

FILA

FILA

- ◉ Fila é uma estrutura de dados e um tipo abstrato de dados
- ◉ Seu comportamento se baseia no armazenamento de tipo First In First Out (FIFO), ou seja, estruturas onde o Primeiro elemento a Entrar é o Primeiro a Sair.

FILA

- ◉ Conceitualmente, a FILA possui duas operações básicas:
 - DEQUEUE: retira o primeiro elemento da Fila; e
 - QUEUE: insere um elemento no final da Fila.
- ◉ Para termos de implementação ainda temos funções de inicialização de filas, verificação se está vazia, acessa primeiro, ...

FILA

- ◉ Em termos de implementação, a Fila pode ser encadeada, estática, genérica ou não genérica.
- ◉ O mais importante é compreender que a Fila é uma lista com regras mais restritas de acesso e operações mais simples.

FILA

- ◉ A fila é muito empregada na Computação, geralmente para sistemas onde há compartilhamento “justo” de recursos:
 - sistemas operacionais: fila de processos;
 - simulação discreta de sistemas: geralmente em problemas onde há recursos restritos e deseja-se simular um comportamento do sistema simulado;
 - etc...

TAD FILA

- Mas como especificamos um Tipo Abstrato de Dado FILA?

TAD FILA

Elemento 0

Elemento 1

Elemento 2

Elemento 3

Como era de se imaginar, em uma fila os elementos estão dispostos em sequência

TAD FILA

Elemento 0

Elemento 1

Elemento 2

Elemento 3

As operações de leitura e remoção devem ser realizadas somente no início, por isso, não se precisa utilizar estruturas de dados muito complexas para implementá-la.

Uma lista encadeada simples já resolve muito bem a implementação de uma Fila.

Ainda é indicado trabalhar com estruturas genéricas, para evitar reimplementações (retrabalhos)

