

Fila Duplamente Encadeada (FDE) de Prioridade contendo indicação de *Frente*, *Cauda* e um referencial móvel (*refMoveI*)

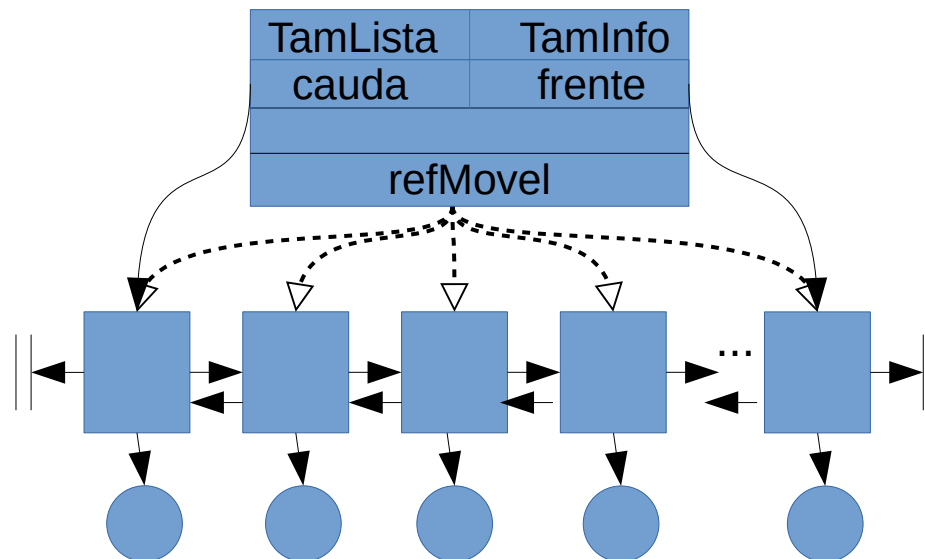
A implementação mais eficiente para uma fila de prioridade consiste no modelo chamado *Heap Priority Queue*;

Heap Priority Queue utiliza uma árvore binária, tema que ainda será abordado;

Futuramente será possível abordar a *Heap Priority Queue*;

A presente proposta tem como objetivo exercitar uma implementação com alguma melhoria de eficiência utilizando os conceitos abordados na disciplina até o presente momento.

Fila Duplamente Encadeada (FDE) de Prioridade contendo indicação de *Frente*, *Cauda* e um referencial móvel (*refMoveI*)



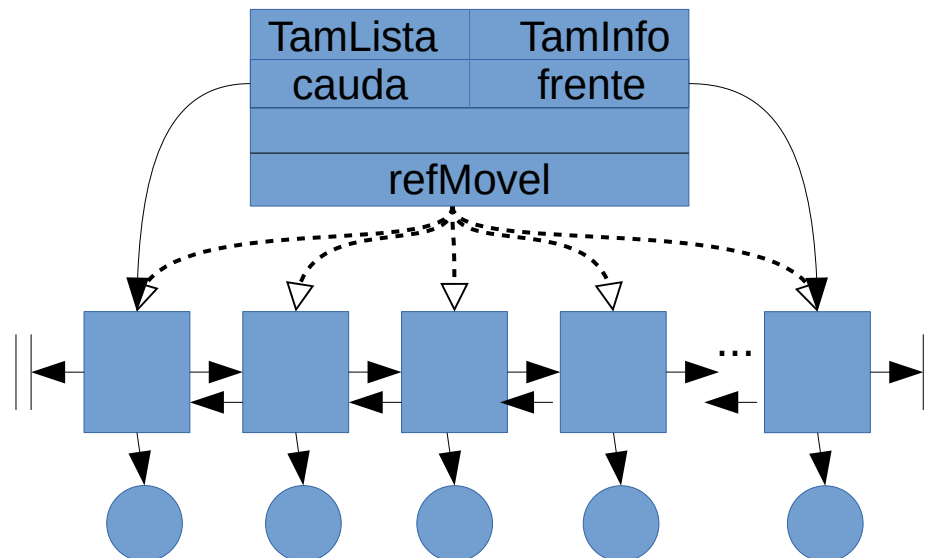
FDE de Prioridade contendo indicação de *Frente*, *Cauda* e um referencial móvel (*refMoveI*)

Frente, *Cauda* apontam para as respectivas extremidades da fila;

Posicionamento do referencial móvel (*refMoveI*):

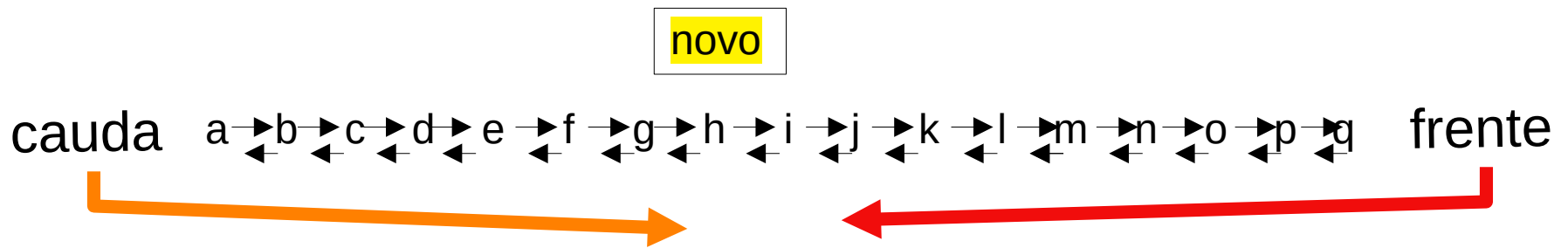
- 1) É igual a Null para a fila vazia ou...
- 2) Aponta para o endereço do elemento mais recentemente inserido.

FDE estiver vazia: *cauda* == *frente* == *refMoveI* == NULL;



Explorando a vantagem do RefMovel na FDE de prioridade:

Pela nossa implementação convencional de inserção por prioridade, o posicionamento do novo elemento é determinado ou por meio de comparativos a partir da *cauda* ou por comparativos a partir da *frente* da fila:



A ideia agora é **tirar proveito** do acréscimo do referencial móvel.

Vamos analisar os possíveis casos!

