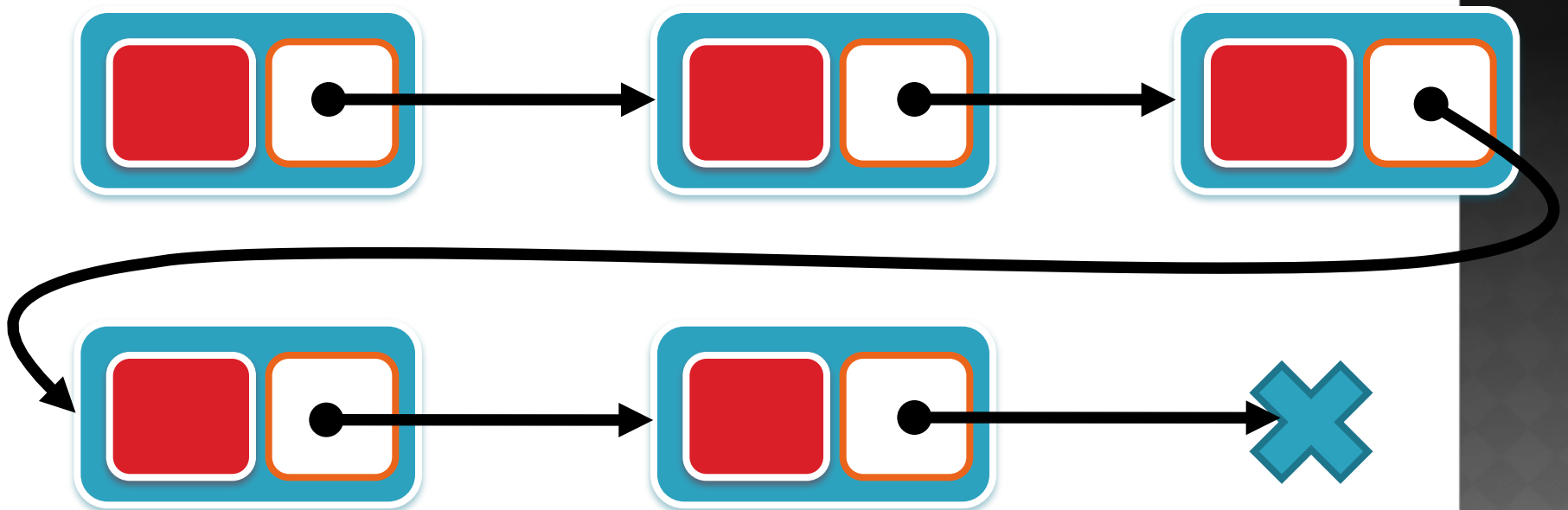


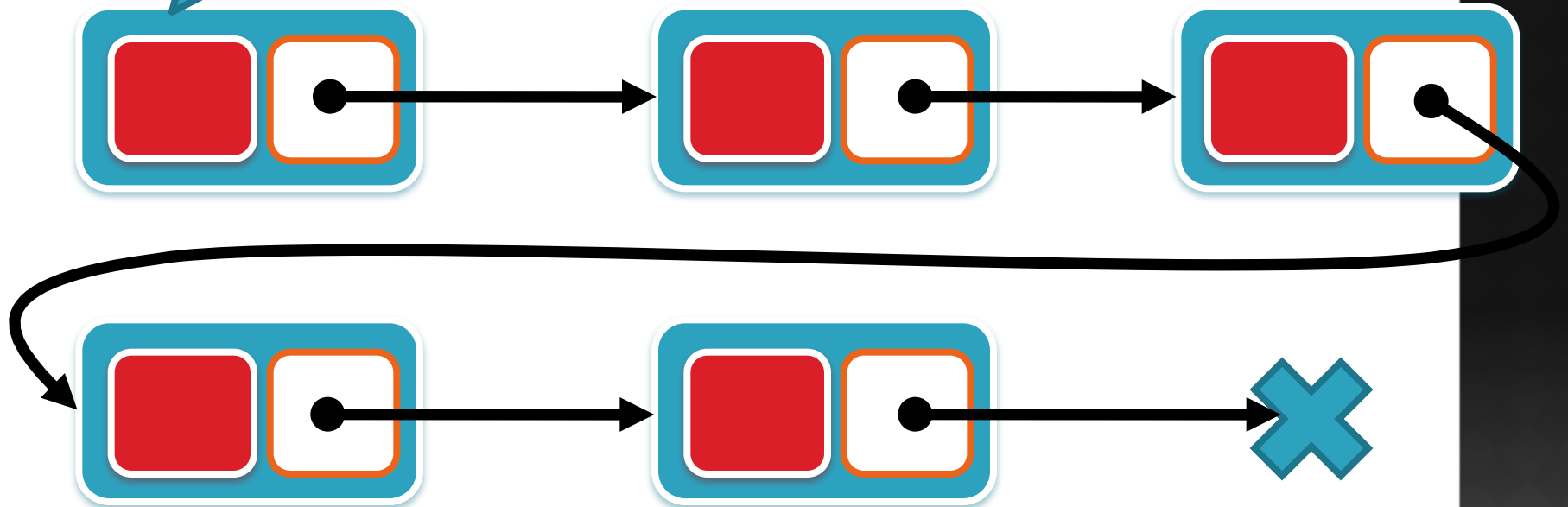
LISTA ENCADEADA

LISTA ENCADEADA

- Uma lista encadeada ou uma lista ligada é uma lista onde cada elemento possui a referência do próximo nó.

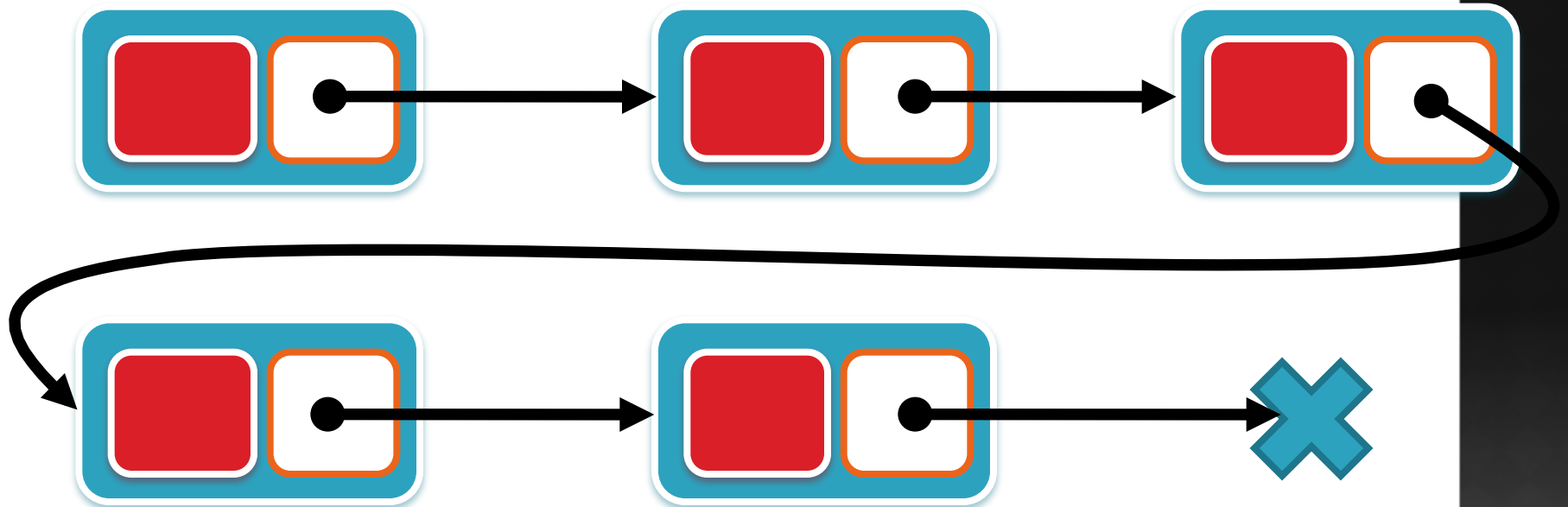


Cada elemento possui dois campos:
vermelho e branco.
Neste exemplo, o vermelho
simboliza o dado armazenado e o
branco representa a referência para
o próximo elemento.

