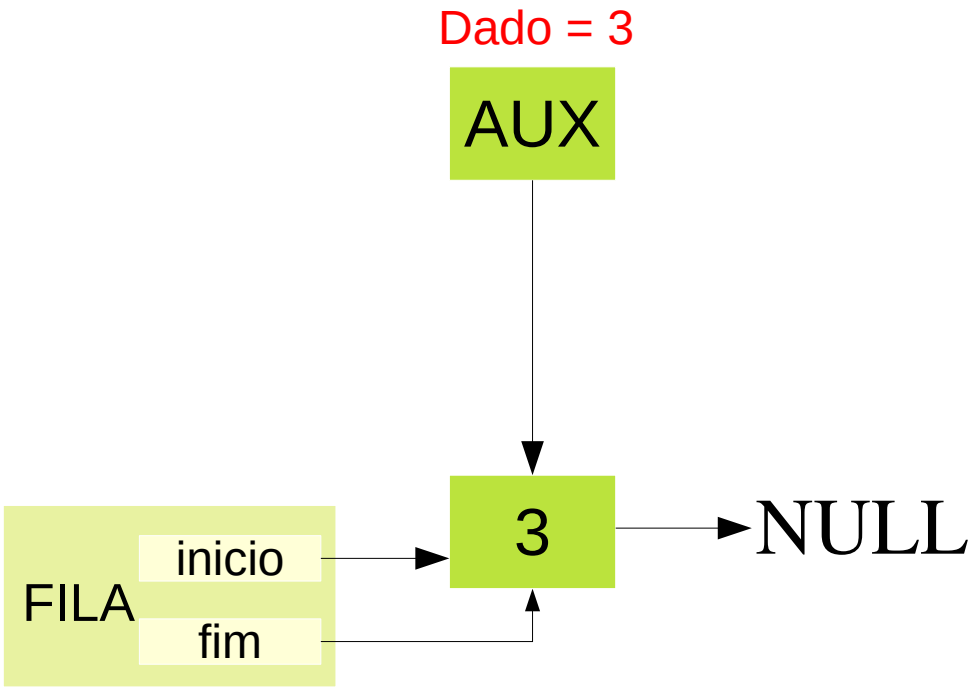


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1

6. Desenfileirar

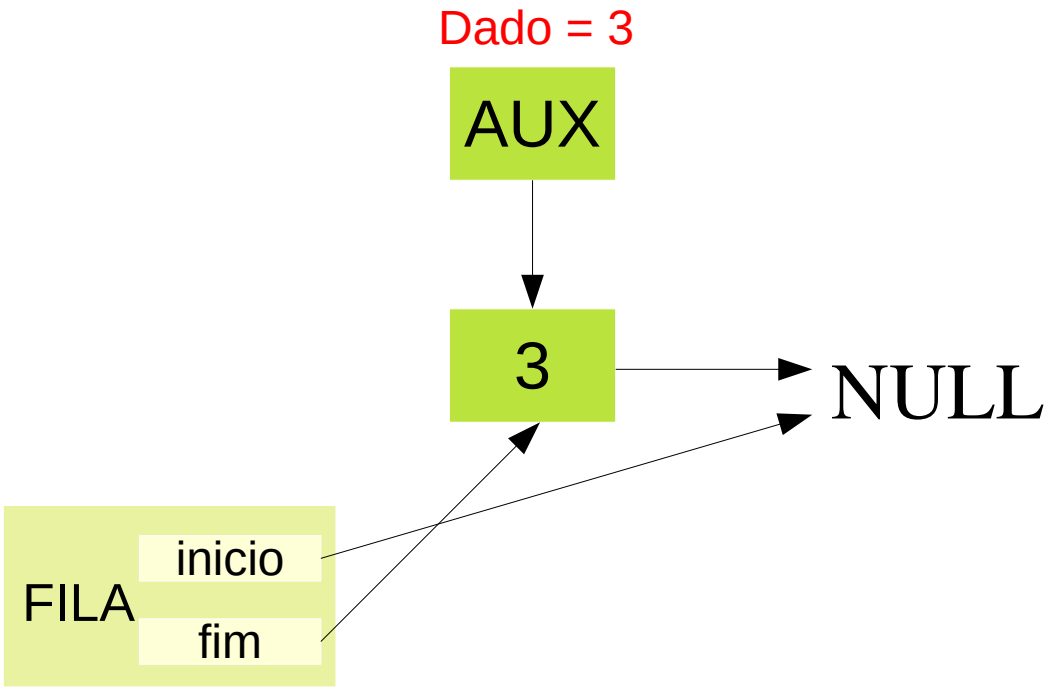


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1

6. Desenfileirar

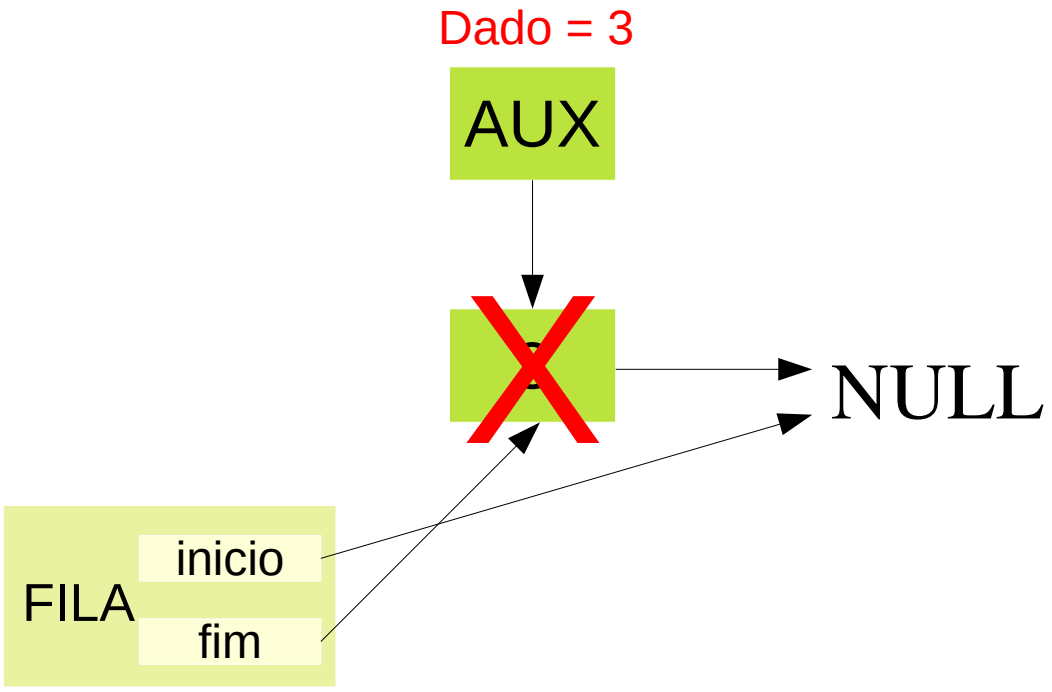


dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 1

6. Desenfileirar



dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

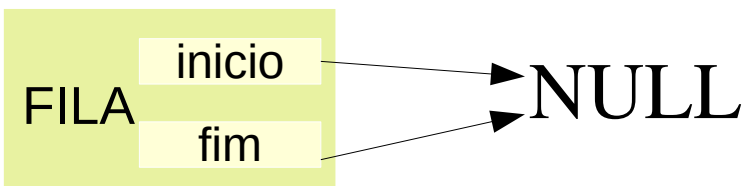
Tamanho = 0

6. Desenfileirar

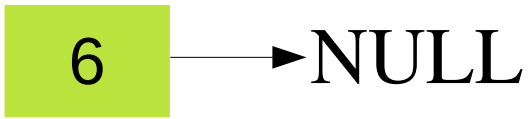
dequeue():