TAD FILA

Elemento 0

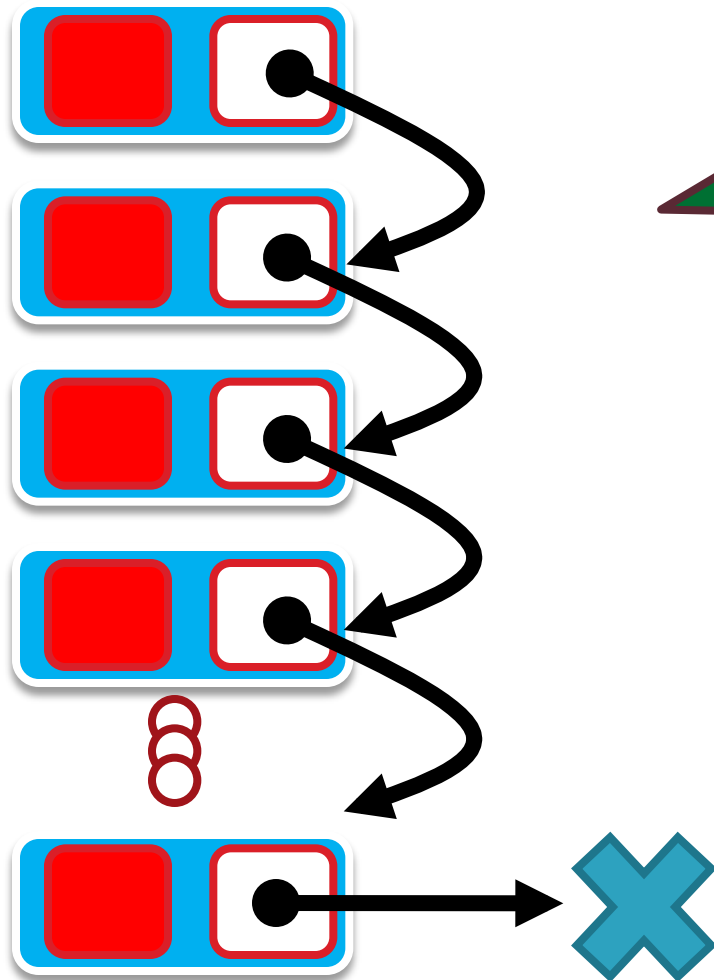
Elemento 1

Elemento 2

Elemento 3

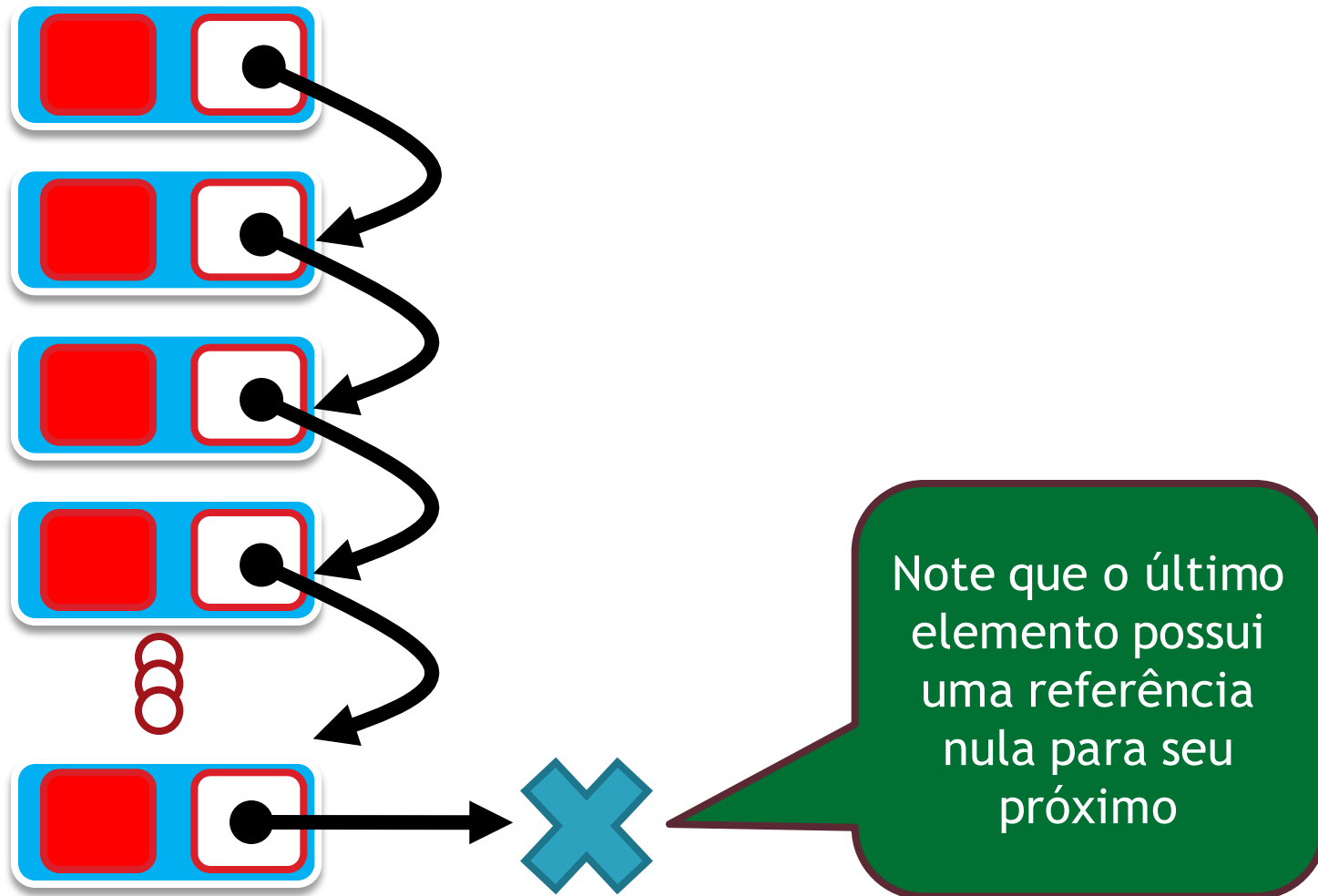
Portanto, iremos trabalhar com uma Fila implementada através da estrutura que classificamos como Lista Encadeada Dinâmica Genérica.

