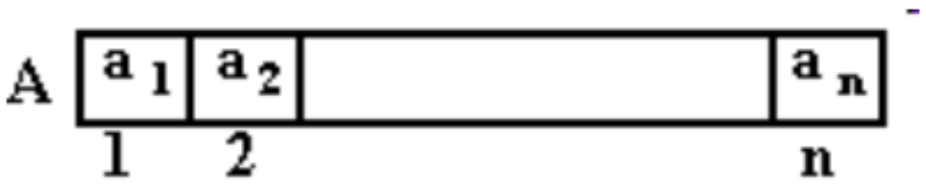
Listas

Listas Estáticas Sequenciais

(Fonte: Material adaptado dos Slides do prof. Monael.)

LES: Listas Estáticas Sequenciais

– **Estáticas**: Os elementos são armazenados em um vetor.



- Operações Básicas
  - Inserção de um elemento na Lista
  - Eliminação de um elemento da Lista
  - Consulta da pertinência de um elemento na Lista

LES: Listas Estáticas Sequenciais

- Seja:
  - L uma lista com n elementos e
  - i um índice da lista tal que $0 \leq i \leq n-1$

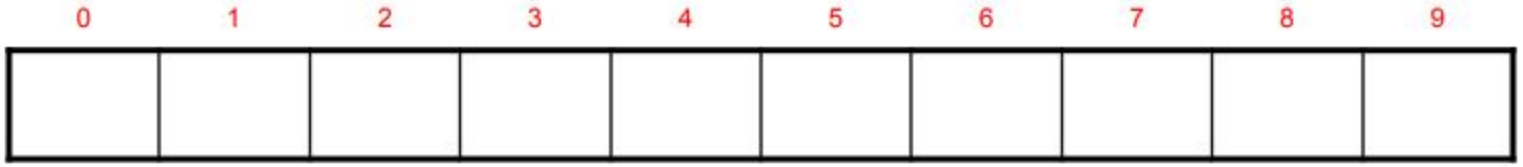
- Características de uma Lista L do tipo LES:

  - Os elementos da lista estão ordenados através de um campo chave;
  - São armazenados fisicamente em posições consecutivas;
  - A inserção de um elemento na posição L[i] causa o deslocamento a direita do elemento de L[i] ao último elemento;
  - A eliminação do elemento L[i] requer o deslocamento à esquerda do elemento L[i+1] ao último;

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 0



Quantidade igual a ZERO, significa que a lista está vazia.
Portanto:

- Inserir o elemento na posição Quantidade
- Incrementar a Quantidade em uma unidade

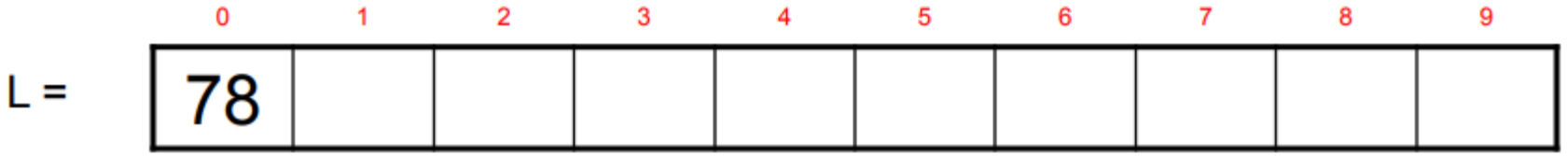
→ Item a ser inserido: 78

(Inserção em uma lista vazia.)

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 1



Item a ser inserido: 78

→ Quantidade igual a ZERO, significa que a lista está vazia.
Portanto:

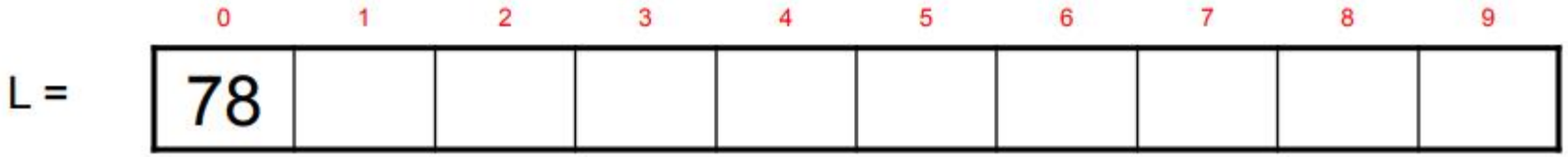
- Inserir o elemento na posição Quantidade
- Incrementar a Quantidade em uma unidade

(Inserção em uma lista vazia.)

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 1



→ Item a ser inserido: 92

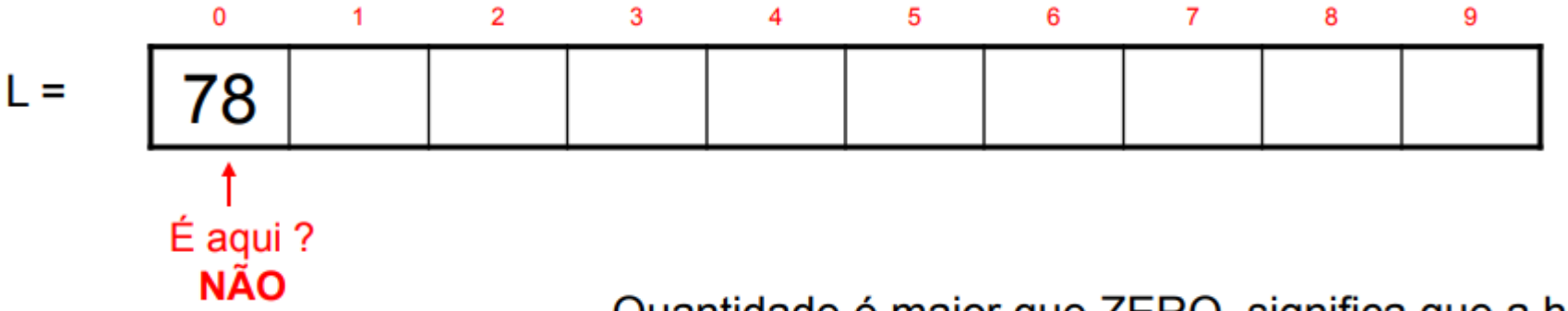
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 1



Item a ser inserido: 92

(Inserção no fim da lista.)