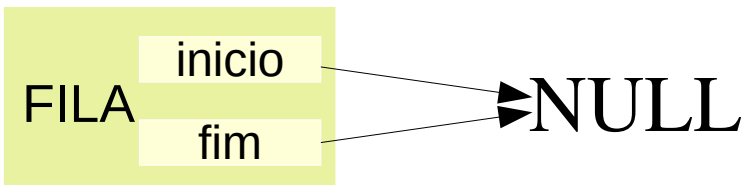
```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 0



7. Enfileirar elementos 6,8,0,3





enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

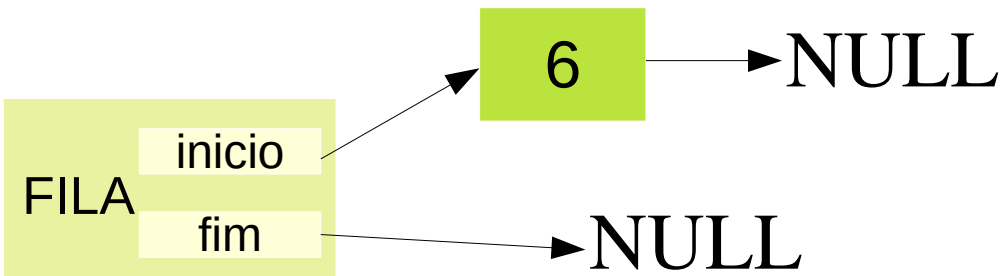
Tamanho = 0

7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 0

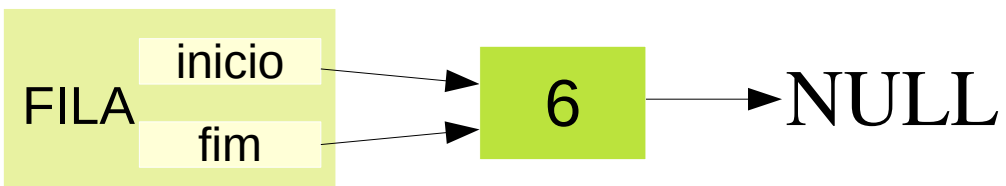


7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 0

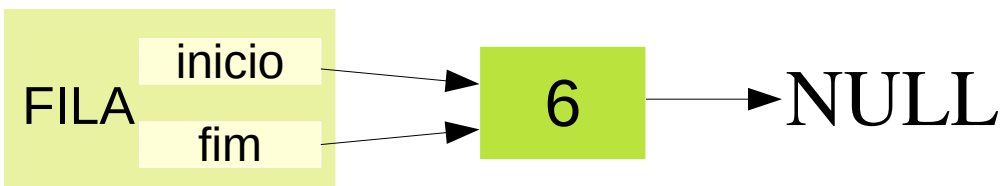


7. Enfileirar elementos 6,8,0,3

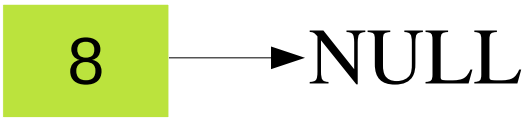
enqueue(valor):

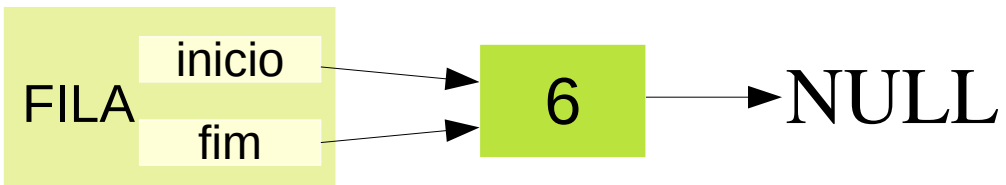
```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1