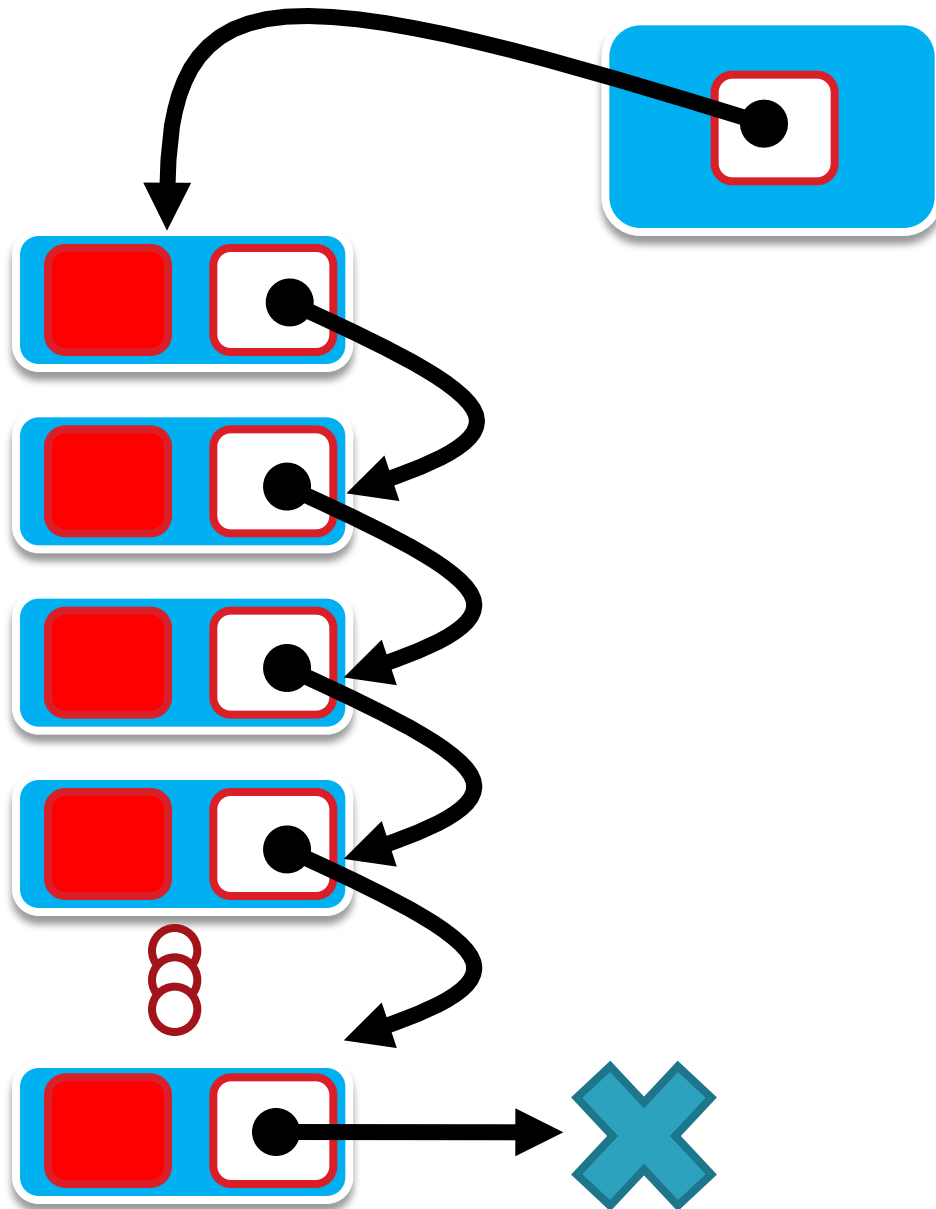
TAD FILA



Toda estrutura de encadeamento simples deve acomodar os dados utilizando referências para os próximos elementos.

TAD FILA





A Fila deve ser representada por uma estrutura que contenha o endereço do Primeiro elemento, pois a partir do mesmo, consegue-se chegar a qualquer elemento

Vamos ver como
ficam cada uma das
estruturas

TFILA

TFile

Referência
do primeiro
elemento

A estrutura que
representa a
Filadeve ter a
referência do topo

TFILA

TFila

TElementoF
* primeiro;

A referência é um
ponteiro para o
primeiro elemento
da Fila

TFILA

TFila

TElementoF
* primeiro;



```
template<typename TIPO>
struct TFila{
    TElementoF<TIPO> * primeiro;
};
```


TFILA

Cada elemento de Fila, como uma lista encadeada simples, possui um registro para dado e a referência para o próximo elemento.