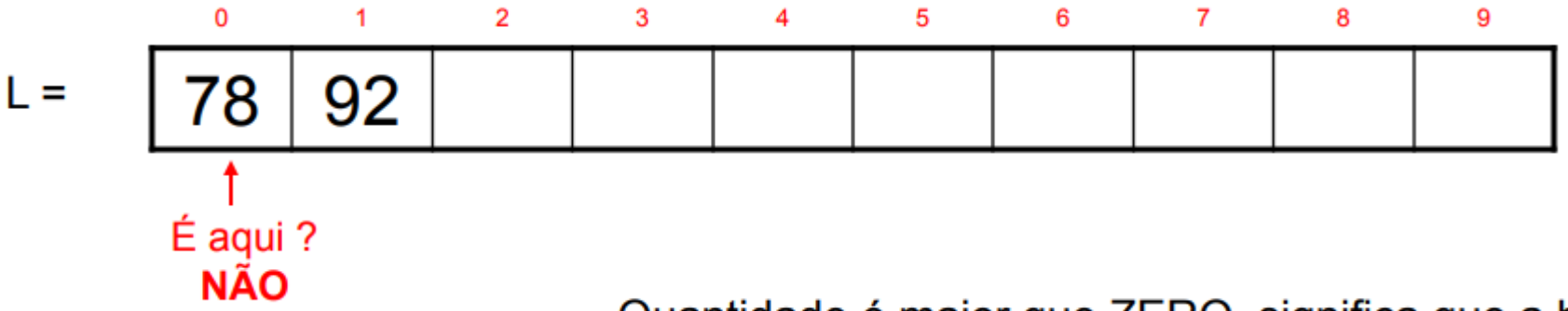
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 2



Item a ser inserido: 92

(Inserção no fim da lista.)

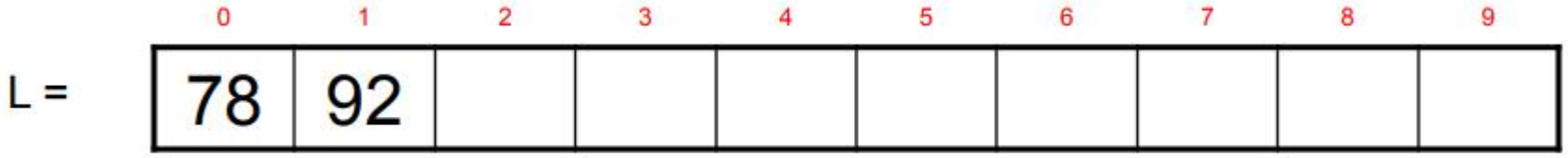
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 2



→ Item a ser inserido: 81

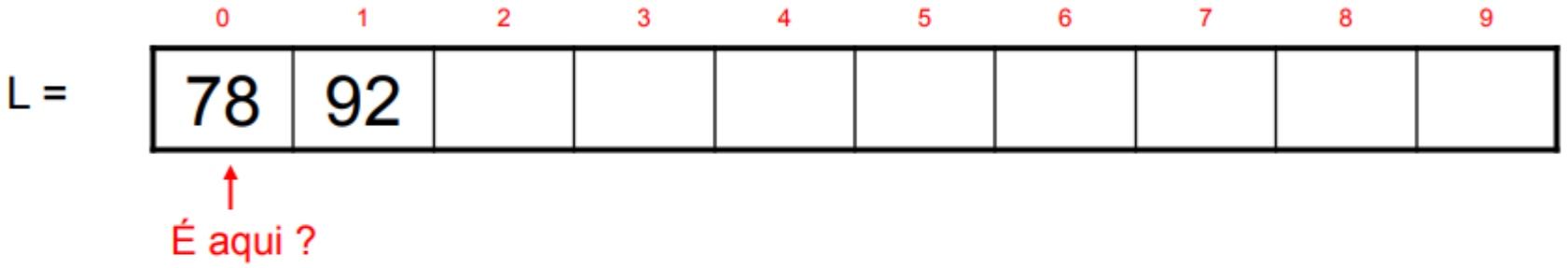
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 2



Item a ser inserido: 81

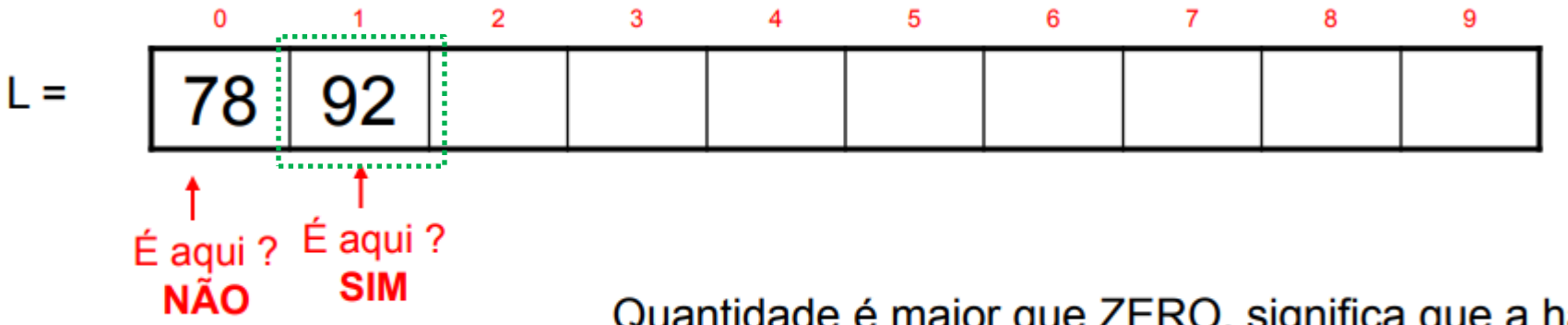
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 2



Item a ser inserido: 81

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

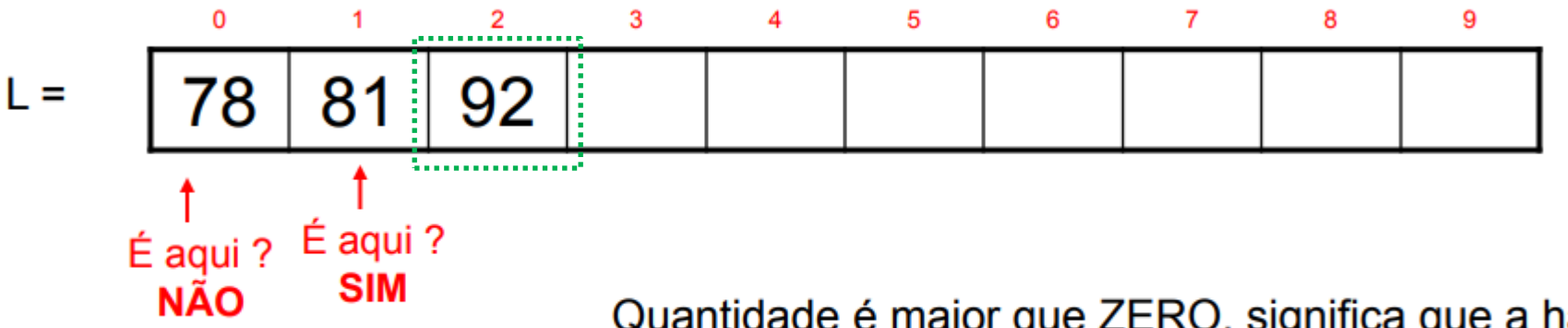
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no meio da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 3