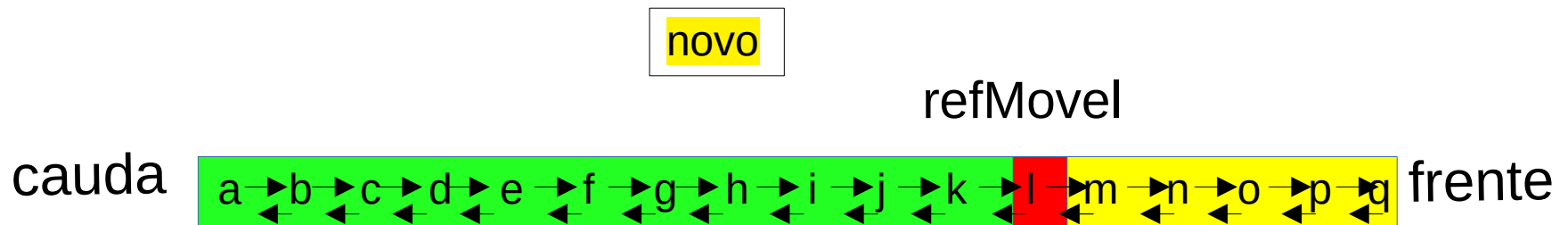
Explorando a vantagem do RefMovel na FDE de prioridade:

A ideia é **tirar proveito** do referencial móvel.

Discutiremos os casos possíveis para a inserção, considerando a idade de um indivíduo como campo chave de prioridade (quanto maior a prioridade mais à frente será a inserção).

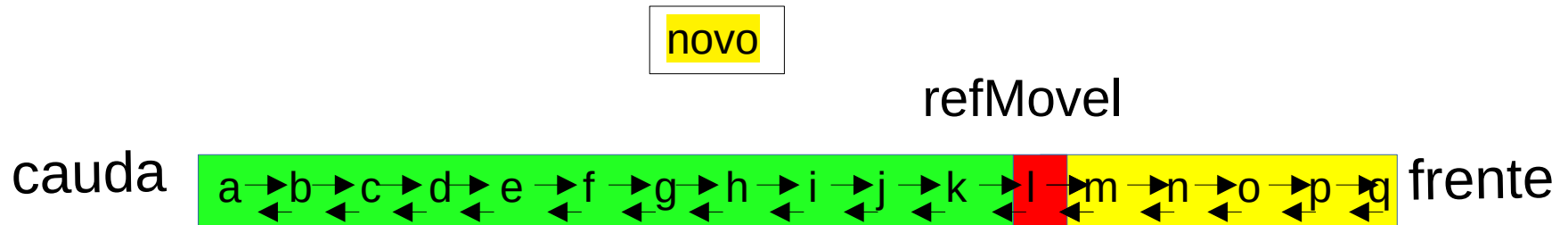
A função idade() retorna a idade armazenada no nó da fila.

Atenção: aqui as implementações são realizadas por **encadeamento**, não por um arranjo do tipo vetor.



Explorando a vantagem do RefMoveI na FDE de prioridade:

A ideia é **tirar proveito** do referencial móvel.



Caso #1

Se $(idade(novo) \leq idade(cauda))$

Então o novo elemento será inserido convencionalmente como nova cauda

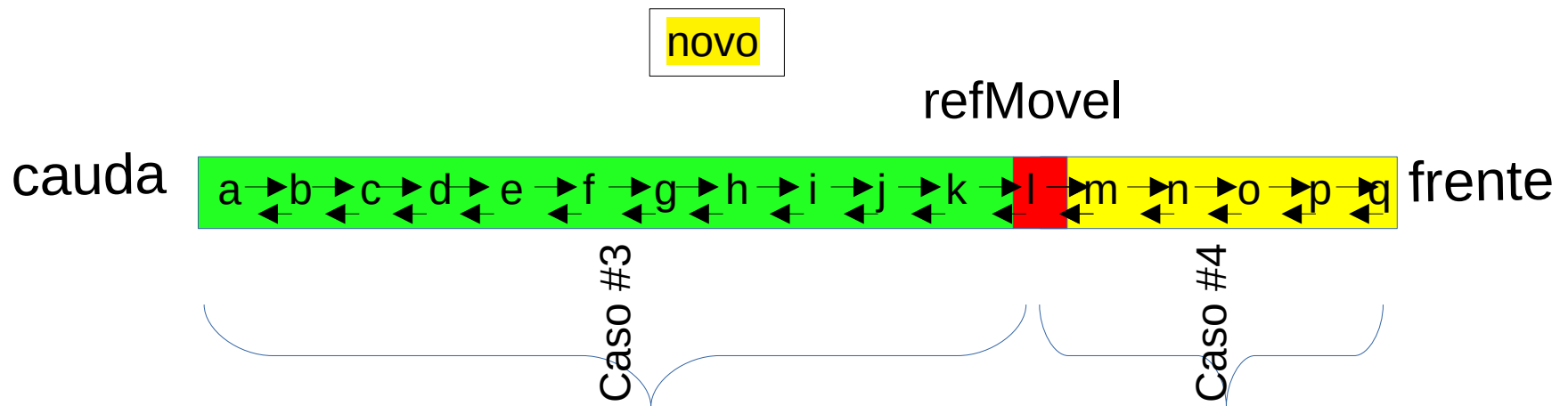
Caso #2

Se $(idade(idade(frente) < idade(novo))$

Então o novo elemento será inserido como novo item de frente

Explorando a vantagem do RefMovel na FDE de prioridade:

A ideia é **tirar proveito** do referencial móvel.