7. Enfileirar elementos 6,8,0,3





enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1

7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

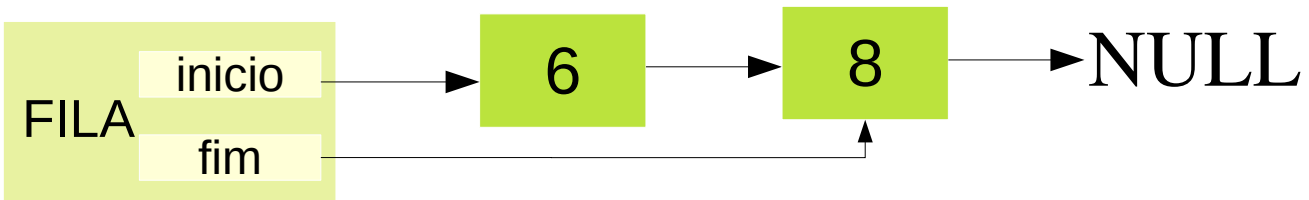
Tamanho = 1

7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 1

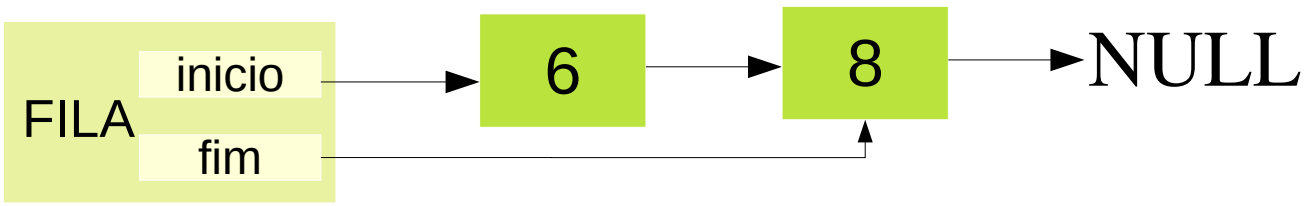


7. Enfileirar elementos 6,8,0,3

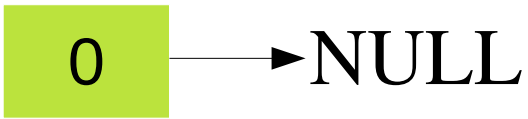
enqueue(valor):