Item a ser inserido: 81

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

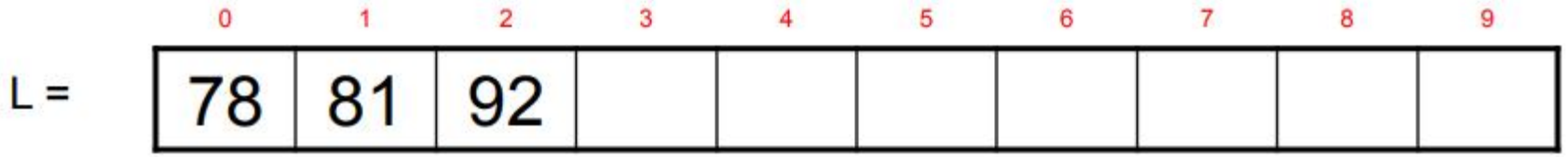
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no meio da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 3



→ Item a ser inserido: 45

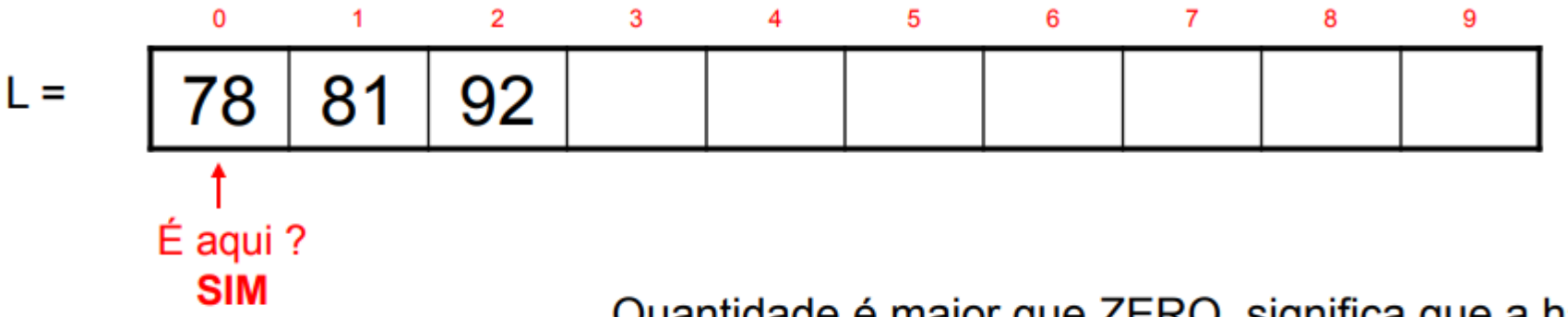
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 3



Item a ser inserido: 45

(Inserção no início da lista.)

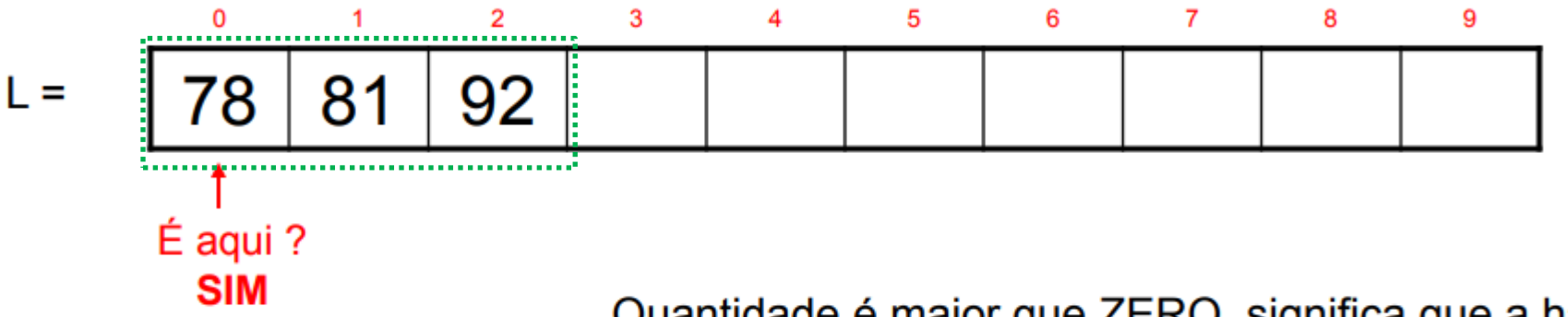
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

→
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 3



Item a ser inserido: 45

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

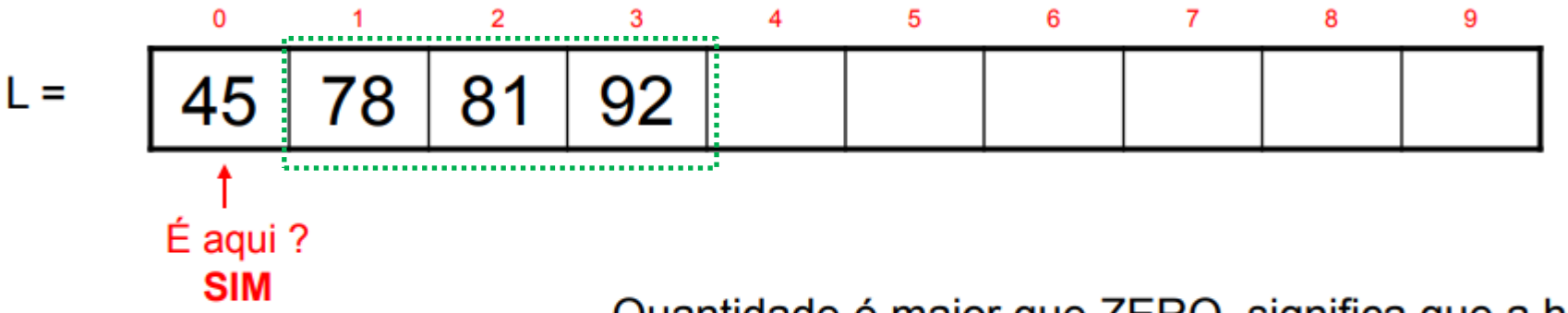
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no início da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 4