Caso #3

Se $(idade(cauda) < idade(novo) \leq idade(refMovel))$

Então a posição do novo elemento estará entre *cauda* e *refMovel*

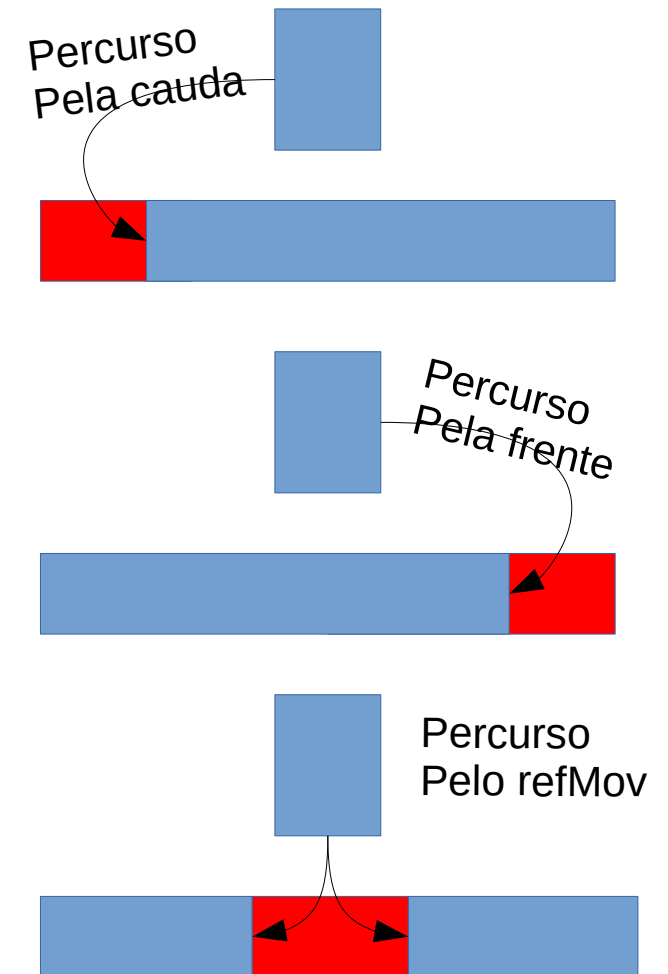
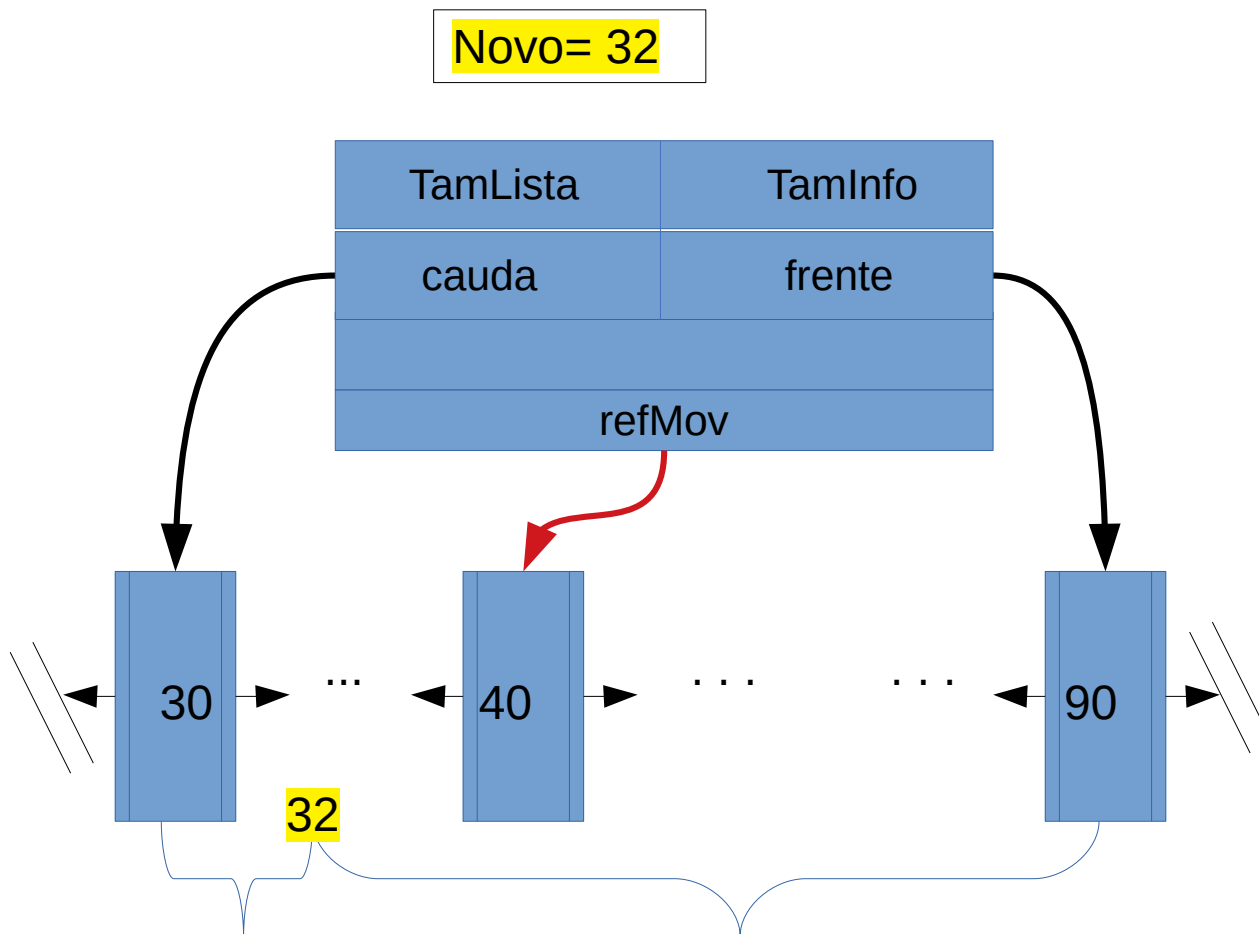
Caso #4

Se $(idade(refMovel) < idade(novo) \leq idade(frente))$

Então a posição do novo elemento estará entre *refMovel* e *frente*

Explorando a vantagem do RefMovel na FDE de prioridade:

Aprimorando um pouco mais: o caminho de busca é determinado pela proximidade entre a chave de prioridade do *novo* e a chave de prioridade de algum dos referenciais (*cauda*, *refMov* ou *frente*).



Explorando a vantagem do RefMoveI na FDE de prioridade:

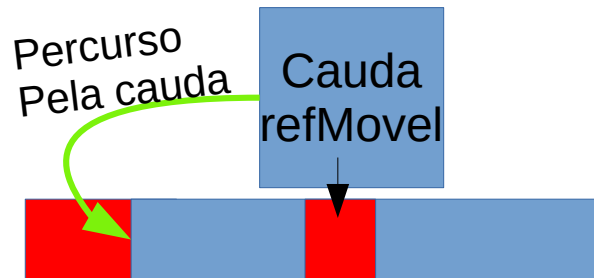
Caso#3:

Se $(idade(cauda) \leq idade(novo) \leq idade(refMoveI))$

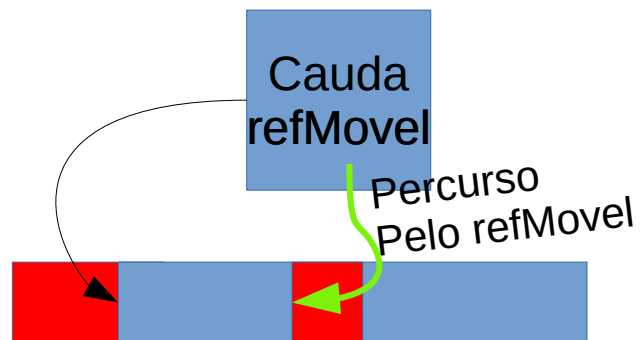
$$\Delta_a = | idade(cauda) - idade(novo) |$$

$$\Delta_b = | idade(refMoveI) - idade(novo) |$$

Se $\Delta_a \leq \Delta_b$: localize a posição do novo pela cauda



Senão:



Explorando a vantagem do RefMoveI na FDE de prioridade:

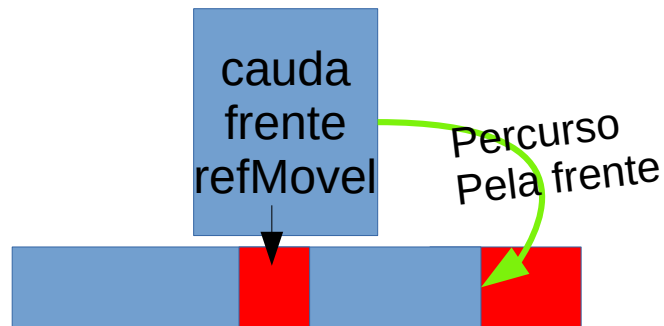
Caso#4:

Se $(idade(refMoveI) \leq idade(novo) \leq idade(frente))$

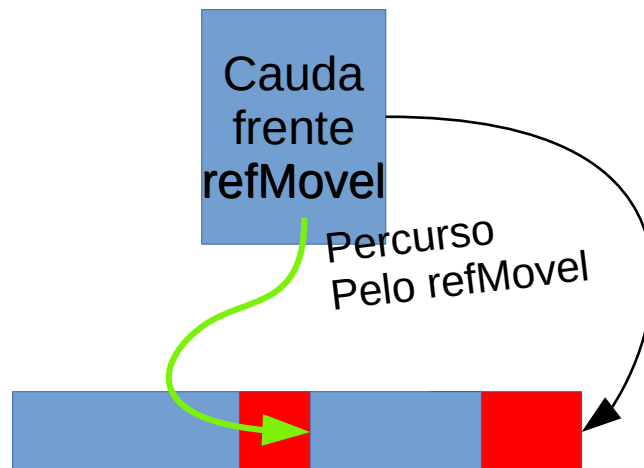
$$\Delta_c = | idade(frente) - idade(novo) |$$

$$\Delta_d = | idade(refMoveI) - idade(novo) |$$

Se $\Delta_c \leq \Delta_d$: localize a posição do novo pela frente

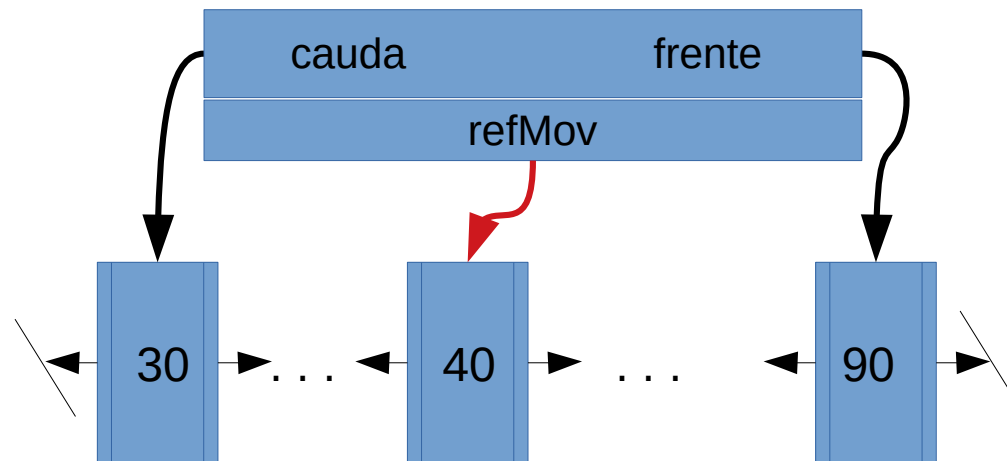


Senão:



A seguir serão feitas algumas simulações explorando a “RefMovel” na FDE de prioridade:

Nos desenhos, considere que a sequência de itens é duplamente encadeada, conforme a figura abaixo:

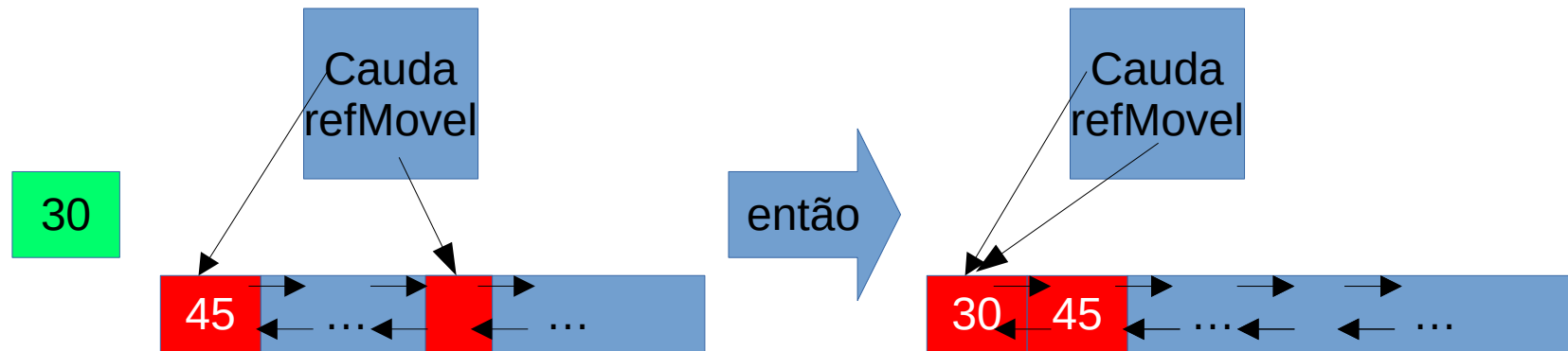


SIMULAÇÃO-1A:

Inserindo 30

Caso #1: $\text{idade}(\text{novo}) \leq \text{idade}(\text{cauda})$

novo elemento será inserido convencionalmente como nova cauda

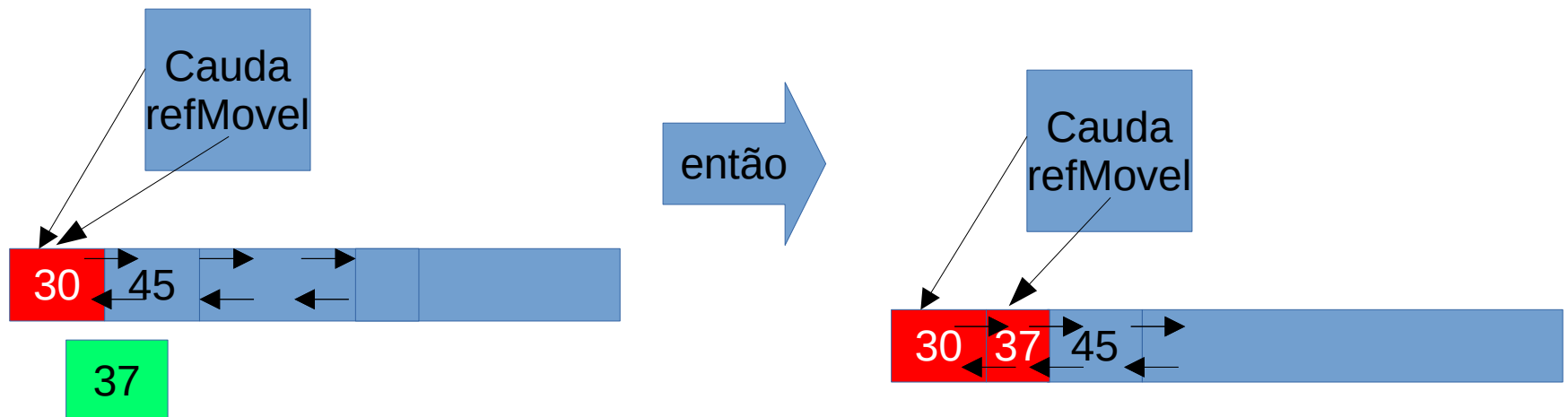


Inserindo 37

Caso#3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMovel})$

$$\Delta_a = |\text{idade}(\text{cauda}) - \text{idade}(\text{novo})| \quad \Delta_b = |\text{idade}(\text{refMovel}) - \text{idade}(\text{novo})|$$

Se $\Delta_a \leq \Delta_b$ (de fato $\Delta_a == \Delta_b$), então: localize a posição do novo nó pela cauda com um laço para a direita!



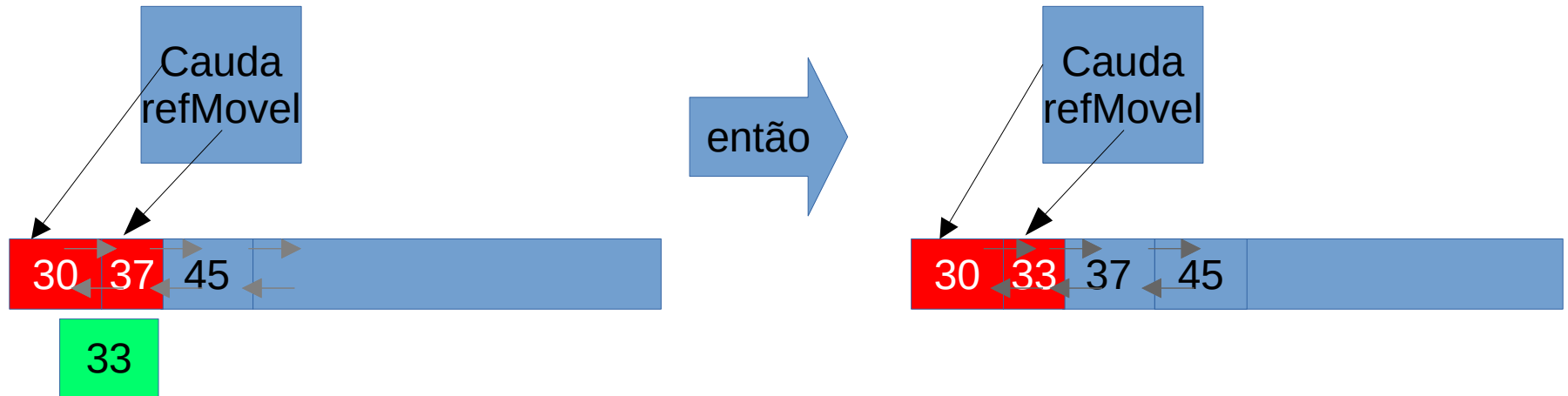
SIMULAÇÃO-1B:

Inserindo 33

Caso#3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMoveI})$

$$\Delta_a = |30 - 33| = 3 \quad \Delta_b = |37 - 33| = 4$$

Se $\Delta_a \leq \Delta_b$, então: localize a posição do novo pela cauda

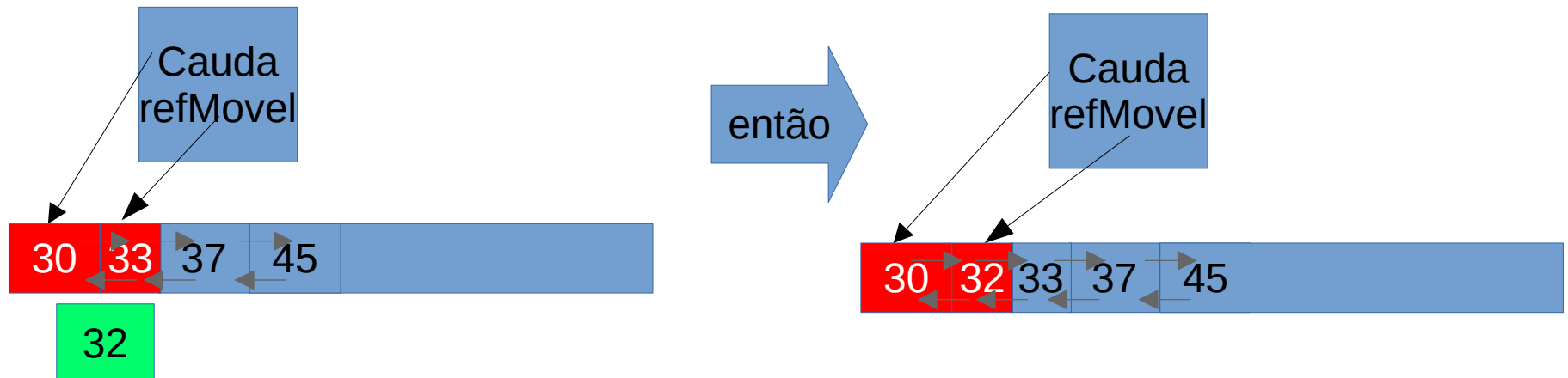


Inserindo 32

Caso#3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMoveI})$

$$\Delta_a = |30 - 32| = 2 \quad \Delta_b = |33 - 32| = 1$$

$\Delta_a \leq \Delta_b$ é Falso, então: localize a posição do novo pelo RefMoveI