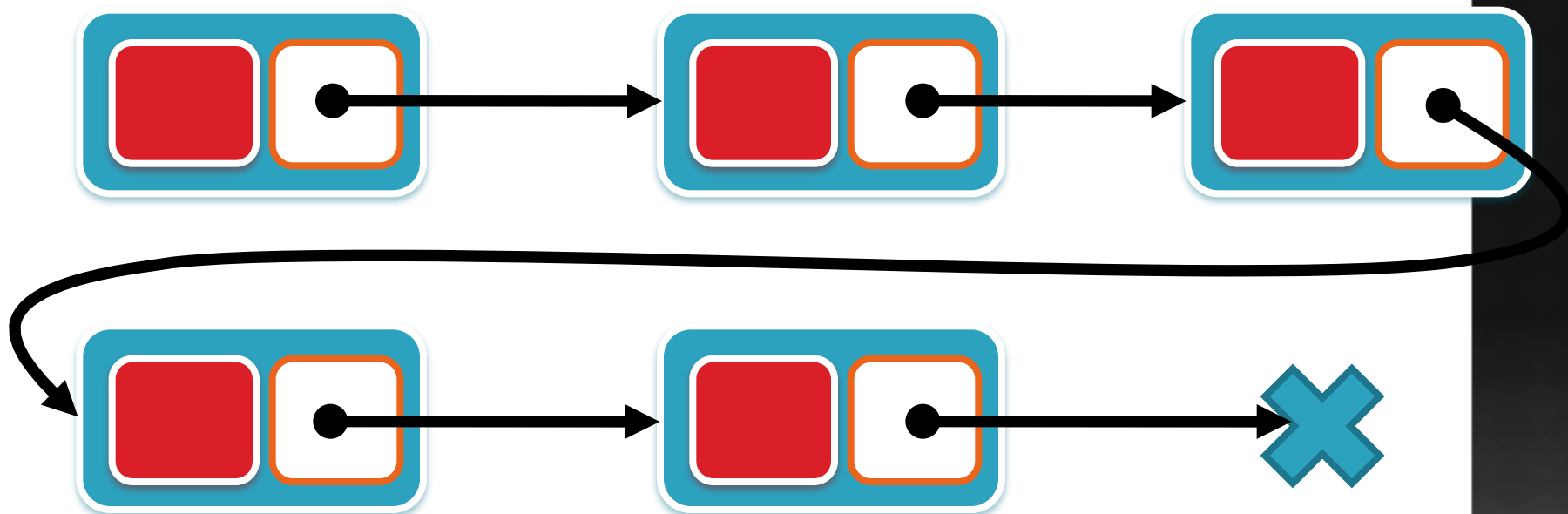


LISTA EN

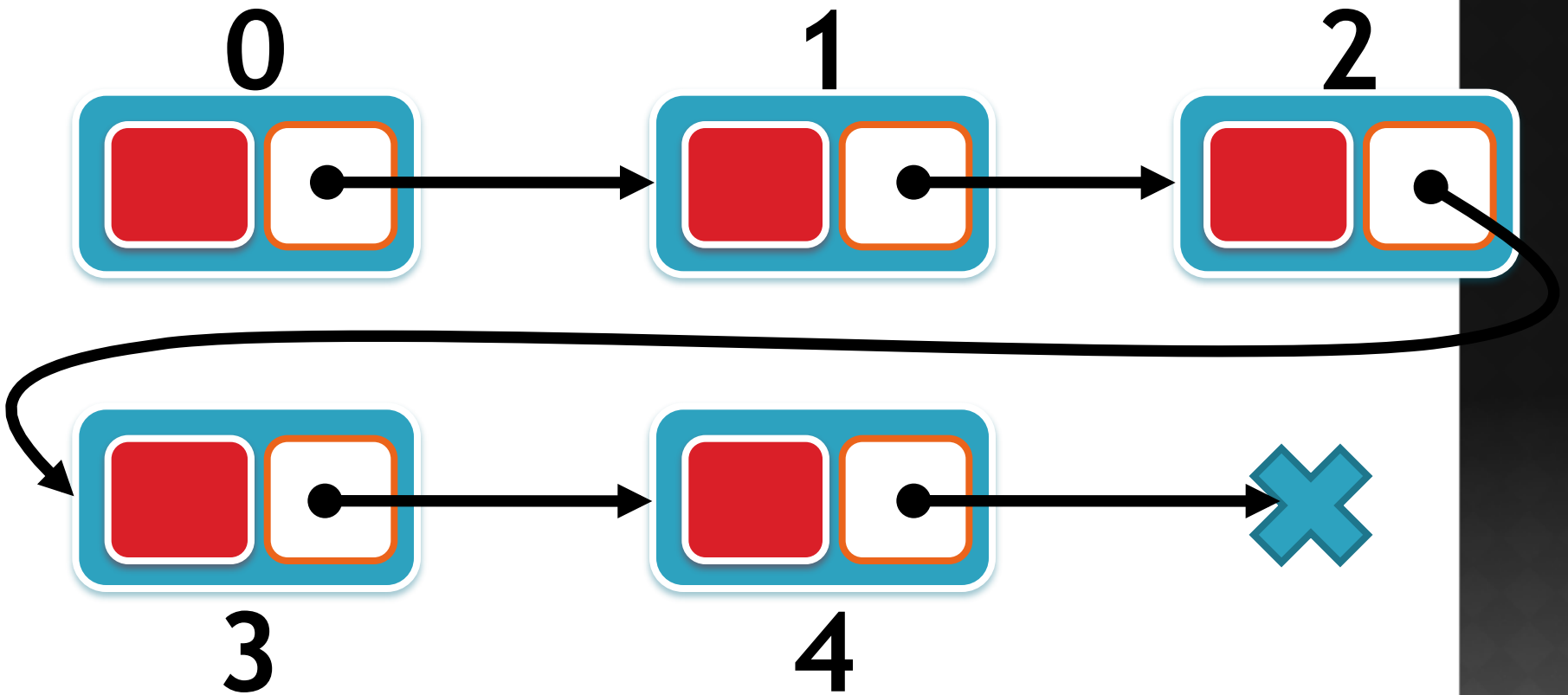
Logo, para no contexto de nossa disciplina é importantíssimo saber o porquê e como utilizar estas estruturas de controle.



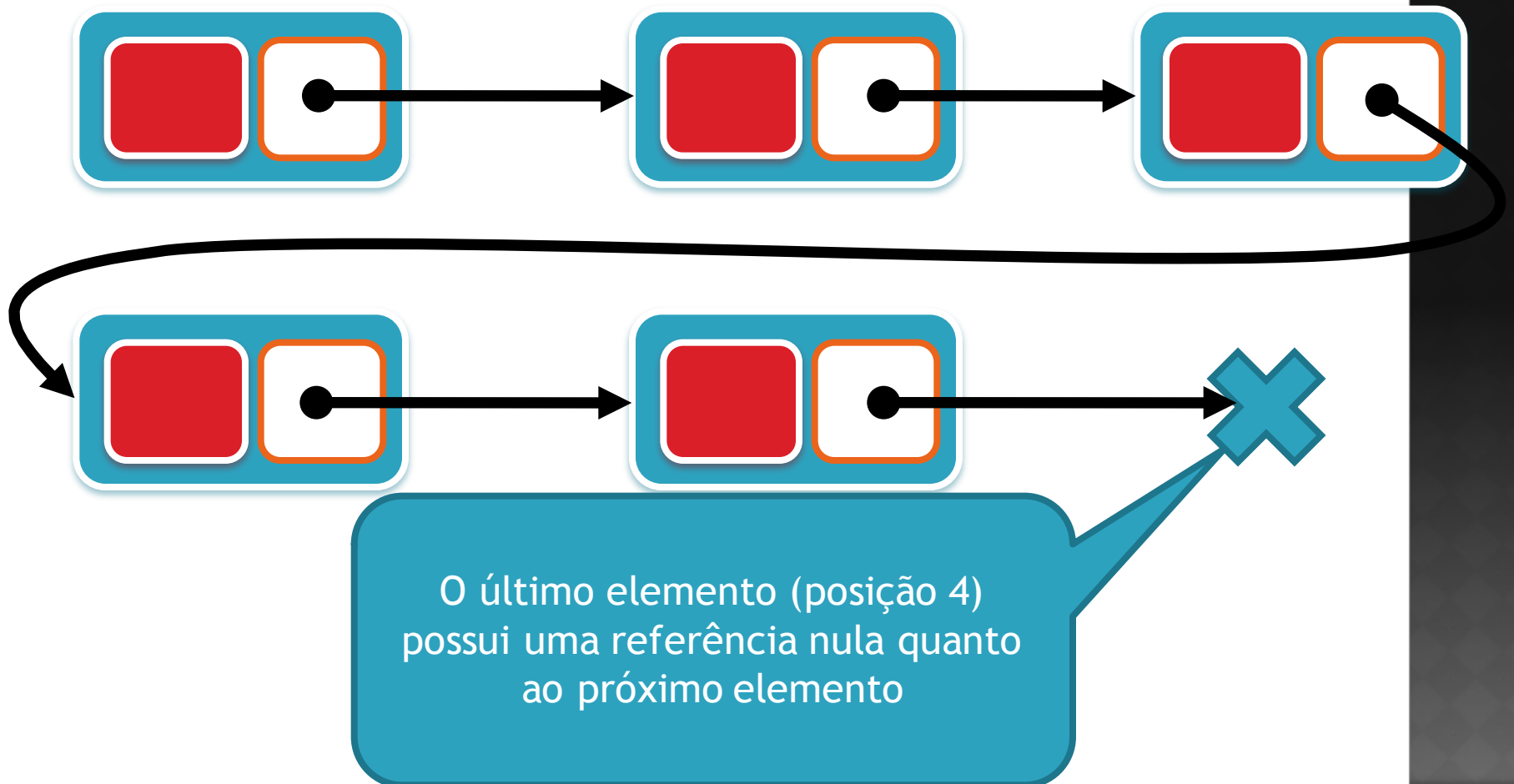
Note que, a partir do primeiro elemento, consegue-se localizar todos os outros, devido ao encadeamento realizado através da referência do próximo.