Item a ser inserido: 45

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

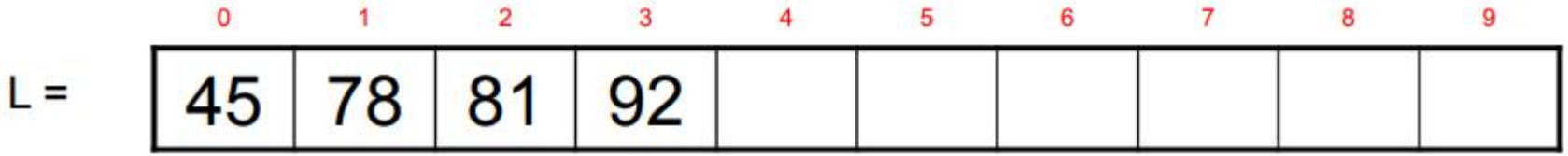
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no início da lista.)

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 4



→ Item a ser inserido: 36

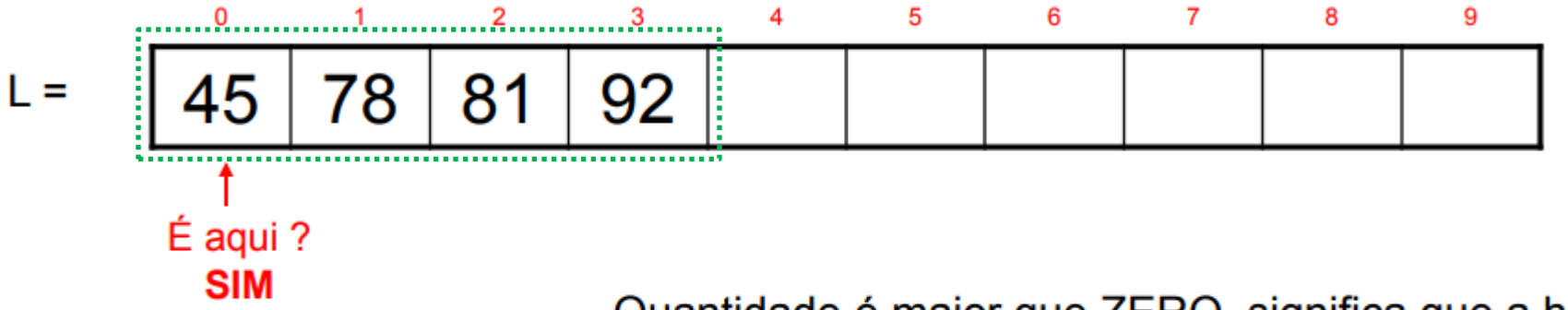
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 4



Item a ser inserido: 36

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

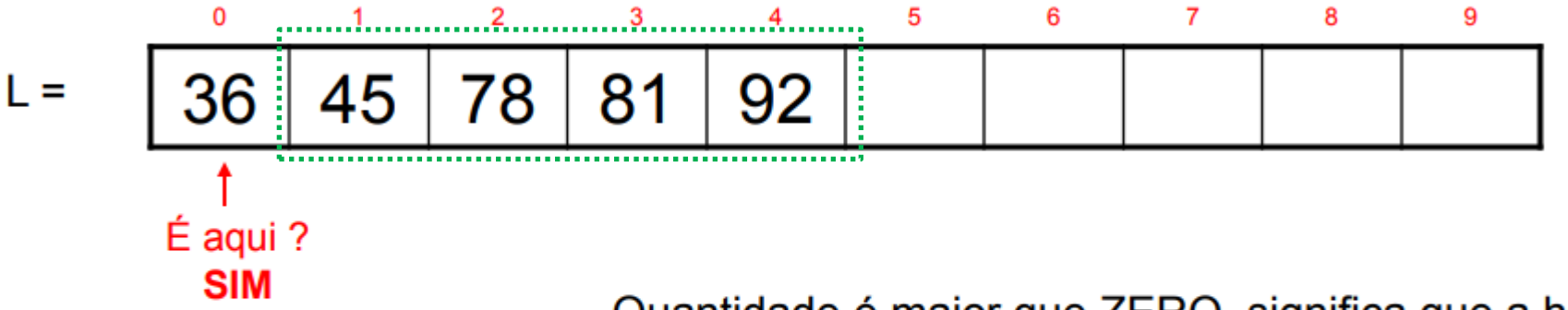
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no início da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 5



Item a ser inserido: 36

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

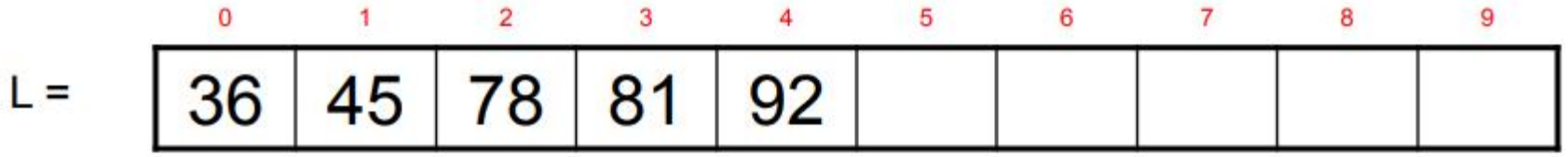
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no início da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 5



→ Item a ser inserido: 98

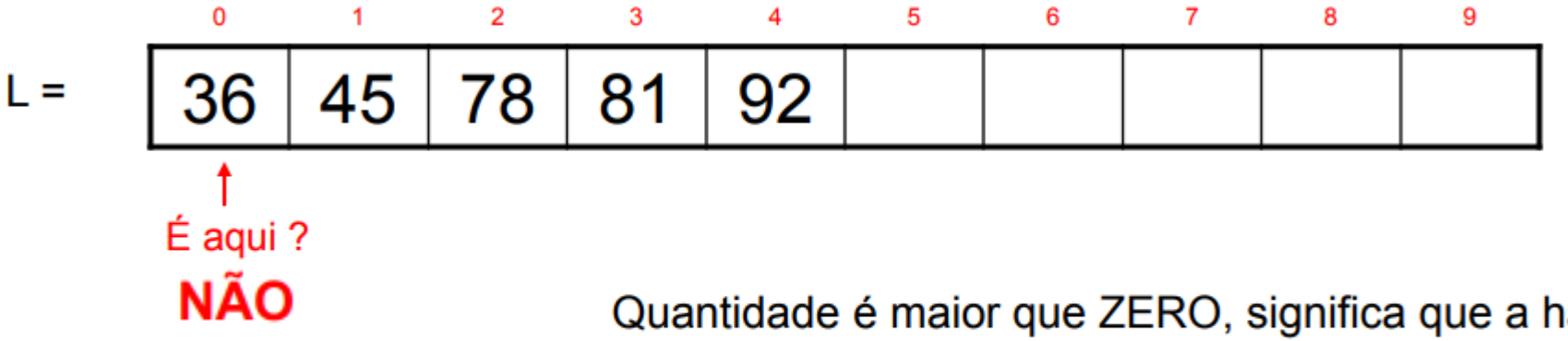
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 5



Item a ser inserido: 98

(Inserção no fim da lista.)