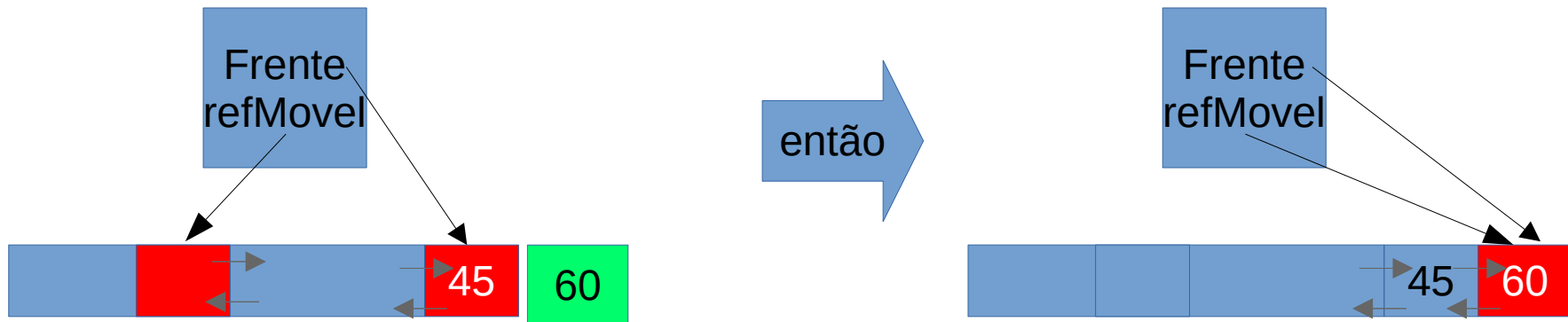


SIMULAÇÃO-2:

Inserindo 60

Caso #2: $\text{idade}(\text{idade}(\text{frente}) < \text{idade}(\text{novo}))$

Então o novo elemento será inserido como novo item de frente

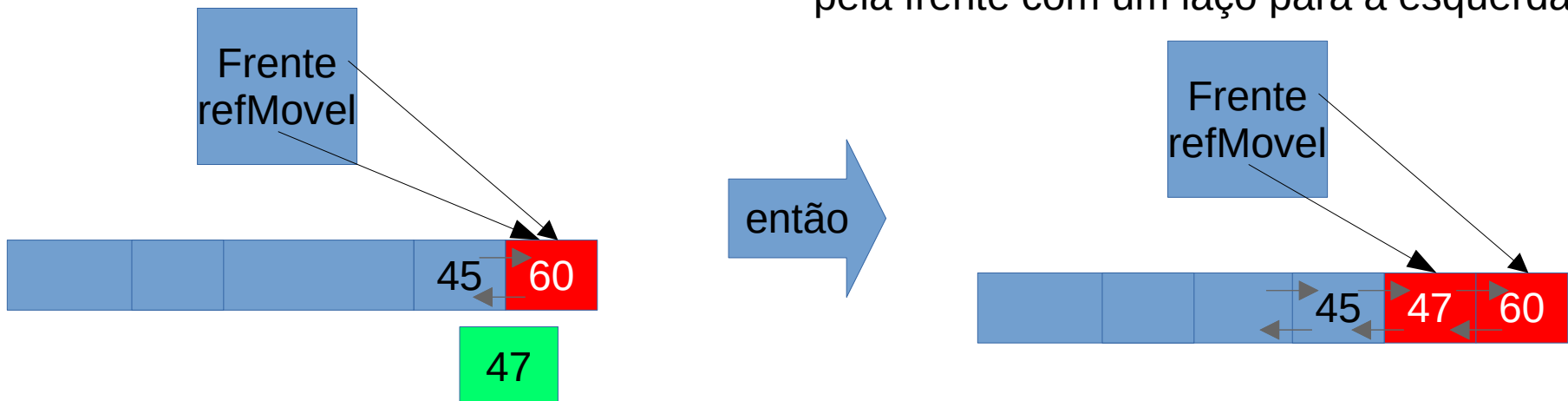


Inserindo 47

Caso#4: $\text{idade}(\text{refMov}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{frente})$

$$\Delta_c = |60-47| = 13 \quad \Delta_d = |60-47| = 13$$

$\Delta_c \leq \Delta_d$ é verdadeiro, (de fato $\Delta_a == \Delta_b$), então: localize a posição do novo nó pela frente com um laço para a esquerda!

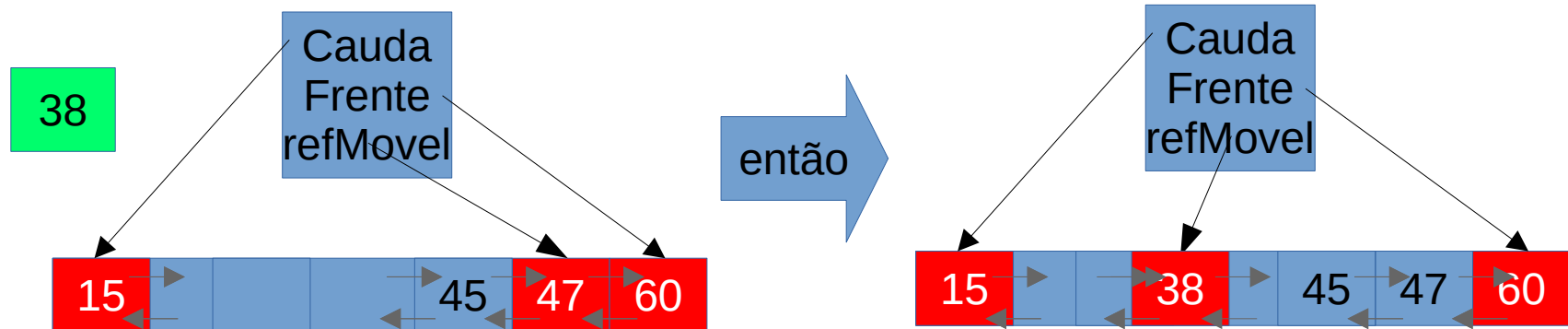


SIMULAÇÃO-3A:

Inserindo 38

Caso #3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMoveI})$

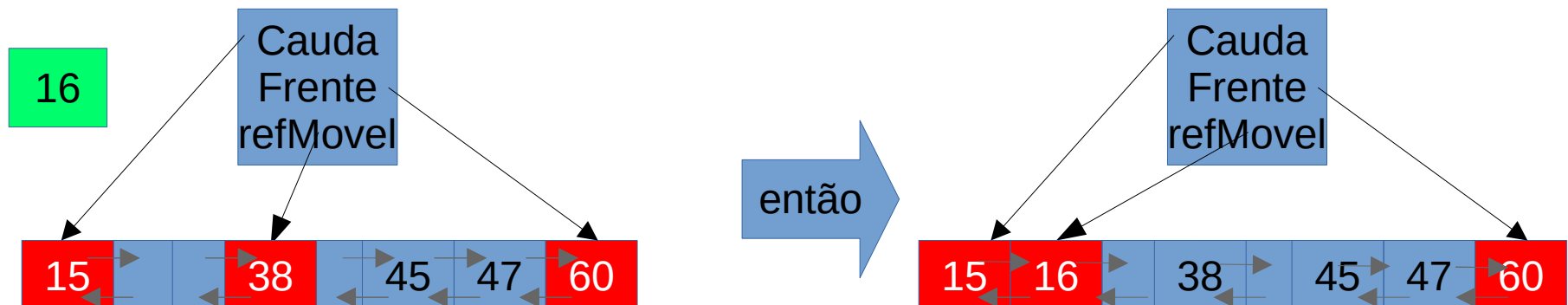
$\Delta_a = |15-38| = 23$ $\Delta_b = |47-38| = 9$, $\Delta_a \leq \Delta_b$ é Falso, então: localize a posição do novo nó pelo refMoveI com um laço para a esquerda!



Inserindo 16

Caso #3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMoveI})$

$\Delta_a = |15-16| = 1$ $\Delta_b = |38-16| = 22$, $\Delta_a \leq \Delta_b$ é verdadeiro, então: localize a posição do novo nó pela cauda com um laço para a direita!

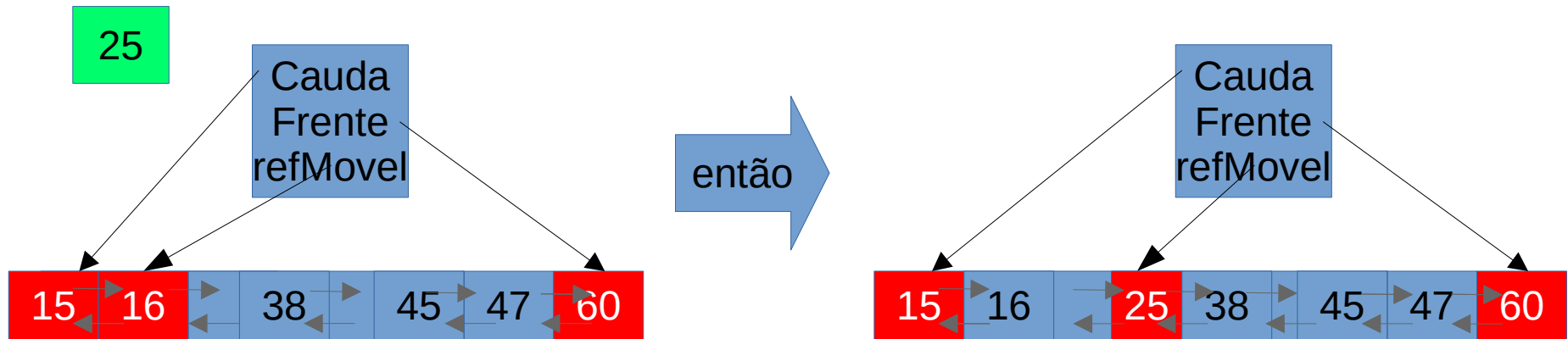


SIMULAÇÃO-3B:

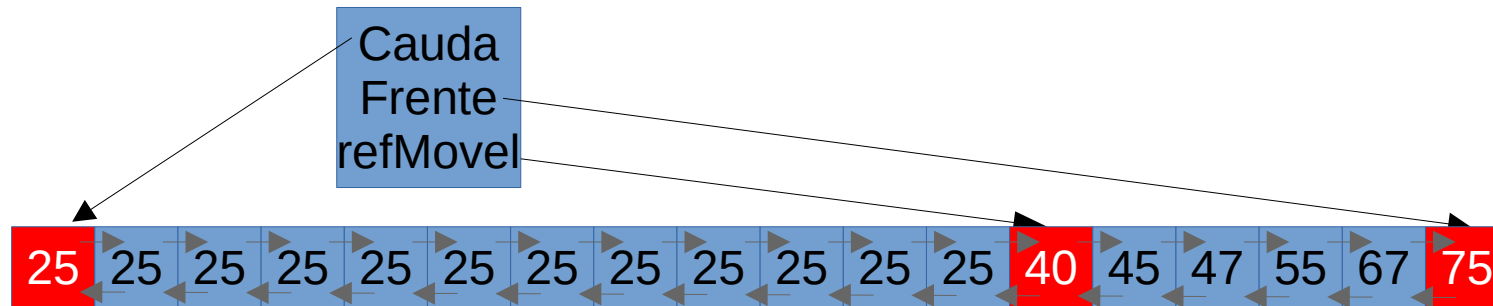
Inserindo 25

Caso #3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMoveI})$

$\Delta_a = |60-25| = 35$ $\Delta_b = |16-25| = 9$, $\Delta_c \leq \Delta_d$ é Falso, então: localize a posição do novo nó pelo refMoveI com um laço para a direita!