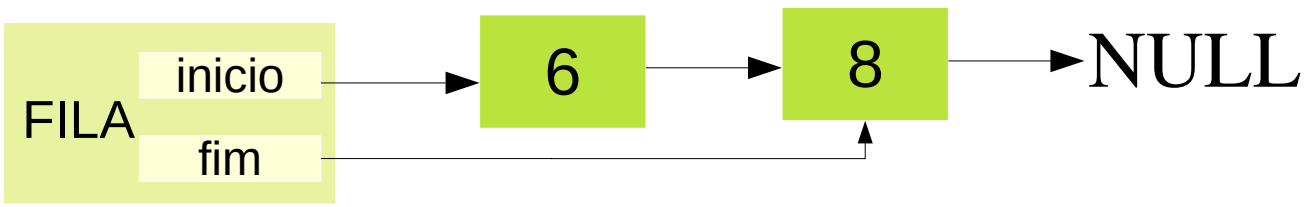
```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 2



7. Enfileirar elementos 6,8,0,3





enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

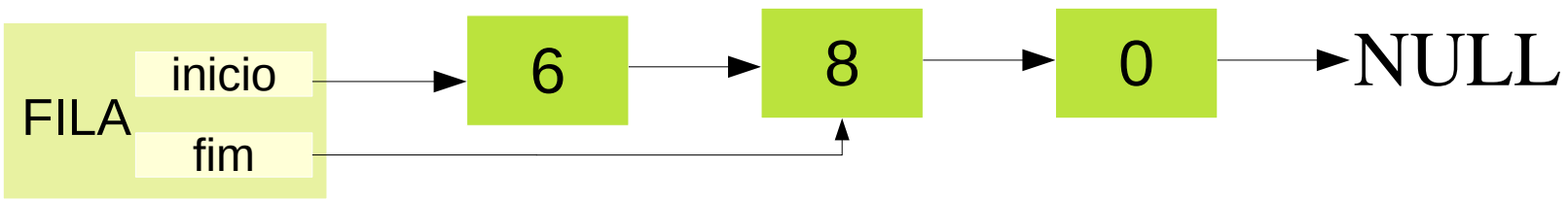
Tamanho = 2

7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 2

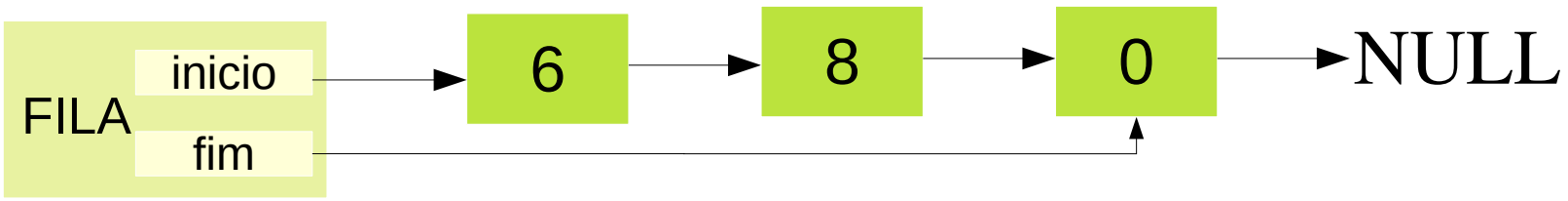


7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 2

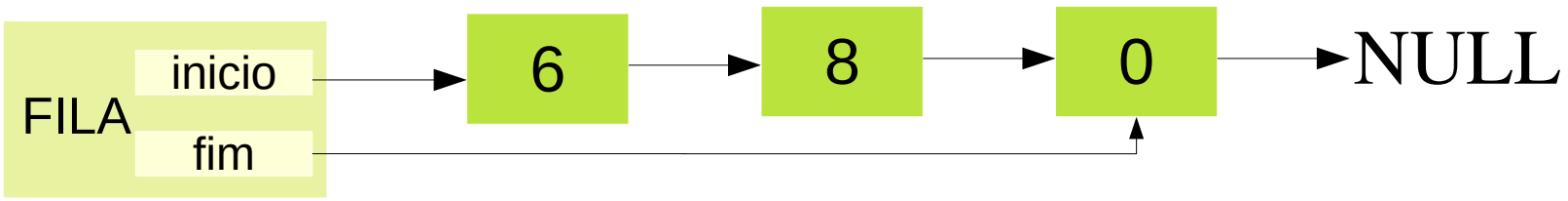


7. Enfileirar elementos 6,8,0,3