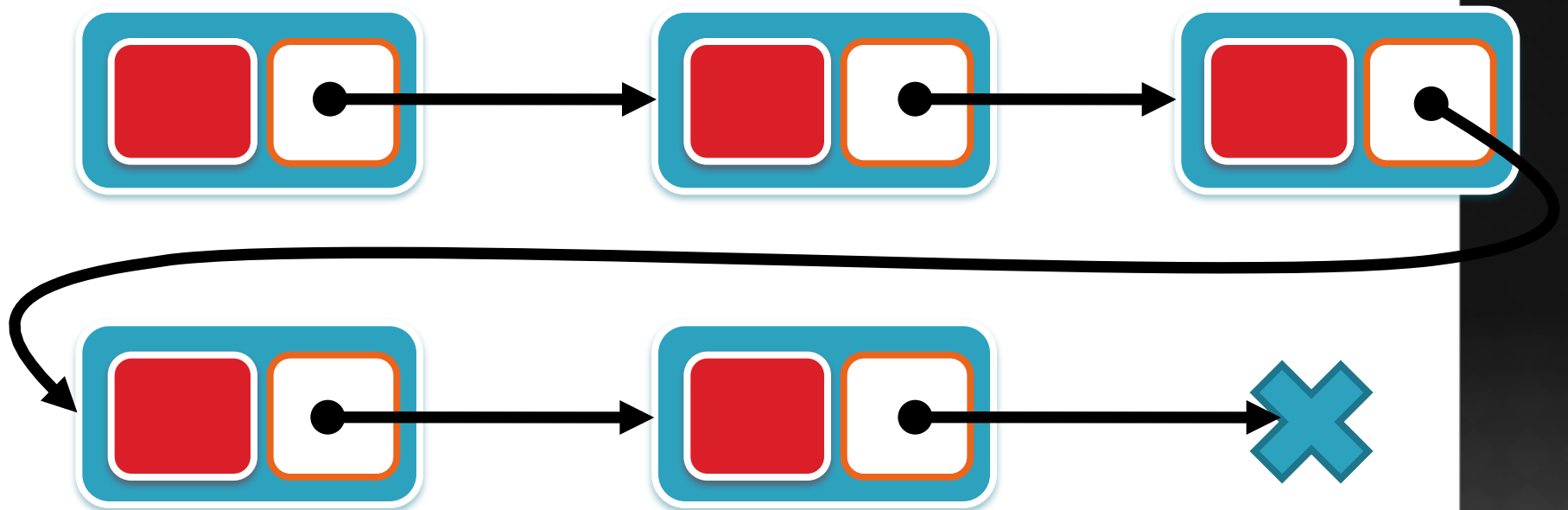
Há 5 elementos nesta lista.
Se os denotarmos em índices, podemos dizer que o primeiro é o elemento 0 e o último é o de índice 4.
Esta referência é indica a posição na lista
(não tem relação com o índice de vetor).



LISTA ENCADEADA



Perguntas:
Quais vantagens?
Quais desvantagens?



Respostas:

Quais vantagens?

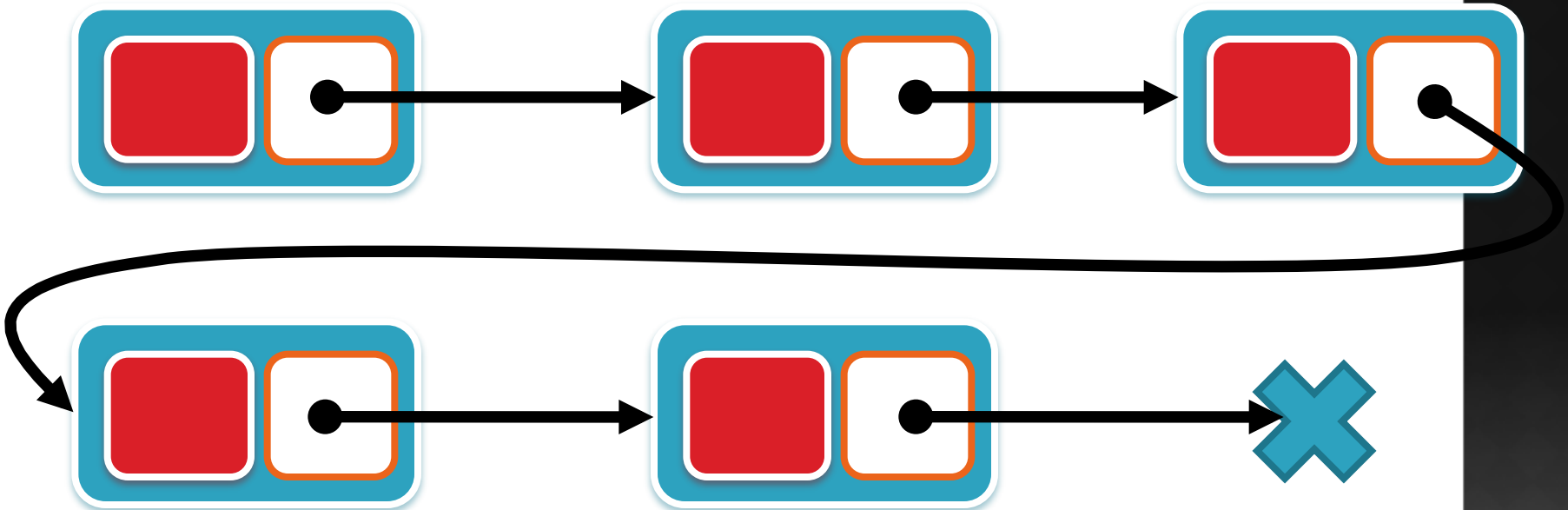
- a posição dos elementos não precisam estar contíguos na memória
- inserções e remoções podem ocorrer de forma mais facilitada, sem a necessidade de deslocamento dos elementos (apenas trocas de referências)
- pode-se utilizar de estruturas sem capacidade máxima definida na compilação

Quais desvantagens?

- para encontrar um elemento de um determinado índice, deve-se passar por todos os seus elementos antecessores
- implementação é mais difícil
- a “desalocação”, na maioria das linguagens, não é automática (deve-se ter muito cuidado)

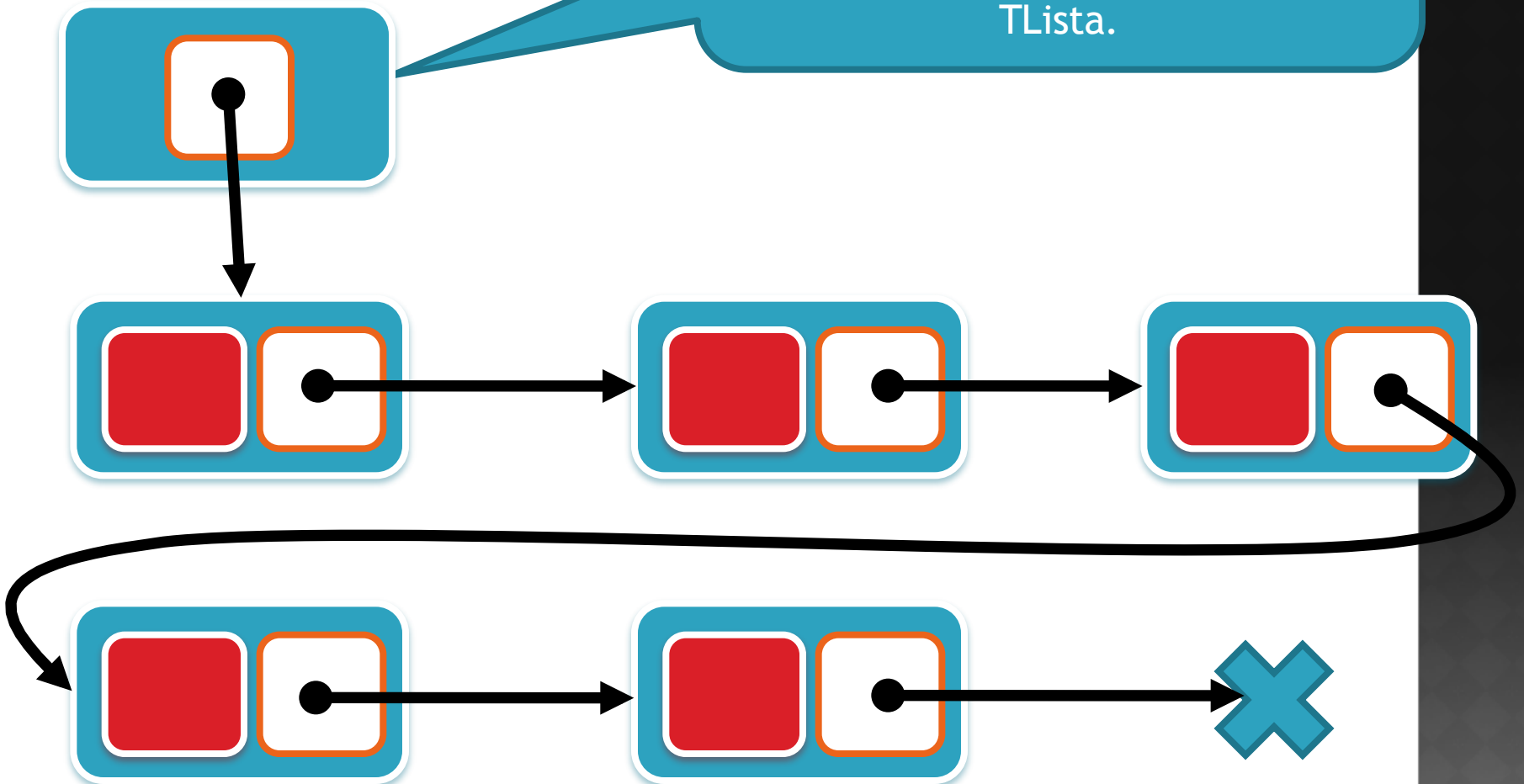
LISTA ENCA

IMPORTANTE: para se manter a integridade da lista, não pode se perder a referência do primeiro elemento!