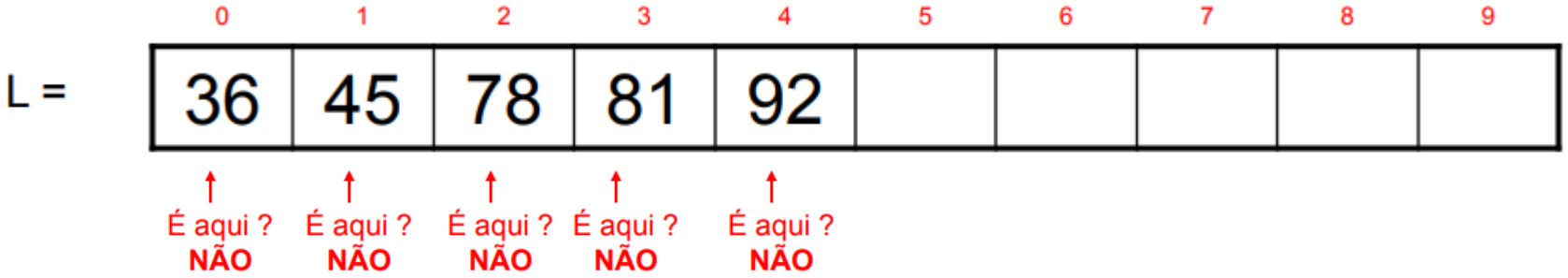
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

→
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 5



Item a ser inserido: 98

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

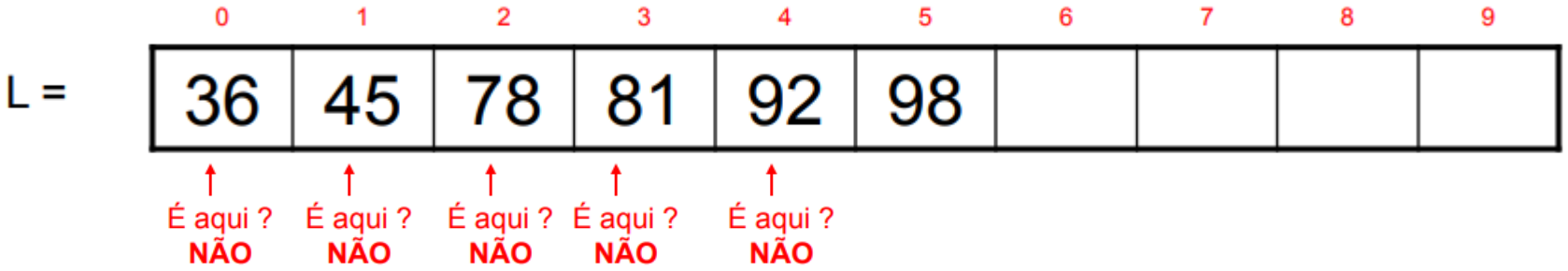
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no fim da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 6



Item a ser inserido: 98

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

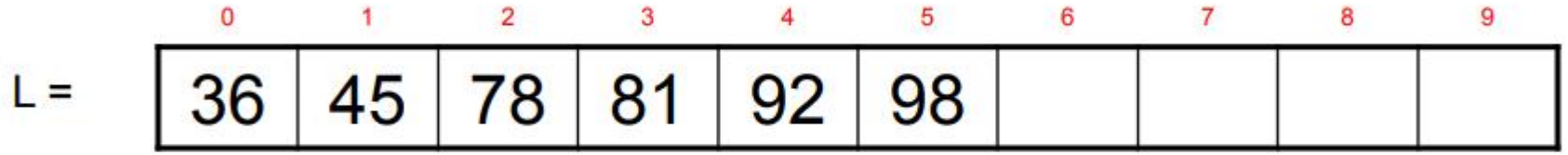
(Inserção no fim da lista.) →

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 6



→ Item a ser inserido: 80

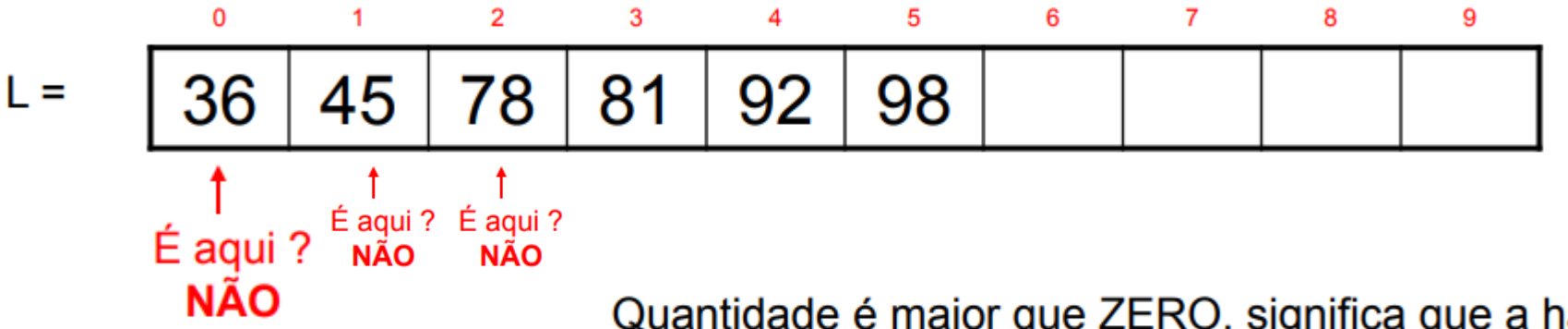
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 6