TElementoF

DADO

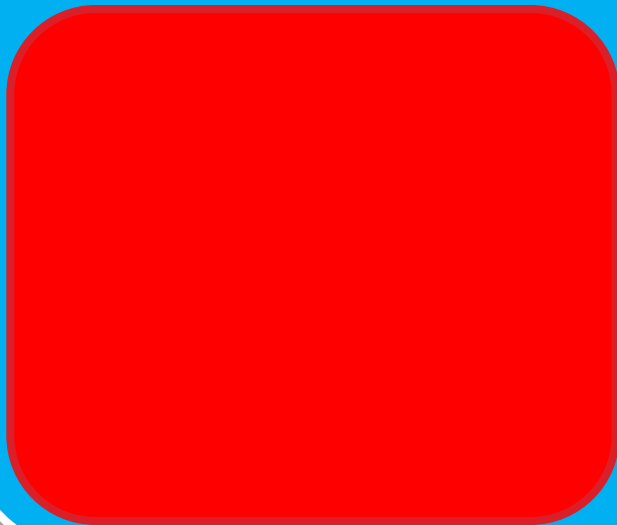
**REFERÊNCIA
PARA
PRÓXIMO**

TFILA

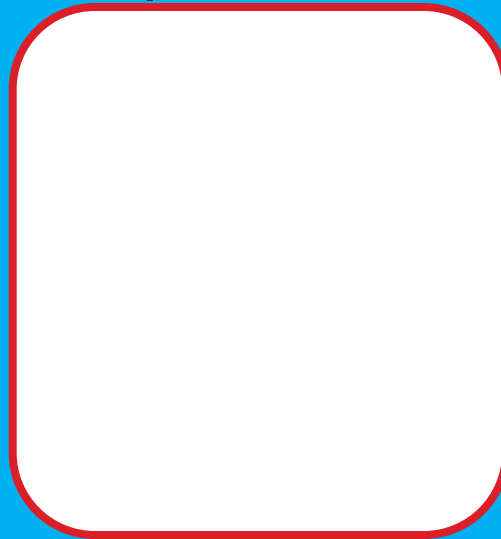
O dado é do tipo genérico.
A referência do próximo
elemento é um ponteiro para
TElementoF.

TElementoF

TIPO dado;



TElementoF
* proximo;



TFILA

TElementoF



```
template<typename TIPO>
struct TElementoF{
    TIPO dado;
    TElementoF<TIPO> *proximo;
};
```

Como funcionam as
principais operações?

QUEUE - ENFILEIRAR

- ◉ A operação QUEUE adiciona um item à Fila.
- ◉ No momento da inserção deve-se:
 1. criar um novo elemento;
 2. acomodar o dado no novo elemento;
 3. deve-se procurar o final da Fila
 4. fazer o final da Fila ter como próximo a referência do novo elemento; e
 5. o registro próximo do novo elemento deve possuir o valor NULL.
- ◉ QUEUE é equivalente à operação “insere no fim”

QUEUE (ENFILEIRAR)

TFila



A fila está vazia!
Logo, a referência
para seu topo é
NULL

QUEUE (ENFILEIRAR)

TFila



(0x09)

Primeiro cria-se o
elemento

QUEUE (ENFILEIRAR)

TFila



(0x09)

Depois acomoda-se o
dato

QUEUE (ENFILEIRAR)

TFila



(0x09)

Como não há mais
elementos na fila, o
novo torna-se o
primeiro

QUEUE (ENFILEIRAR)

TFila

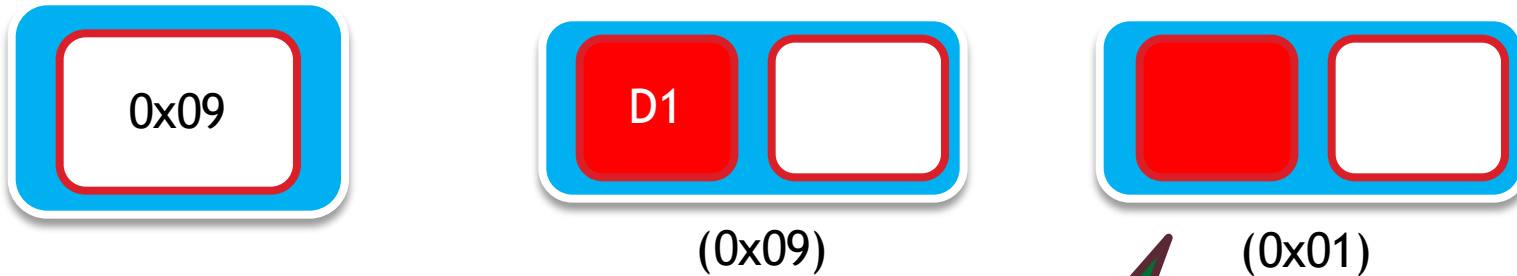


(0x09)