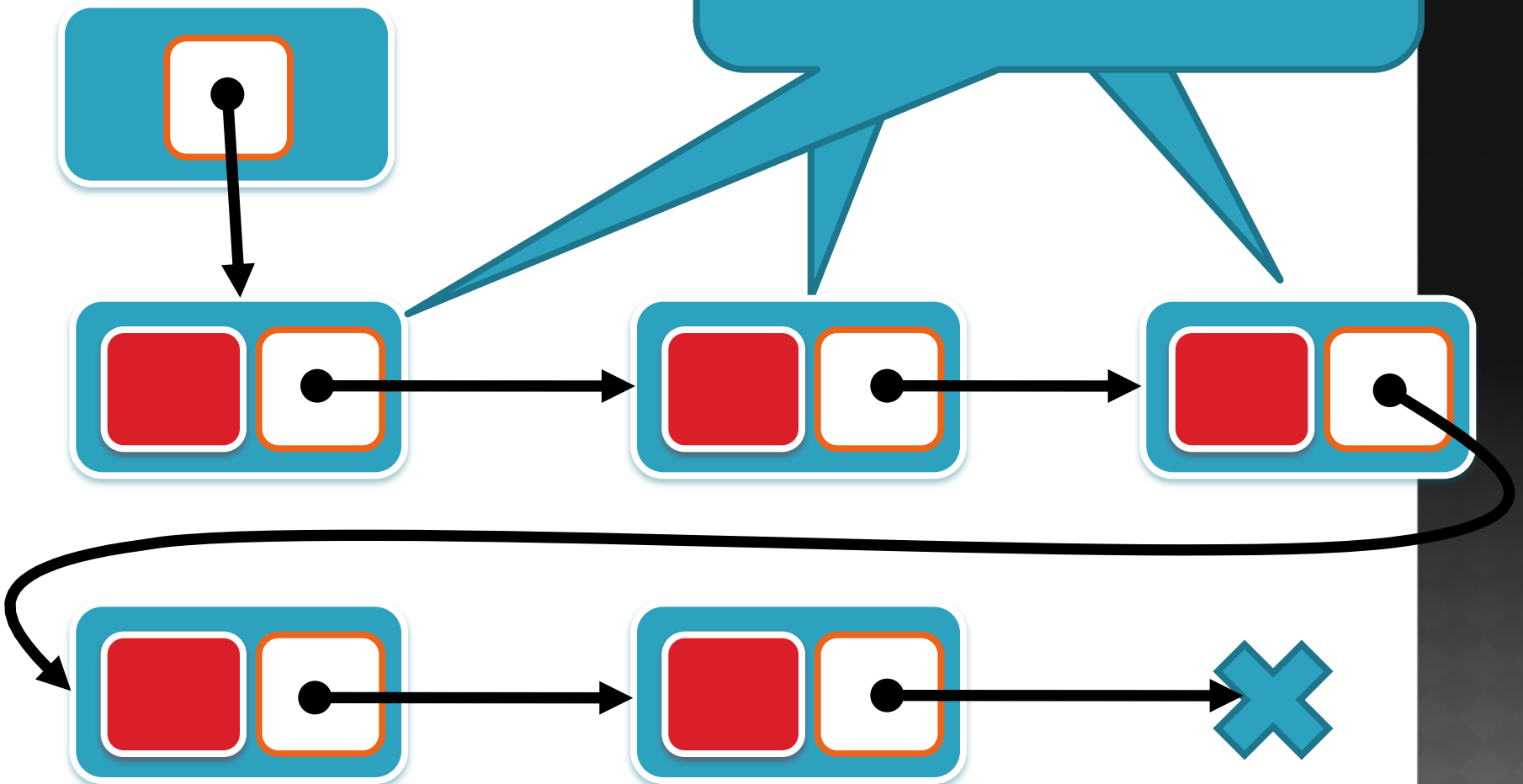
LISTA ENCADEADA

Devido a isto, devemos ter um registro da referência do primeiro elemento, em outra estrutura. Esta estrutura chamaremos de TLista.



LISTA ENCADEADA

A estrutura dos elementos, chamaremos de TEemento.



Vamos evoluir
conceitualmente cada
uma das estruturas para
depois pensarmos na
codificação da TAD (C++)

TELEMENTO

Temos dois campos: um para o dado (vermelho) e outro para a referência do próximo elemento (branco)

DADO

REFERÊNCIA

TLISTA

O TLista terá apenas um registro apenas, que conterá a referência do primeiro elemento da nossa lista encadeada

REFERÊNCIA

CODIFICAÇÃO DO TAD

- ◉ Vendo as estruturas necessárias para nossa TAD lista encadeada, vamos fazer algumas pequenas reflexões antes da implementação em C++:

CODIFICAÇÃO DO TAD

- ◉ Uma lista encadeada pode ser implementada através de vetores ou matrizes. Deste modo, teríamos uma estrutura contígua, armazenando os elementos da lista em posição aleatória.



Neste caso, a referência do próximo elemento seria o índice do vetor e estaríamos limitados a uma capacidade máxima de armazenamento.

CODIFICAÇÃO DO TAD

- ◉ Para nossas aulas, iremos trabalhar com estruturas encadeadas relacionadas a alocação dinâmica devido aos motivos:
 - trabalharmos o mais cedo possível com este conceito (muito necessário) na disciplina;
 - estruturas encadeadas tornam-se mais interessantes se trabalhadas de modo a não ter um limite pré-estabelecido.

CODIFICAÇÃO DO TAD

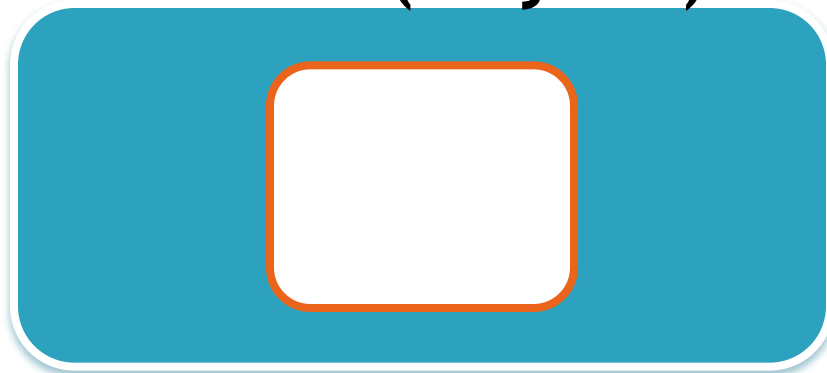
- ◉ O exemplo que será utilizado a seguir contextualizará uma lista encadeada para armazenamento de músicas
- ◉ Logo estaremos criando nossa *playlist* de músicas
- ◉ **QUAIS SERÃO OS DADOS?**
- ◉ **COMO SERÁ REALIZADA A REFERÊNCIA?**

(próximo...)

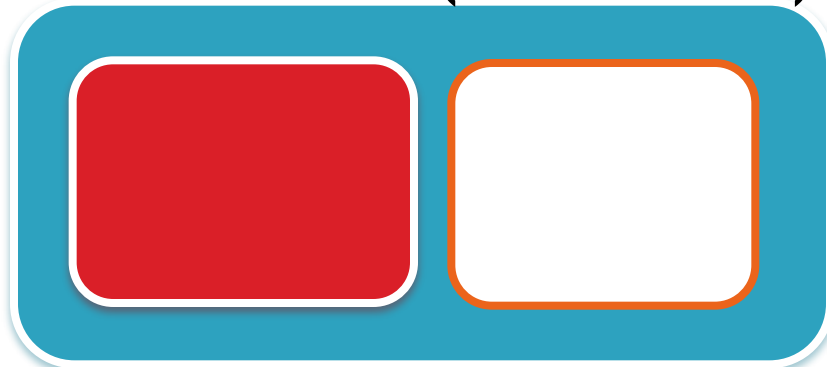
CODIFICAÇÃO DO TAD

- ◉ Vamos definir:

TLista (PlayList)



TElemento (elemento)



CODIFICAÇÃO DO TAD

- ◉ Então teremos como dados, as músicas (chamaremos de TMusica) e como referência utilizaremos variáveis do tipo ponteiros (que armazenam endereços de memória)
- ◉ As variáveis de referência serão ponteiros para quais tipos de dados?

CODIFICAÇÃO DO TAD

- ◉ Não podem ser para TMusica, pois não teríamos o encadeamento necessário, conforme ilustrado abaixo:

Fear of the
dark
-
Iron Maiden

Flight of
Icarus
-
Iron Maiden

Bard's Song
-
Blind
Guardian

Somente com a estrutura de músicas não conseguiremos realizar o encadeamento necessário para formar a lista. Por isso, iremos utilizar como referência os ponteiros para TElemento

CODIFICAÇÃO DO TAD

- ⦿ Tendo definido os dados e o que se tomará como referência, vamos definir como ficarão as estruturas
- ⦿ Iniciaremos com a estrutura do elemento

CODIFICAÇÃO DO TELEMENTO

TElemento

musica



proximo

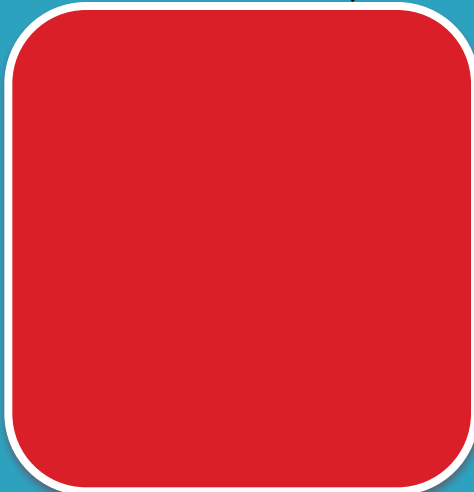


CODIFICAÇÃO DO T

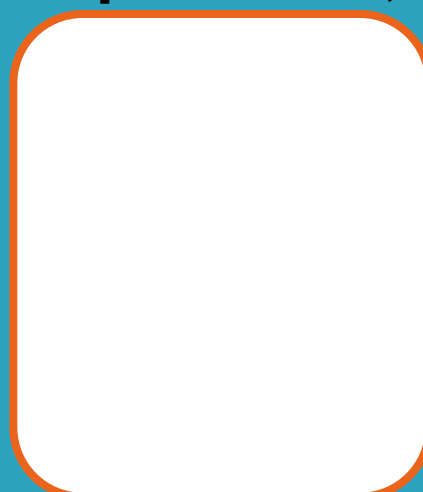
Os registros serão:
-musica: do tipo TMusica; e
-proximo: ponteiro para TElemento.