Item a ser inserido: 80

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

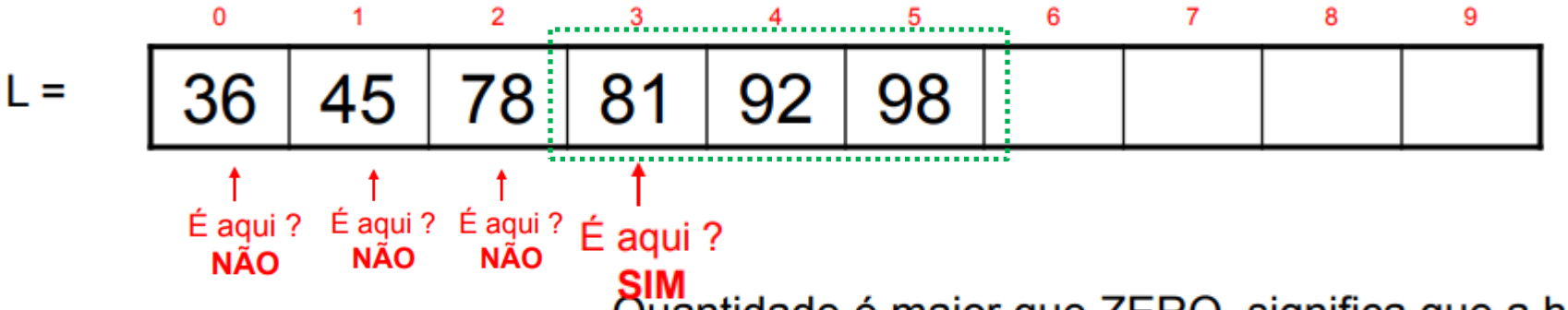
→
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no meio da lista.)

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 6



Item a ser inserido: 80

Quantidade é maior que ZERO, significa que a há itens na lista.

Portanto:

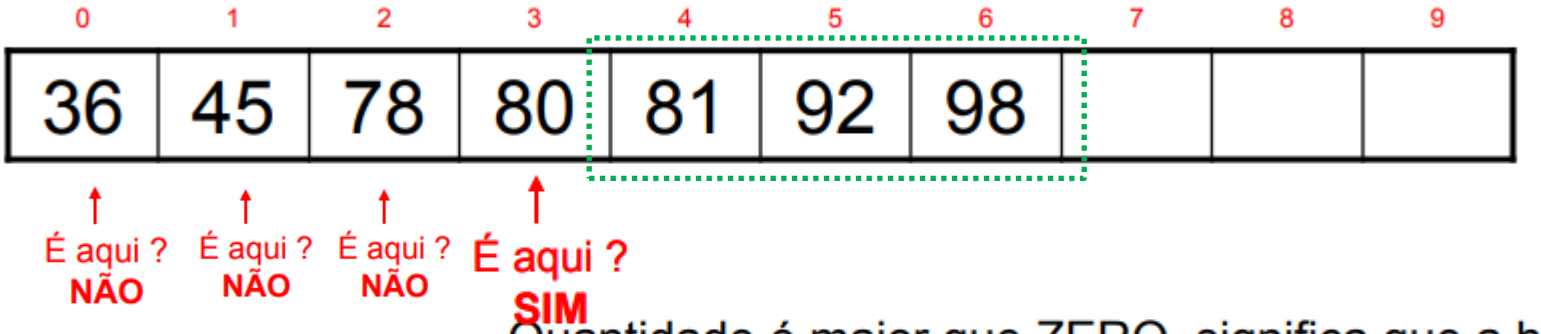
- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no meio da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

Quantidade: 7



Item a ser inserido: 80

Quantidade é maior que ZERO, significa que a há itens na lista.

Portanto:

- procurar a posição para inserir o elemento.
- ao encontrar inserir o elemento.
- incrementar a Quantidade em uma unidade

(Inserção no meio da lista.) →

LES: Listas Estáticas Sequenciais

- 1) Operação: Inserção

(Lembretes...)

(Vazia) – Caso 1: Inserir primeiro elemento em uma lista vazia:
  - Checar a quantidade, se for Zero adicionar no índice 0 da lista.

(Início) – Caso 2: Inserir primeiro elemento de uma lista não vazia:
  - Checar a quantidade, se for maior que Zero, então desloque todos os itens da lista um item para a direita e adicione o item novo no índice 0 da lista.

(Fim) – Caso 3: Inserir o último elemento de uma lista:
  - Quando chegar ao final da lista (atual igual quantidade). Adicione o item novo no índice quantidade da lista (última posição).

(Meio) – Caso 4: Inserir no meio da lista:
  - Quando encontra a posição de inserção do novo elemento. Desloque todos os elementos do final da lista até a posição de inserção um índice para direita e adicione o novo item na posição correta.

LES: Listas Estáticas Sequenciais

- 2) Operação: Busca

Como faremos a consulta de um item na lista L ?

- Busca Sequencial (Linear)
- Busca Binária

```
// Função recebe um vetor, o tamanho, e o elemento buscado.
// Devolve um índice do vetor ou -1 caso não encontre o
// elemento.
int buscaBin (int *v, int n, int x)
```

LES: Listas Estáticas Sequenciais

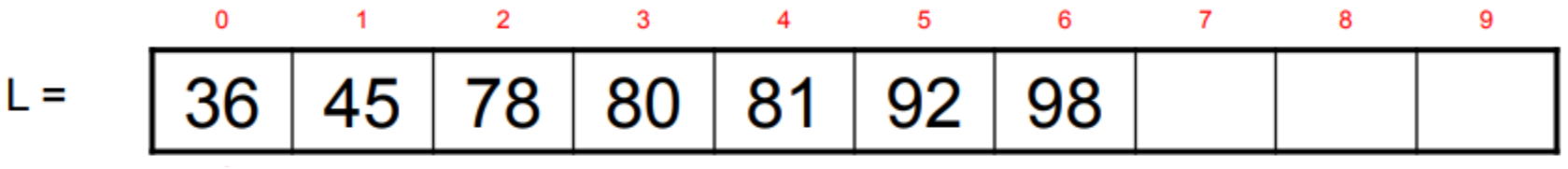
- 3) Operação: Remoção

  - Observe cada situação da remoção
    - Remover do início da Lista
    - Remover do meio da Lista
    - Remover do final da Lista

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 7



→ Item a ser removido: 36

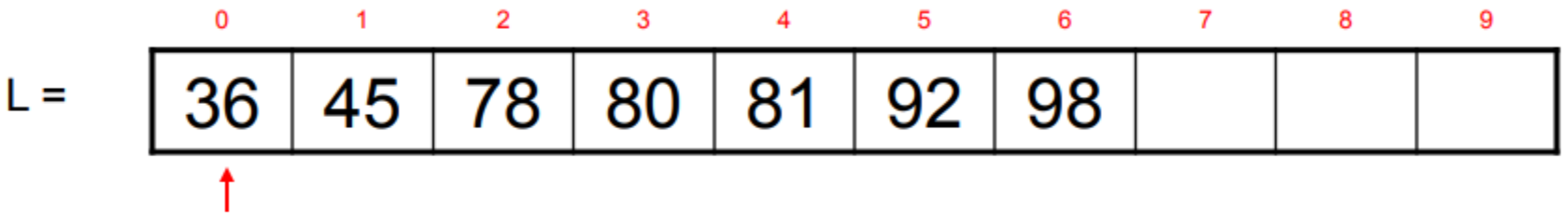
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 7



busca
devolve 0

Item a ser removido: 36

(Remoção do início da lista.)