A referência do
próximo no novo
recebe NULL

QUEUE (ENFILEIRAR)

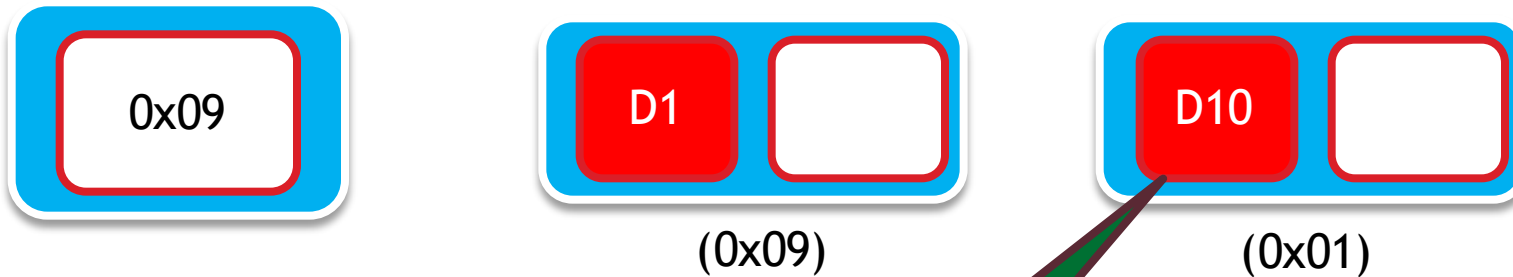
TFila



Para acomodar um novo dado na Fila, primeiro cria-se o elemento

QUEUE (ENFILEIRAR)

TFila



Depois acomoda-se o
dato

QUEUE (ENFILEIRAR)

TFila



(0x09)



(0x01)

Procura-se o último elemento da fila e atualiza-se o registro de próximo para o endereço do novo elemento

DEQUEUE (DESENFILAR)

- ◉ A operação DEQUEUE retira um item da fila.
- ◉ Note que esta operação, deve retornar o dado do elemento recém retirado.
- ◉ Para retirar deve-se:
 1. realizar uma cópia do dado do primeiro elemento;
 2. gravar a referência do primeiro elemento da fila;
 3. gravar como primeiro elemento da fila a referência do endereço armazenado no registro próximo do primeiro elemento atual;
 4. liberar memória alocada pelo endereço do antigo primeiro elemento; e
 5. retornar dado do elemento recém removido.
- ◉ DEQUEUE é equivalente à operação “remove início”

DEQUEUE (DESENFILAR)

TFila



(0x09)



(0x01)

Vamos realizar a
operação de
DEQUEUE

DEQUEUE (DESENFILAR)

TFila



(0x09)



(0x01)



Inicialmente deve se
realizar uma cópia
do dado do primeiro
elemento

DEQUEUE (DESENFILAR)

TFila



(0x09)