TElemento

TMusica
musica;



TElemento
*** proximo;**



CODIFICAÇÃO DO TELEMENTO

TElemento

TMusica
musica;

TElemento
* proximo;

```
struct TElemento{  
    TMusica musica;  
    TElemento * proximo;  
};
```



CODIFICAÇÃO DO TLISTA

TLista

inicio

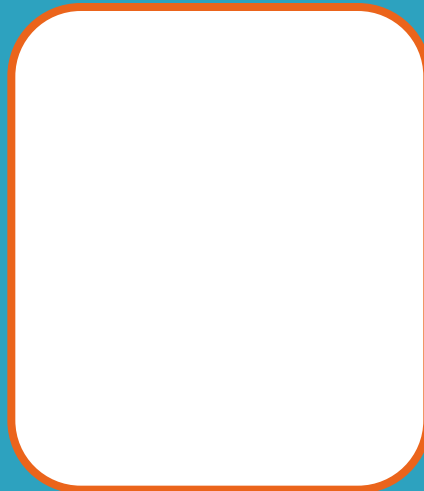
A diagram showing a blue rounded rectangle representing the TLista structure. Inside this rectangle, at the top, is the word 'inicio'. Below 'inicio' is a white rounded rectangle with an orange border, representing the first node of the list.

CODIFICAÇÃO DO T

O único registro é o que armazena o início da lista:
-início: ponteiro para o primeiro elemento

TLista

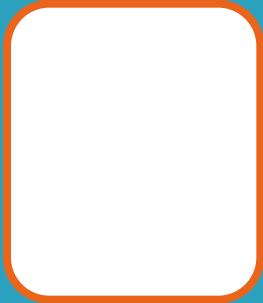
TElemento
*** início**



CODIFICAÇÃO DO TLISTA

TLista

TElemento
* inicio



```
struct TLista{  
    TElemento * inicio;  
};
```



FUNCIONAMENTO DA TAD

- ◉ Depois de discutirmos e definirmos as estruturas relacionadas a uma lista encadeada, a seguir veremos seu funcionamento através de uma sequência de ilustrações

FUNCIONAMENTO DA TAD

Esta é nossa lista encadeada. Ela está vazia.

Logo, não possui primeiro endereço de elemento para armazenar.

Devido a isto, armazena-se NULL no registro de início

Lista

NULL

FUNCIONAMENTO DA TAD

TMusica



TLista



A música X é criada e deseja
inserí-la na lista

FUNCIONAMENTO DA TAD

TElemento



0x08

TLista



Para isto, precisa-se utilizar um elemento para armazená-la. Logo criamos um TElemento, atribuímos a música ao mesmo

FUNCIONAMENTO DA TAD

TElemento



0x08

TLista

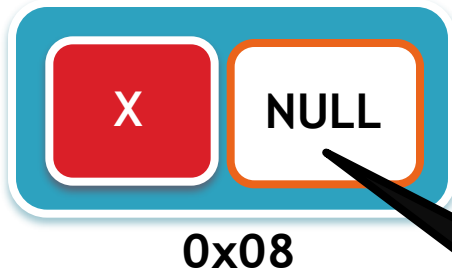


Como a lista está vazia, ao inserir um novo elemento, este torna-se o primeiro.

Consequentemente, o registro início de TLista armazena o endereço do elemento utilizado para a música X

FUNCIONAMENTO DA TAD

TElemento



TLista



Como este elemento é o último,
seu ponteiro para próximo deve
armazenar um endereço de
memória nulo

FUNCIONAMENTO DA TAD

TElemento



0x08

TMusica



Deseja-se agora inserir a música Q

TLista



FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TLista



Envolvemos a música Q através de um novo elemento alocado, cujo o endereço é 0x12

FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

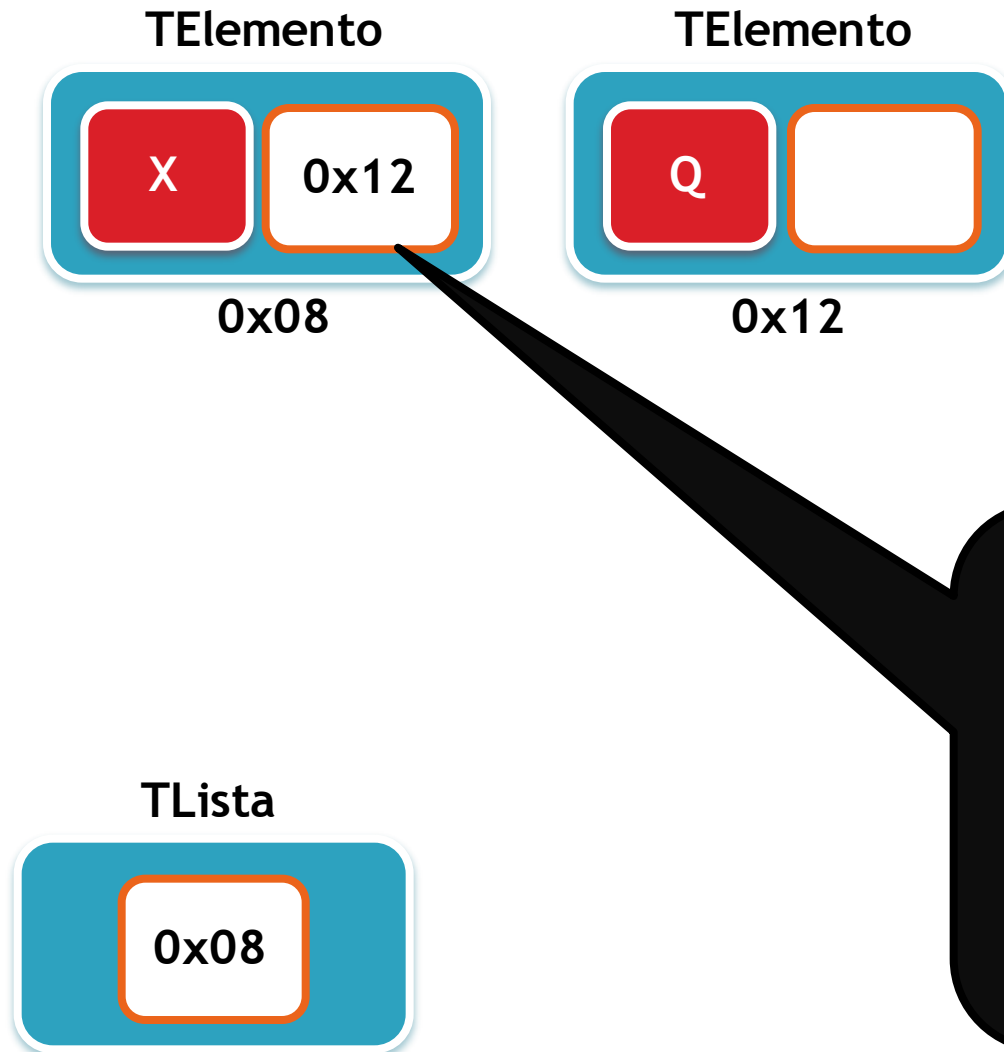
TLista



Agora temos que encadear o novo elemento à lista.

Como faremos?