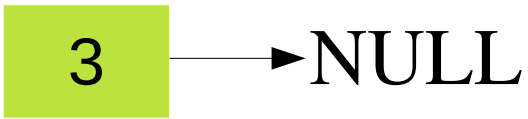
enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 3



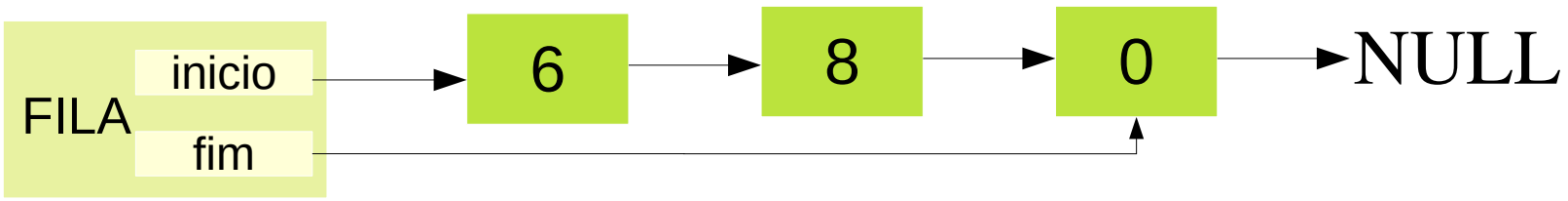
7. Enfileirar elementos 6,8,0,3



enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 3



7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

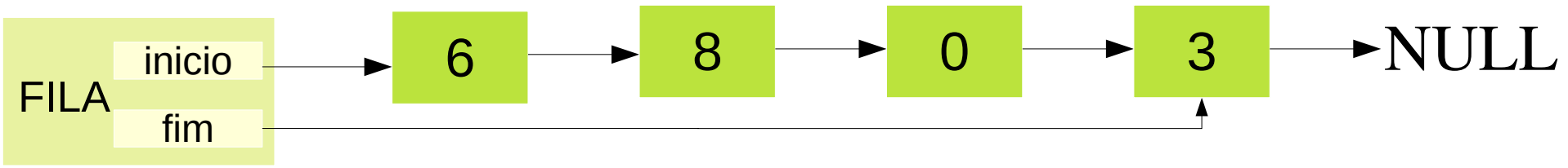
Tamanho = 3

7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 3

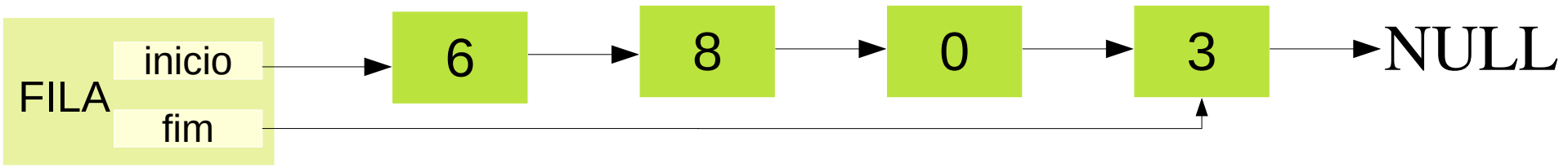


7. Enfileirar elementos 6,8,0,3

enqueue(valor):