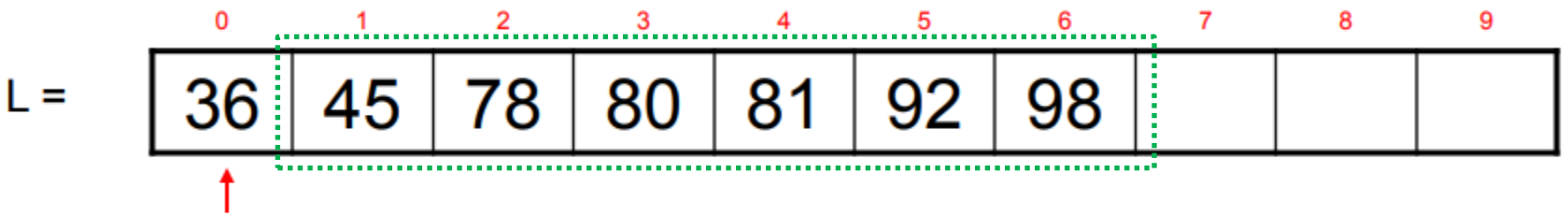
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 7



busca
devolve 0

Item a ser removido: 36

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

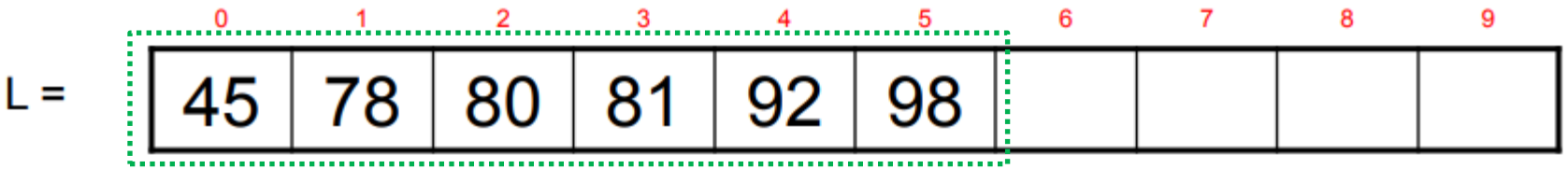
- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

(Remoção do início da lista.)

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 6



Item a ser removido: 36

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

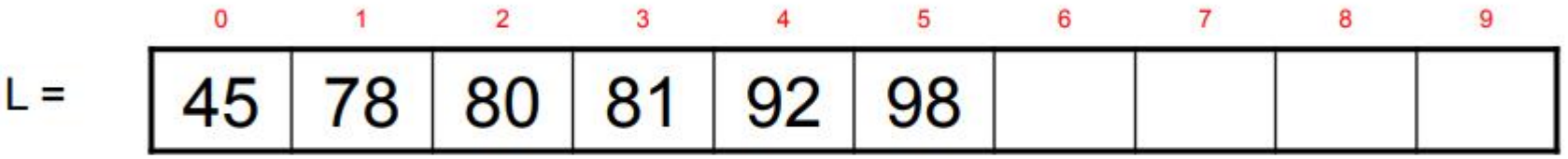
- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

(Remoção do início da lista.)

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 6



→ Item a ser removido: 98

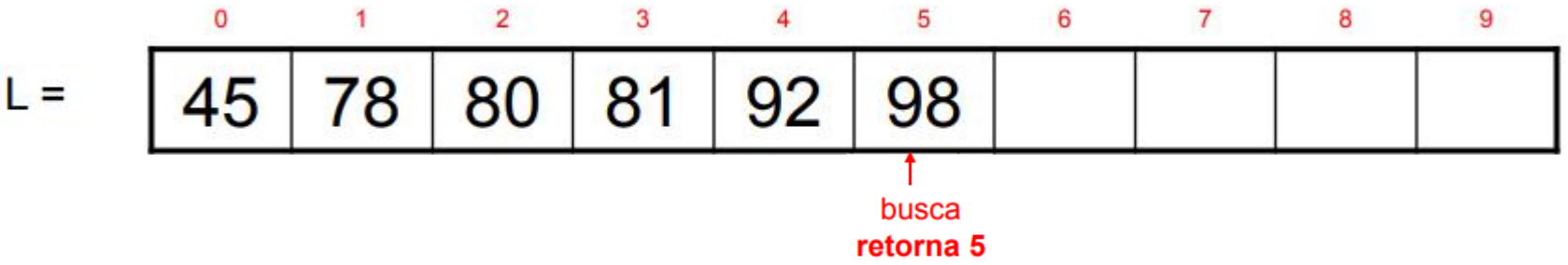
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 6



Item a ser removido: 98

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

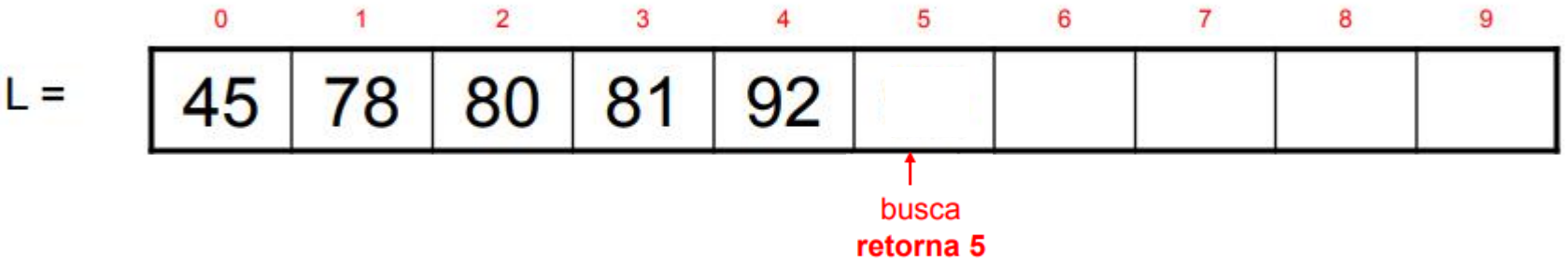
- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

(Remoção do fim da lista.)