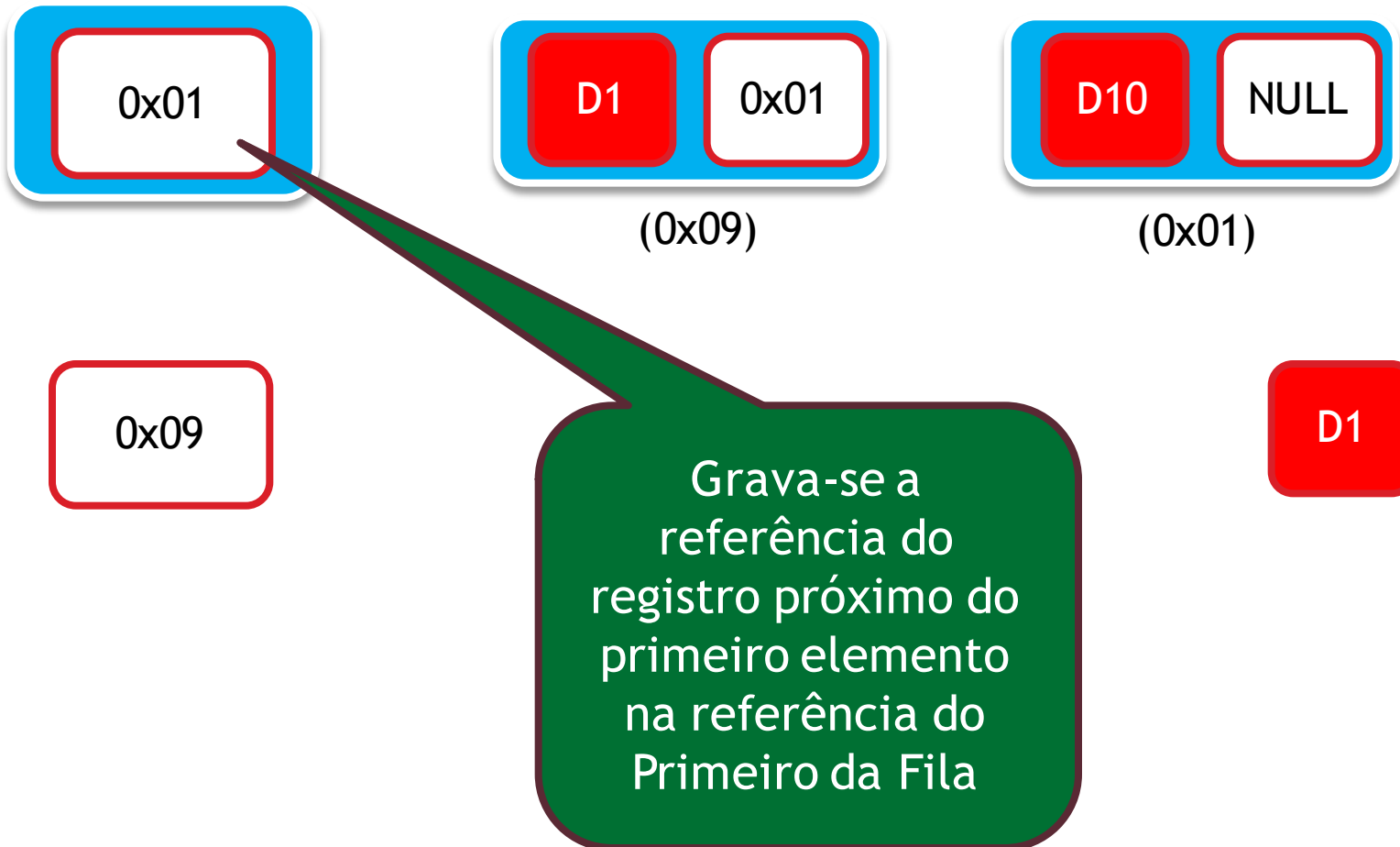
(0x01)



Depois, deve-se
gravar a referência
de memória do
primeiro elemento
atual

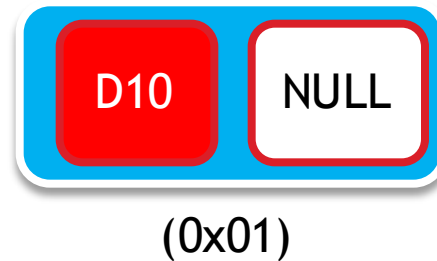
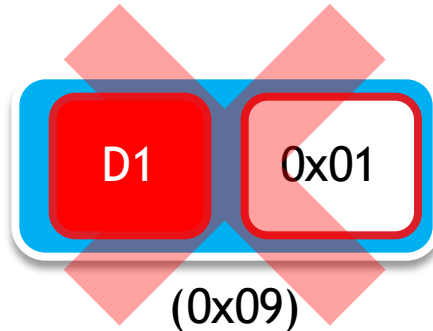
DEQUEUE (DESENFILAR)

TFila



DEQUEUE (DESENFILAR)

TFila



Agora solicita-se
liberação da
memória do
endereço do antigo
primeiro elemento

DEQUEUE (DESENFILAR)

TFila



(0x01)



Ao final, o dado
salvo deverá ser
retornado!

SIMULAÇÃO UTILIZANDO FILAS

EXERCÍCIOS

- ◉ Realizaremos alguns exercícios de simulação utilizando Filas, pois são exemplos de aplicação muito interessantes para o nosso tema.
- ◉ Que tal prepararmos o terrenos e entendermos um pouco mais sobre o assunto?