```
novo ← criar_noh(valor);
se filaVazia()
  inicio ← novo;
senão
  fim.proximo ← novo;
fim ← novo;
tamanho++;
```

Tamanho = 4

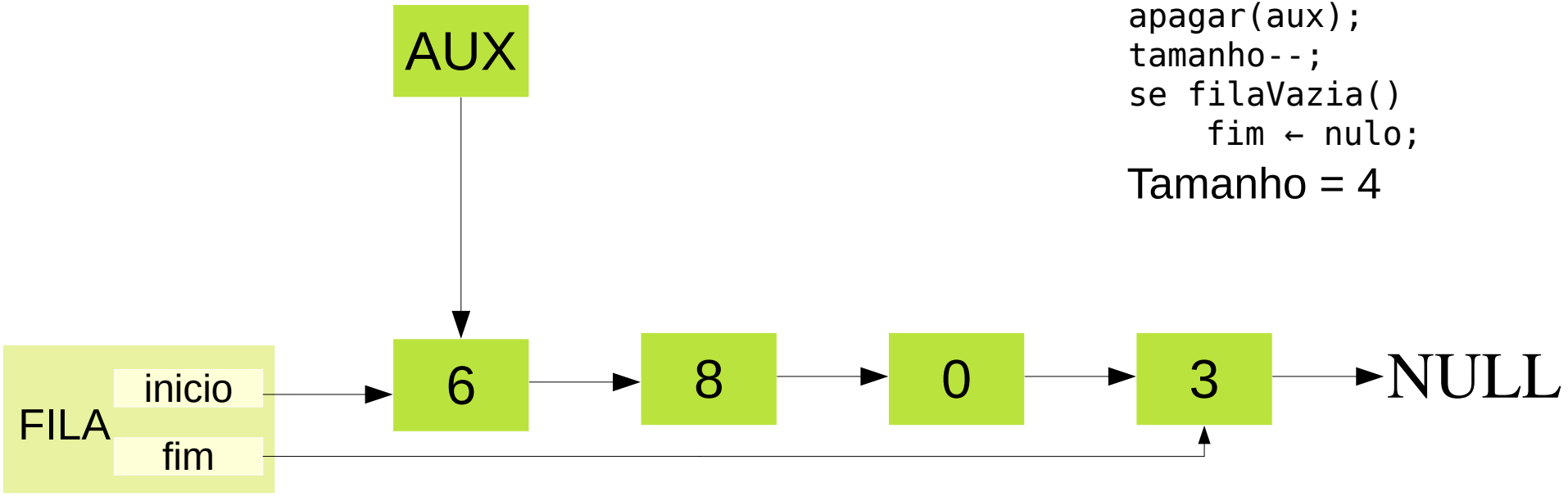


8. Desenfileirar

dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 4

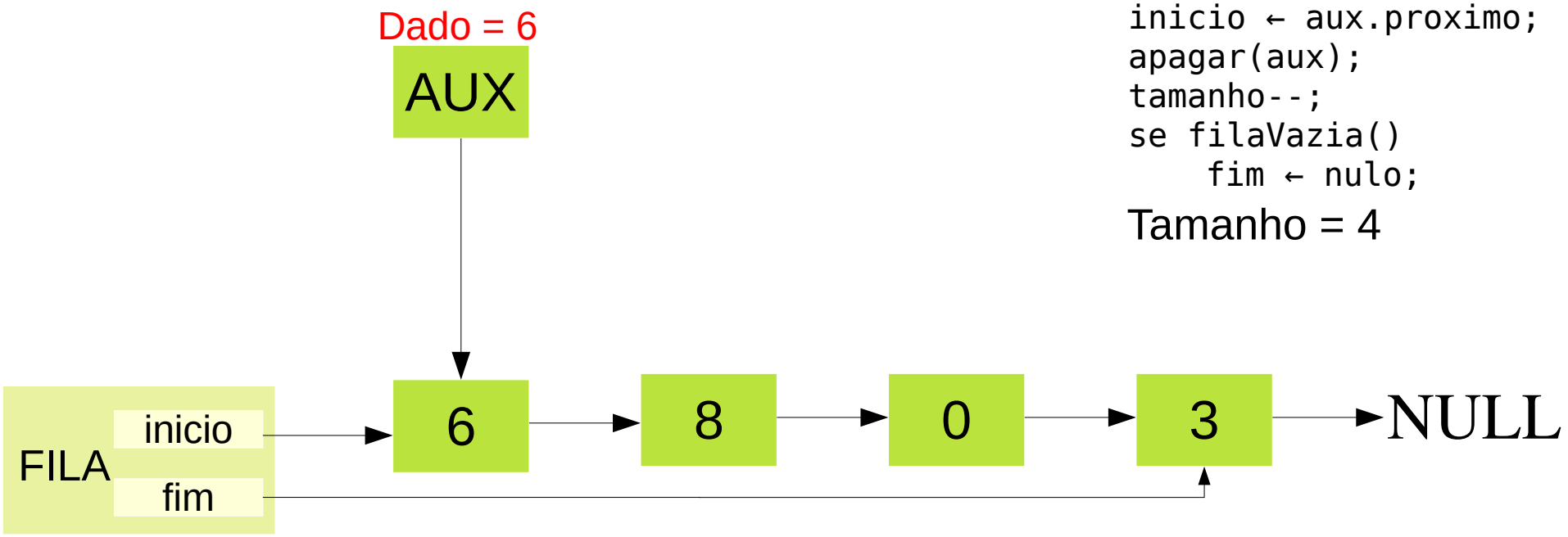


8. Desenfileirar

dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 4