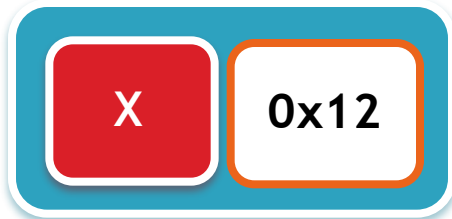
FUNCIONAMENTO DA TAD



Como inserimos no final da lista, o registro próximo do elemento 0x08, passará a armazenar o endereço do próximo elemento: 0x12

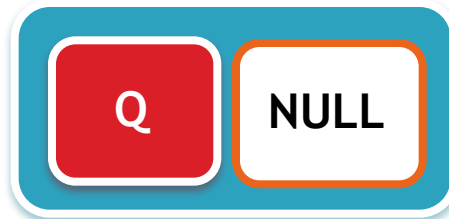
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

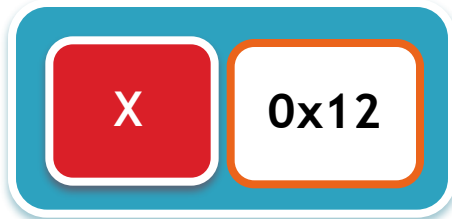
TLista



Como o novo elemento tornou-se o último, deve atualizar seu registro de próximo para NULL

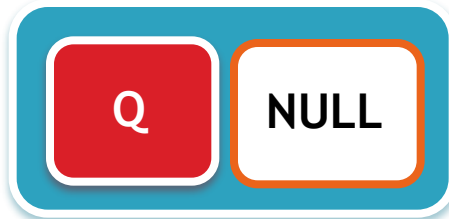
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TMusica



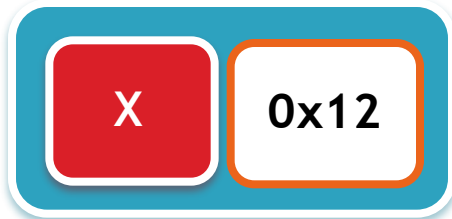
TLista



Deseja-se adicionar, ao final da playlist, a música S

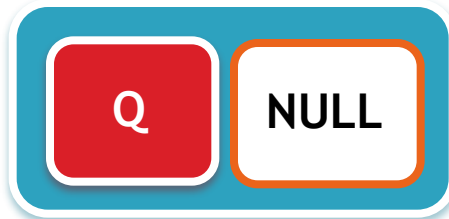
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TElemento



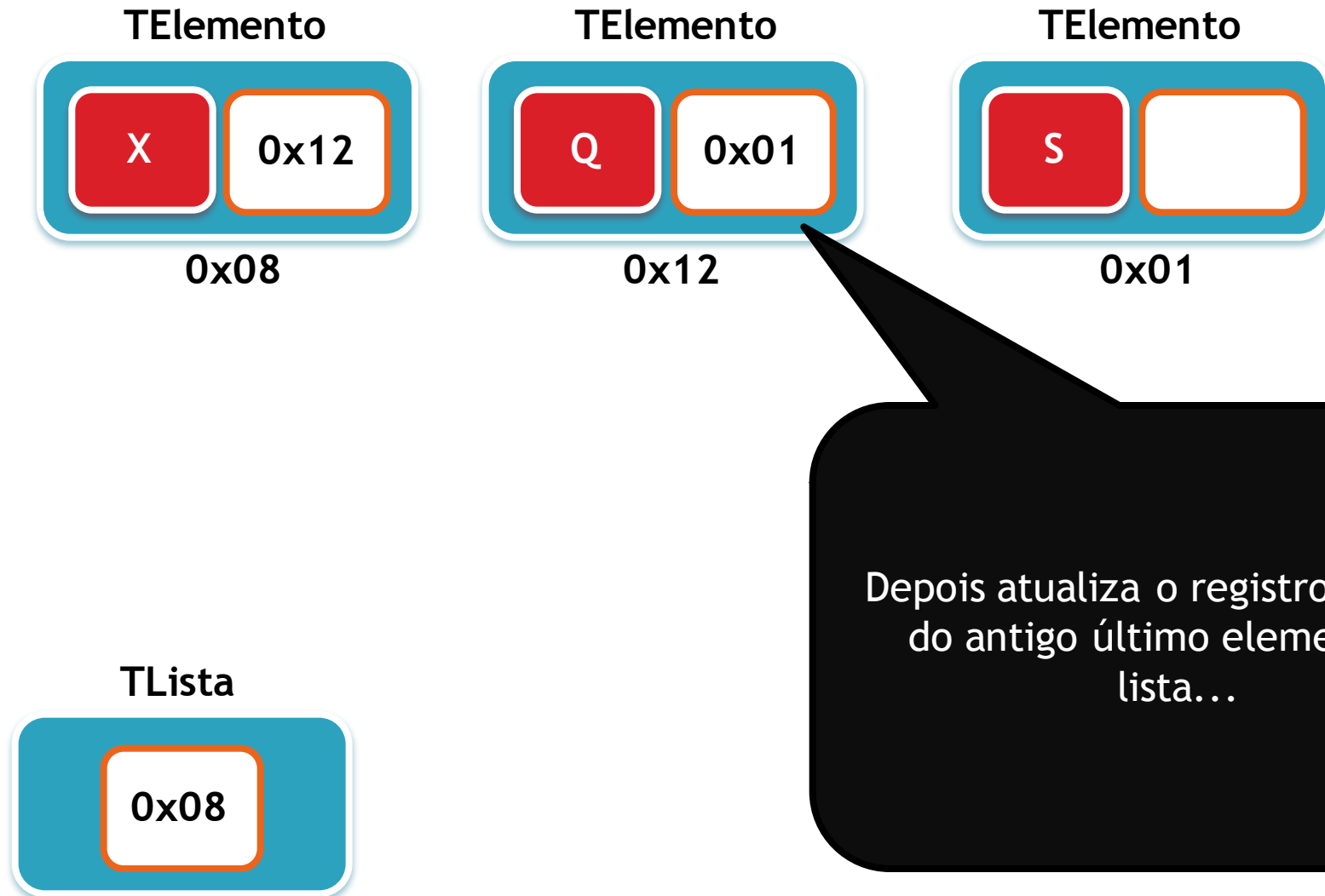
0x01

TLista



Envolve-se o dado em um novo elemento

FUNCIONAMENTO DA TAD

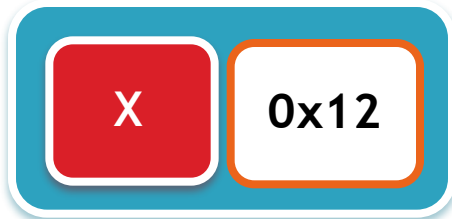


FUNCIONAMENTO DA TAD



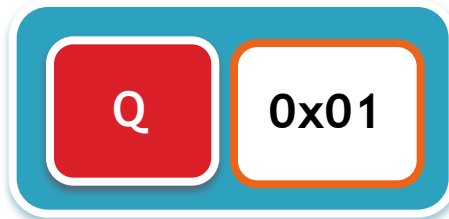
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TElemento



0x01

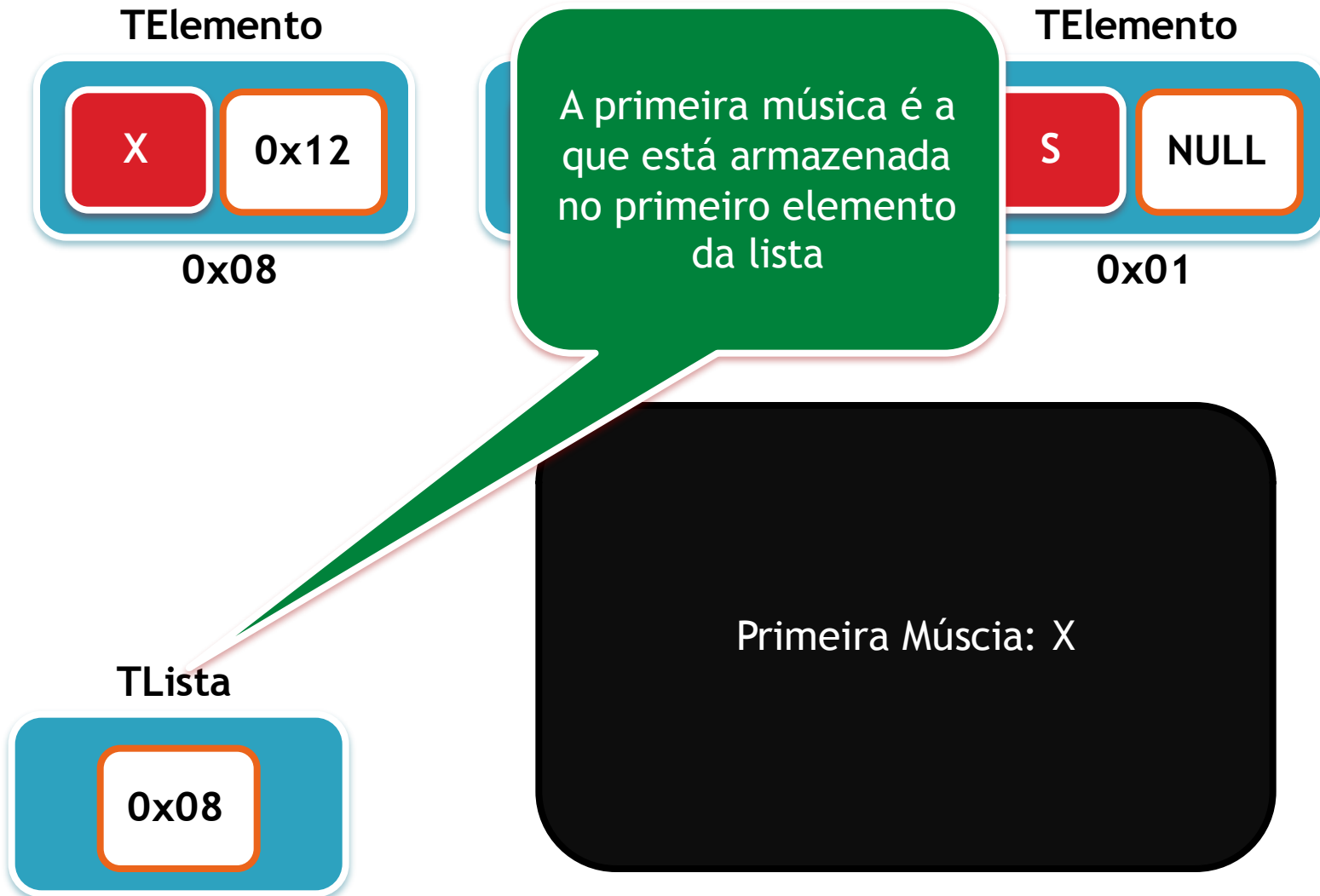
TLista



Deseja-se agora reproduzir as músicas armazenadas na lista na ordem estabelecida na mesma.