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 5



Item a ser removido: 98

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

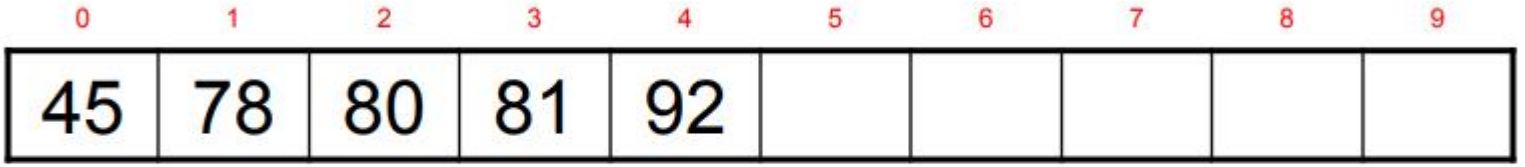
- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

(Remoção do fim da lista.)

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 5



→ Item a ser removido: 80

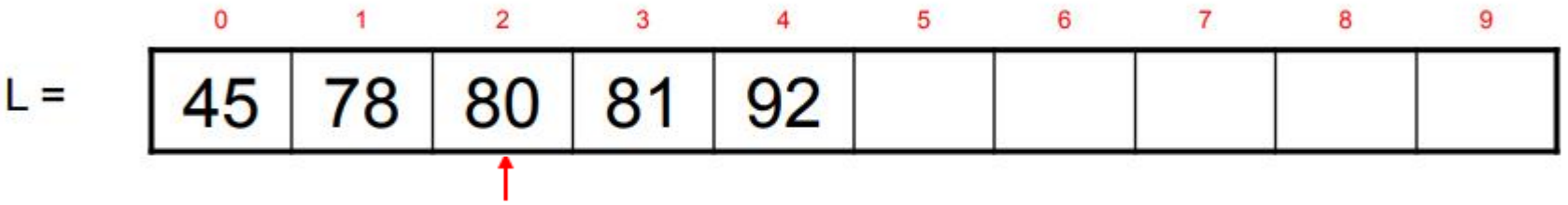
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 5



Item a ser removido: 80

(Remoção do meio da lista.)

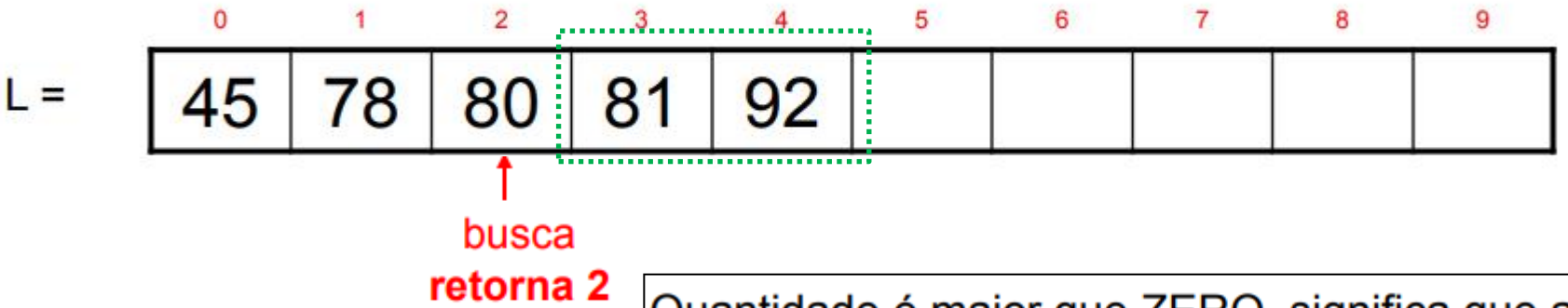
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

→ • Invoque a rotina de busca para o elemento a ser removido.
• Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
• Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 5



Item a ser removido: 80

(Remoção do meio da lista.)

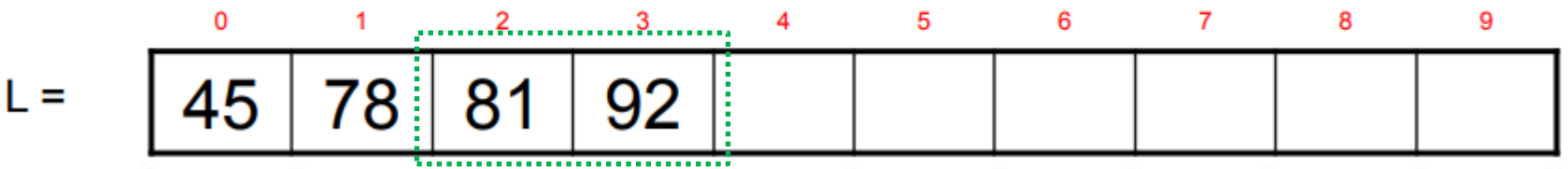
Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- 3) Operação: Remoção

Quantidade: 4



Item a ser removido: 80

(Remoção do meio da lista.)

Quantidade é maior que ZERO, significa que a há itens na lista.
Portanto:

- Invoque a rotina de busca para o elemento a ser removido.
- Caso o elemento esteja na lista, remova-o e decremente Quantidade de uma unidade
- Caso contrário, exiba uma mensagem.

LES: Listas Estáticas Sequenciais

- Estruturas:
  - tLista
  - tItem → Por simplificação do exemplo nossos itens serão inteiros.

LES: Listas Estáticas Sequenciais

- Implementação da Estrutura da Lista

```
1.  struct tLista
2.  {
3.        int *itens;
4.        int quantidade;
5.        int tamanho;
6.  };
```

Por simplificação os itens são um vetor de inteiros.

- Inicialização da Lista

```
1.  void iniciaLista(struct tLista *lista, int n)
2.  {
3.        lista->itens = (int*) malloc(n * sizeof(int));
4.        lista->tamanho = n;
5.        lista->quantidade = 0;
6.  }
```

LES: Listas Estáticas Sequenciais

- Sugestões para Funções:
  - `void iniciaLista(struct tLista *, int);`
  - `int busca(struct tLista, int);`
    - `int buscaLinear(struct tLista);`
    - `int buscaBinaria(struct tLista, int, int, int);`
  - `int lerItem(void);`
  - `void insere(struct tLista *, int);`
    - `void deslocaDir(struct tLista*, int);`
  - `void consulta(struct tLista, int);`
  - `int remove(struct tLista *);`
    - `void deslocaEsq(struct tLista*, int);`