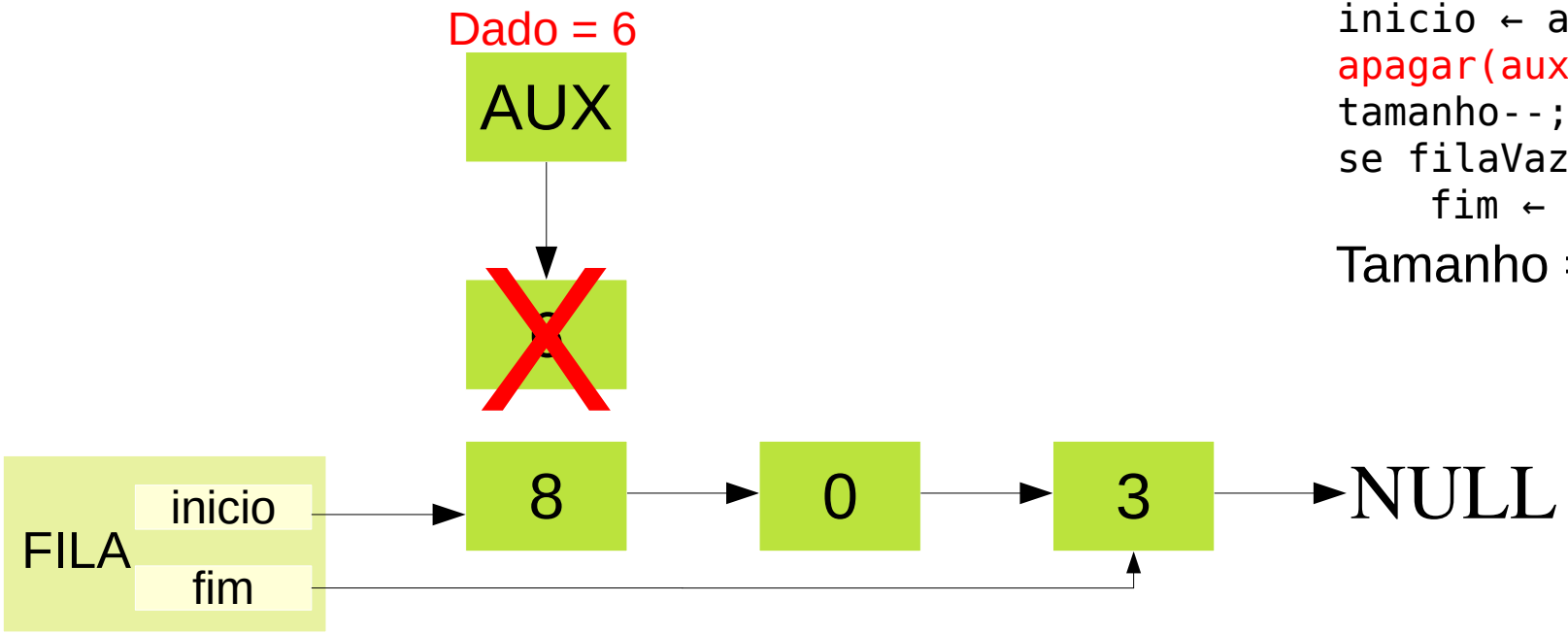


8. Desenfileirar

dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 4

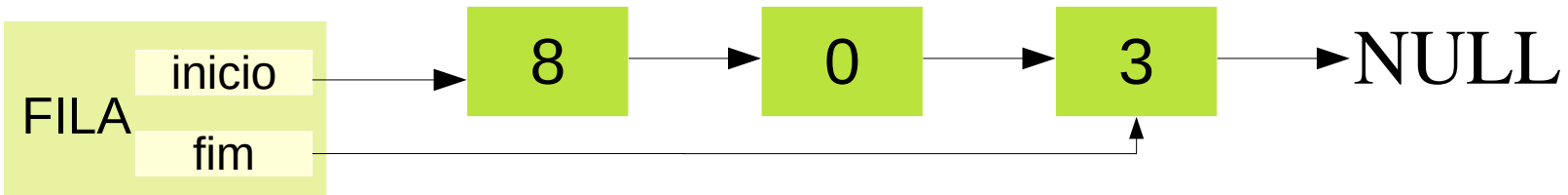


8. Desenfileirar

dequeue():

```
aux ← inicio;
dado ← aux.valor;
inicio ← aux.proximo;
apagar(aux);
tamanho--;
se filaVazia()
    fim ← nulo;
```

Tamanho = 3



Fila resultante Tamanho = 3