Qual seria a ordem?

FUNCIONAMENTO DA TAD



FUNCIONAMENTO DA TAD

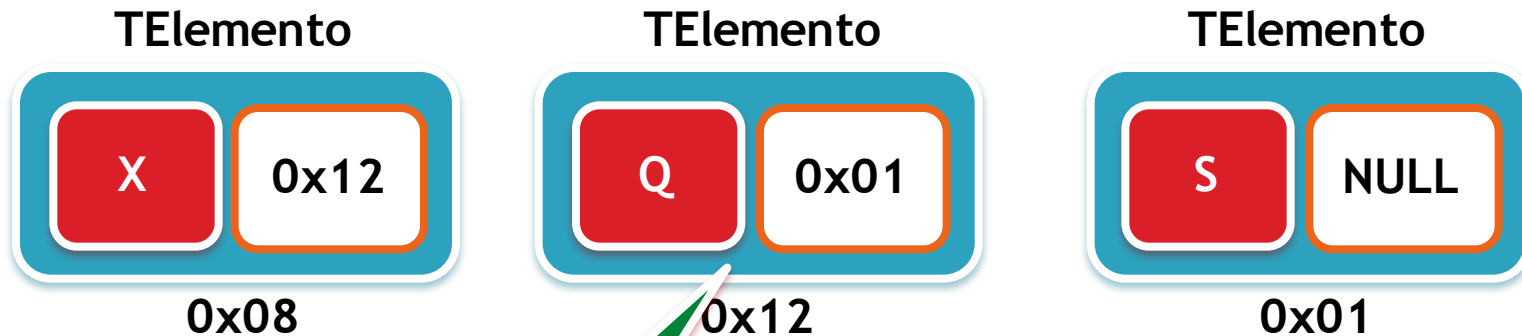


A segunda música está localizada no elemento com endereço igual ao armazenado no registro próximo do primeiro elemento.

0x08

Primeira Música: X
Segunda Música: Q

FUNCIONAMENTO DA TAD

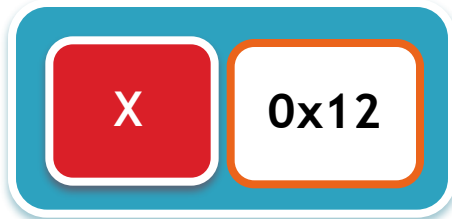


A terceira música está armazenada no elemento, cujo endereço é o que está gravado no registro próximo do elemento 0x12

Primeira Música: X
Segunda Música: Q
Terceira Música: S

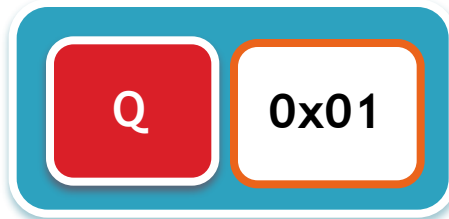
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TElemento



0x01

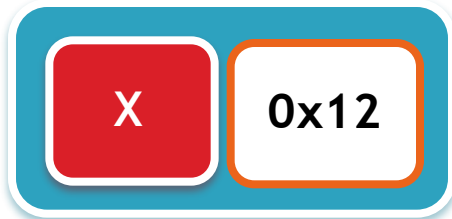
TLista



Iremos considerar a seguinte configuração de posições na lista: o primeiro elemento está na posição 0 , o segundo na posição 1 e assim sucessivamente.

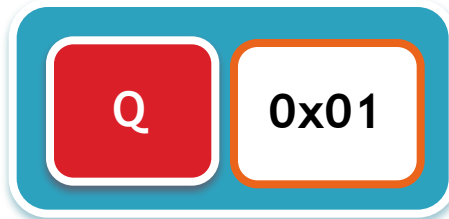
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TElemento



0x01

TLista

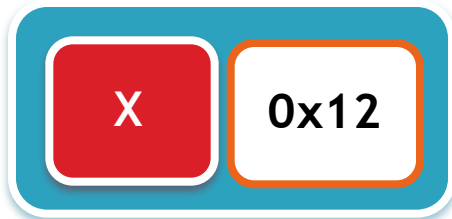


Logo, se inserir a música H na posição 1, ela deverá ser a segunda da lista!

Vamos fazer isso!

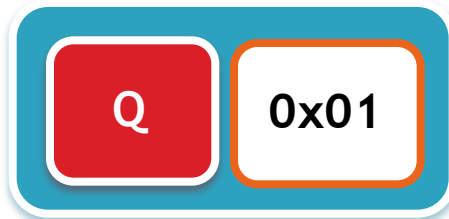
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TElemento



0x01

TMusica



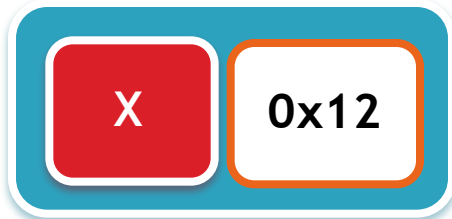
TLista



A música H está na memória.

