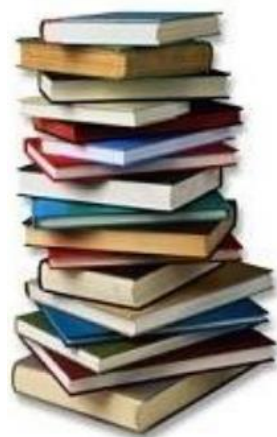


# Estruturas de Dados

## Pilhas

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

# Pilhas



- Uma pilha é uma estrutura de dados que admite inserção e remoção de itens apenas em uma extremidade, chamada topo.
- Mais especificamente, uma pilha “*stack*” é uma estrutura sujeita à seguinte regra de operação:
  - sempre que houver uma inclusão ou remoção de elemento isso acontecerá no topo.
  - Ao remover um elemento, deixará a pilha o que estiver a menos tempo na pilha.
- Em outras palavras, o último objeto inserido na pilha é também o primeiro a ser removido.
- Essa política é conhecida pela sigla LIFO (*Last-In-First-Out*).

# Pilhas

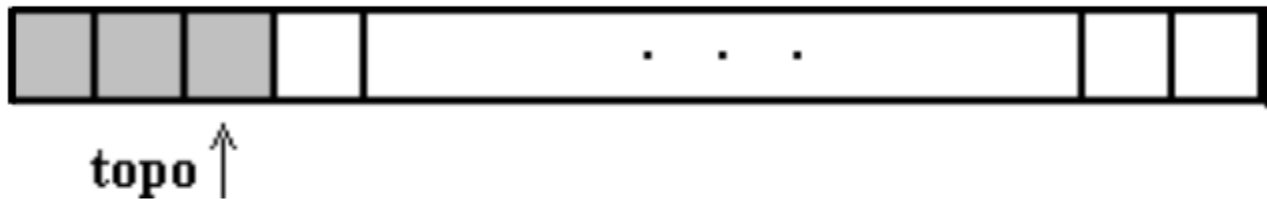
- Motivação:
  - Por que usar pilhas?
    - Custo

Operação	Listas		Pilha
Inserção	$O(n)$		$O(1)$
Remoção	$O(n)$		$O(1)$
Busca	Linear	Binária	$O(n)^*$
	$O(n)$	$O(\log_2 n)$	

\* Não se aplica a busca propriamente em uma pilha, na realidade se desempilha itens (remoção) até que se encontre o item desejado.

# Pilhas

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



# Pilhas Estáticas

- Seja:
  - S uma pilha com n elementos e
  - i um índice tal que  $0 \leq i \leq n-1$
- Características de uma Pilha S:
  - Os itens da pilha são dispostos conforme sua entrada na pilha.
  - Só é permitido a inclusão de itens no topo da pilha. (LIFO)
  - São armazenados fisicamente em posições consecutivas;
  - A eliminação de itens só é permitida do topo da pilha. (LIFO)
  - LIFO (Last In First Out) – “Último que Entra Primeiro que Sai”

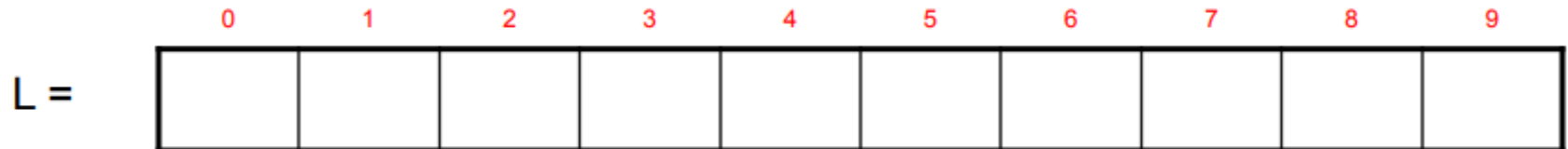
# Pilhas Estáticas

- Operações Básicas
  - Inserção de um elemento na Pilha (Empilhar)
  - Eliminação de um elemento da Pilha (Desempilhar)

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)

topo : -1

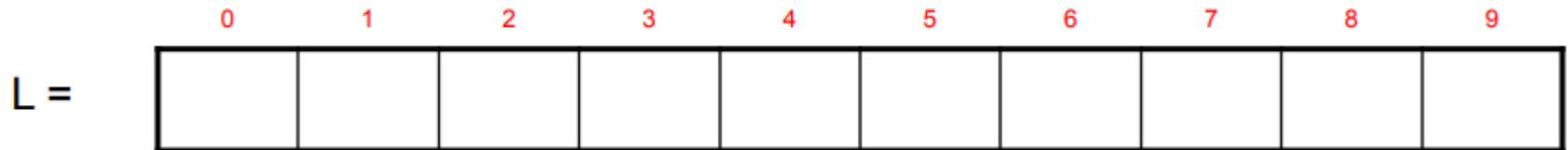


Para fins de simplificação do exemplo de funcionamento, usaremos um vetor de inteiros. Em uma aplicação ter-se-ia um vetor de estruturas.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)

topo : -1



Se a pilha não estiver cheia.