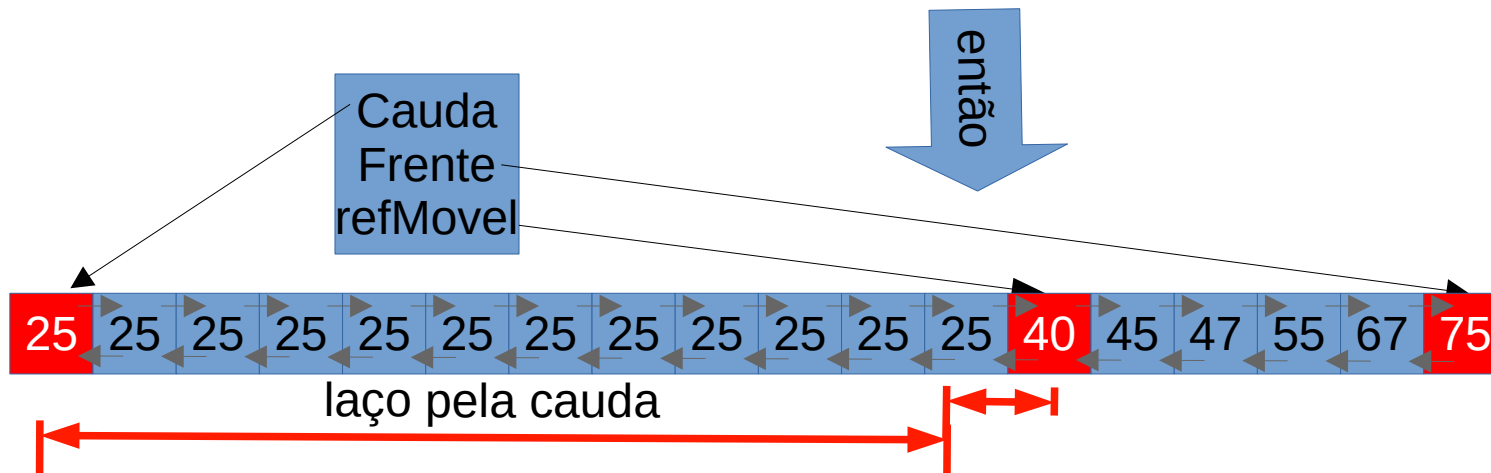
Quando o processo falha:



Inserindo 26

Caso #3: $\text{idade}(\text{cauda}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{refMove!})$

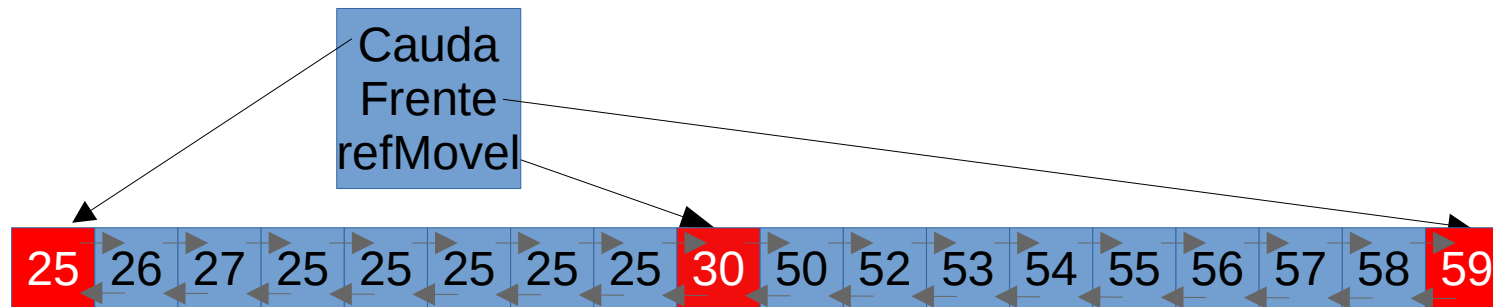
$\Delta_a = |25-26| = 2$ $\Delta_d = |40-26| = 14$, $\Delta_a \leq \Delta_b$ é verdadeiro, então: localize a posição do novo nó pela cauda com um laço para a esquerda!



Note que a escolha do laço pela cauda (ao invés do refMove!) levará erroneamente a mais iterações!!

Caso similar ocorre em repetições de idades pela frente.

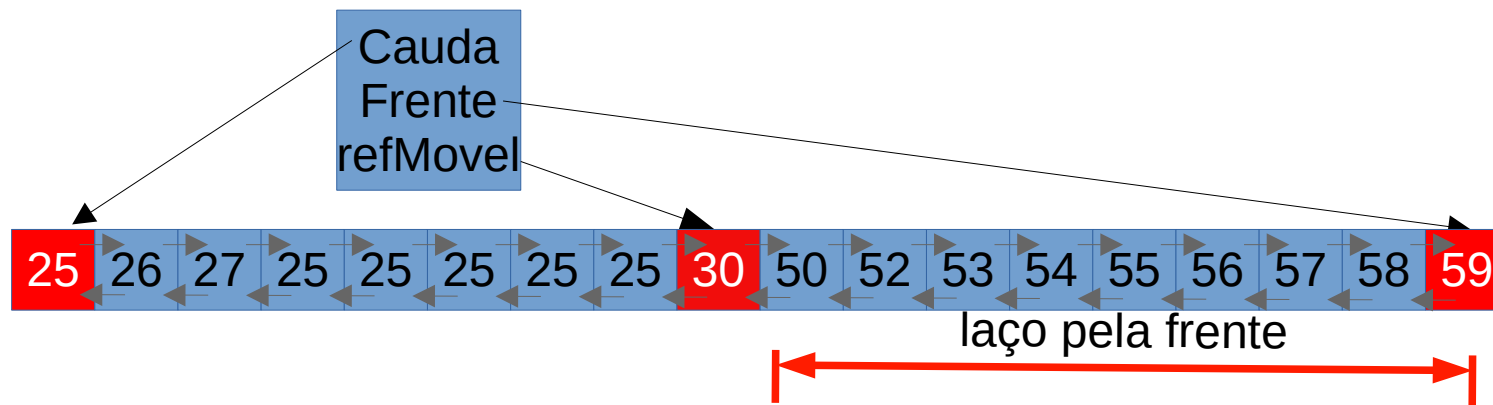
Quando o processo falha:



Inserindo 49

Caso #4: $\text{idade}(\text{refMoveI}) \leq \text{idade}(\text{novo}) \leq \text{idade}(\text{frente})$

$\Delta_a = |30-49| = 19$ $\Delta_d = |49-59| = 10$, $\Delta_b \leq \Delta_a$ é verdadeiro, então: localize a posição do novo nó com um laço erroneamente pela frente para a esquerda!



Note que a escolha do laço pela frente (ao invés do refMoveI) levará erroneamente a mais iterações!!

Estas e outras singularidades podem comprometer as distancias calculadas e não fornecem laço mais curto no caminho até a posição de inserção. Nesses casos, o desempenho desta implementação equivale à implementação convencional sem refMoveI.

Exercício:

Utilize o código da FDE de prioridade discutido em sala e, junto com o seu colega de equipe, busque projetar as operações básicas sobre a FDE de prioridade com referencial móvel.

Considere que o referencial móvel (*refMoveI*):

- É igual a Null para a fila vazia ou...

- Apona para o endereço do elemento mais recentemente inserido.

A próxima tarefa versará sobre uma FDE de prioridade com referencial móvel.

O modelo (FDE de prioridade) foi tratado em sala e é descrito no pdf publicado no Moodle.