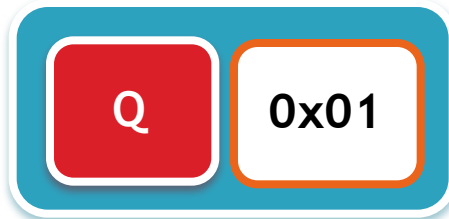
FUNCIONAMENTO DA TAD

TElemento



0x08

TElemento



0x12

TElemento



0x01

TElemento



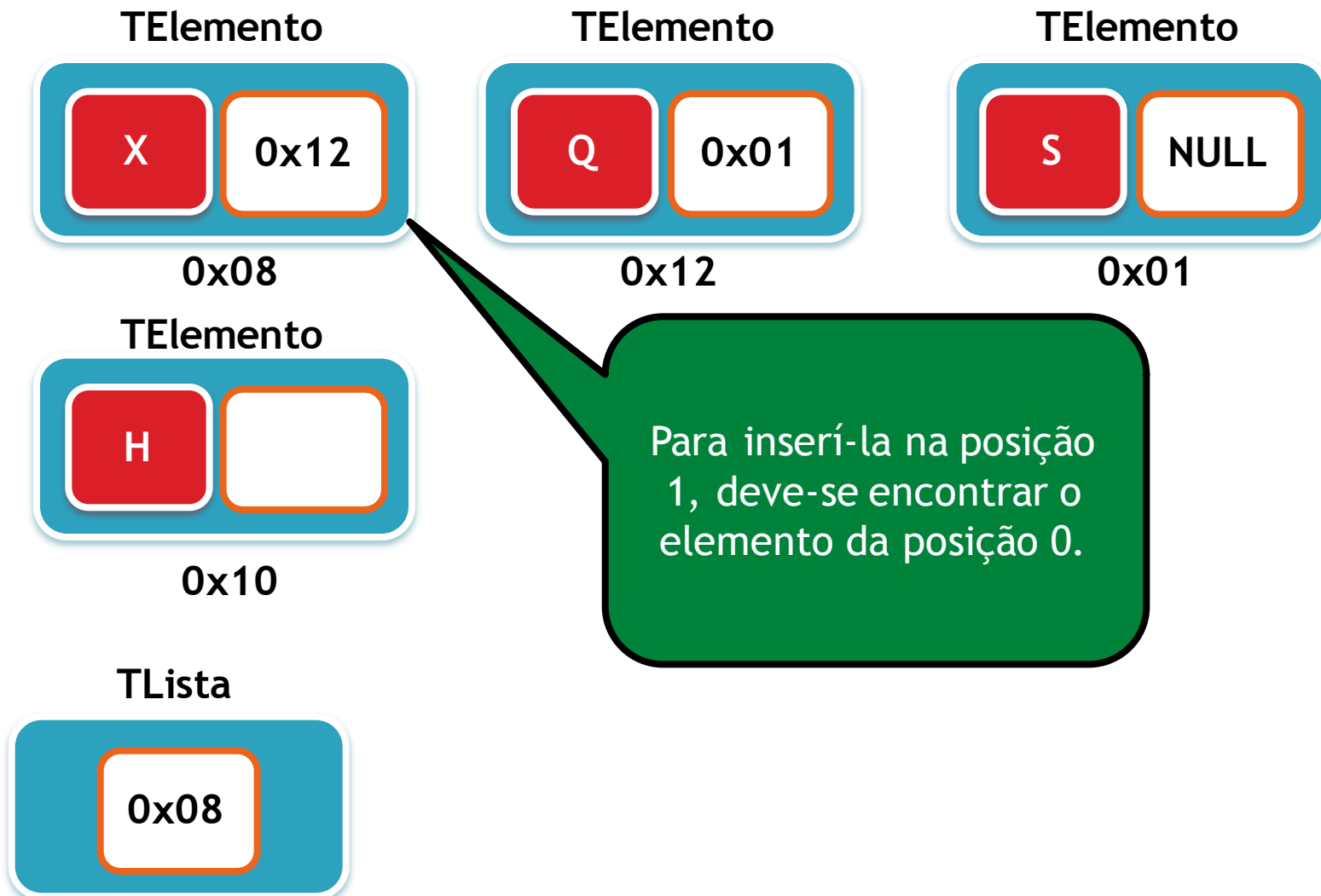
0x10

TLista

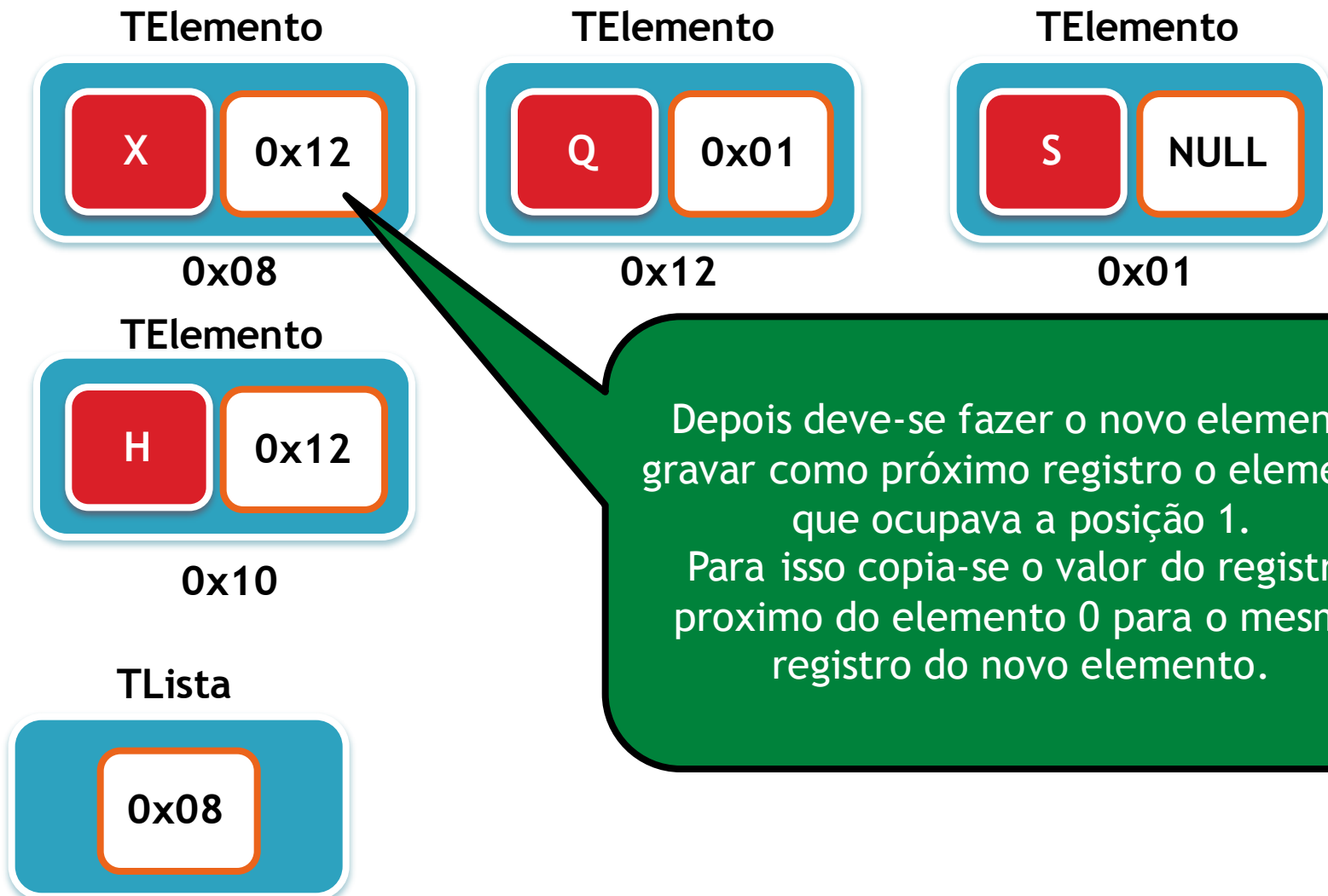


Para inserí-la na lista, deve-se envolvê-la em um elemento

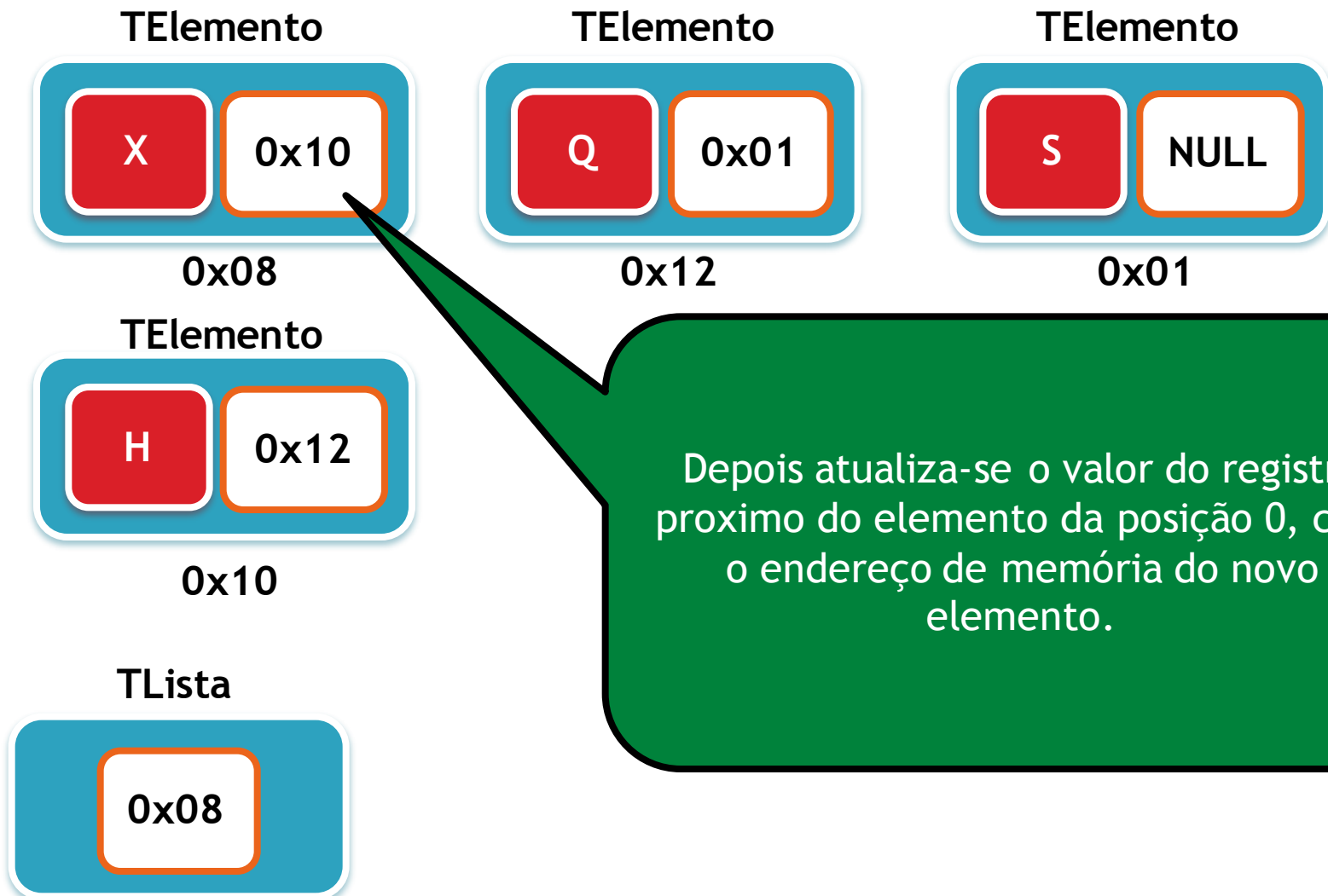
FUNCIONAMENTO DA TAD



FUNCIONAMENTO DA TAD

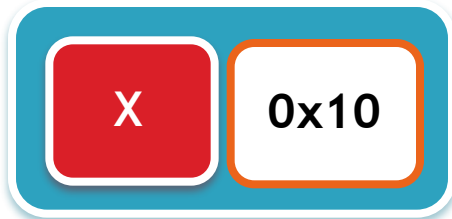


FUNCIONAMENTO DA TAD



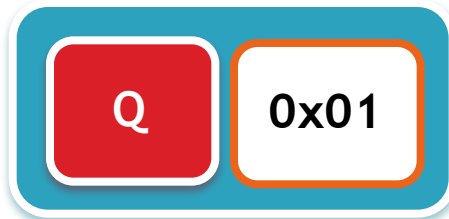
FUNCIONAMENTO DA TAD

TElemento



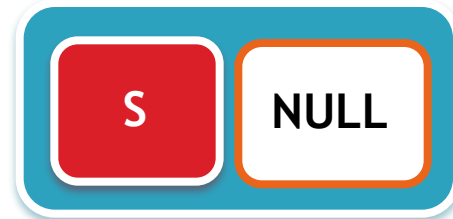
0x08

TElemento



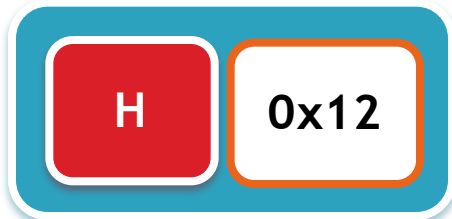
0x12

TElemento



0x01

TElemento



0x10

TLista



Primeira Música: X
Segunda Música: H
Terceira Música: Q
Quarta Música: S

Confirmam se o encadeamento está
correto.

FUNÇÕES DA TAD

◉ Implemente agora as funções:

- inicializa_lista_enc
- insere_final_lista_enc