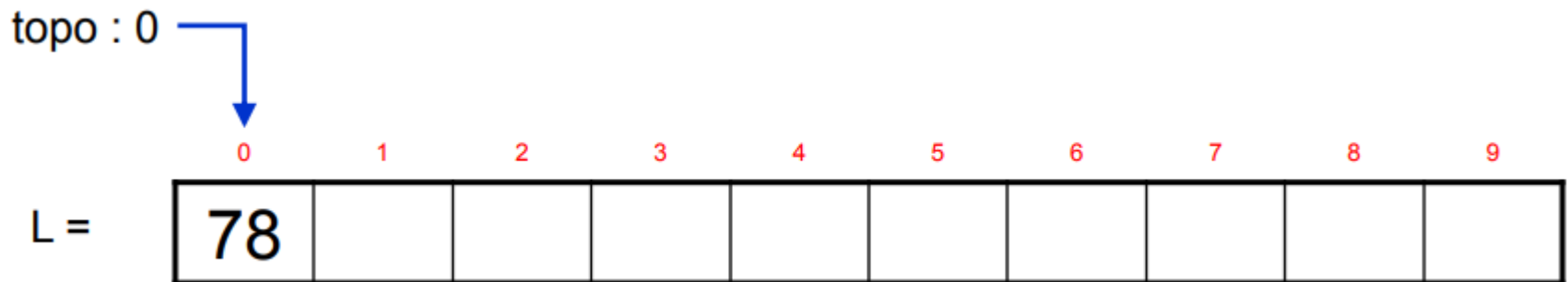
→ Item a ser inserido: 78

- Incrementar topo de uma unidade.
- Empilhar o elemento na posição topo.



# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



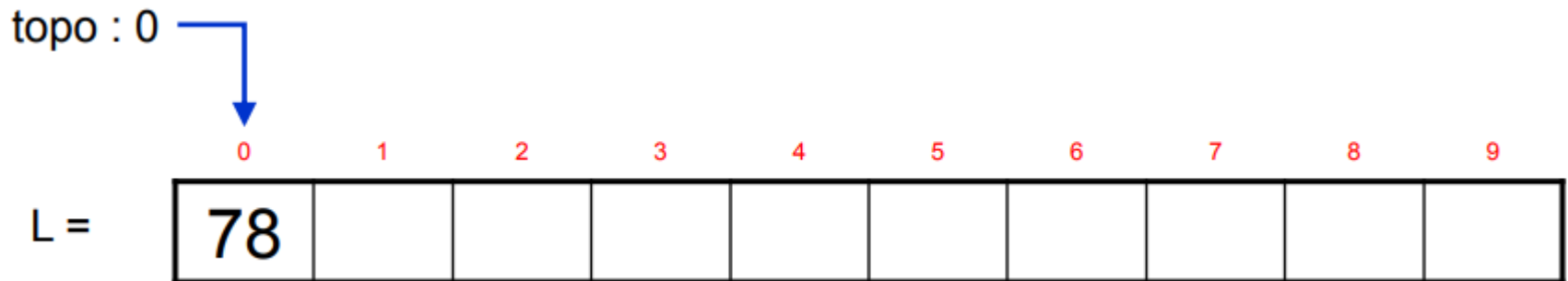
Se a pilha não estiver cheia.

Item a ser inserido: 78

- Incrementar topo de uma unidade.
- • Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



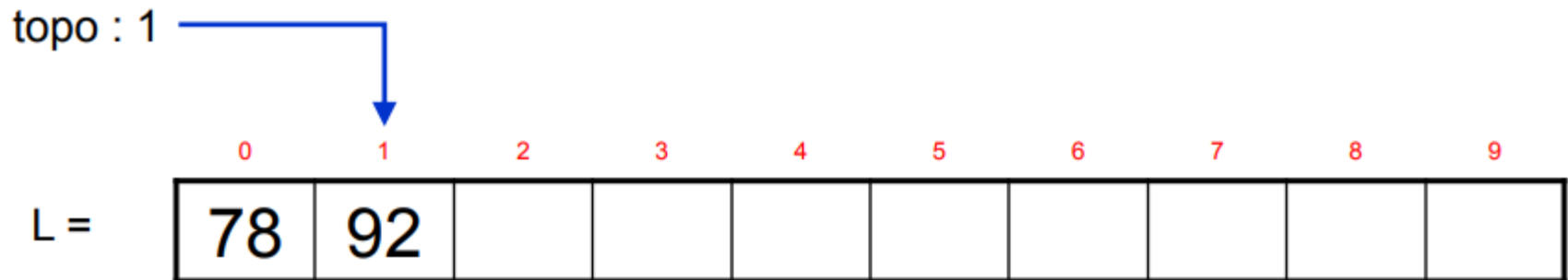
Se a pilha não estiver cheia.

→ Item a ser inserido: 92

- Incrementar topo de uma unidade.
- Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



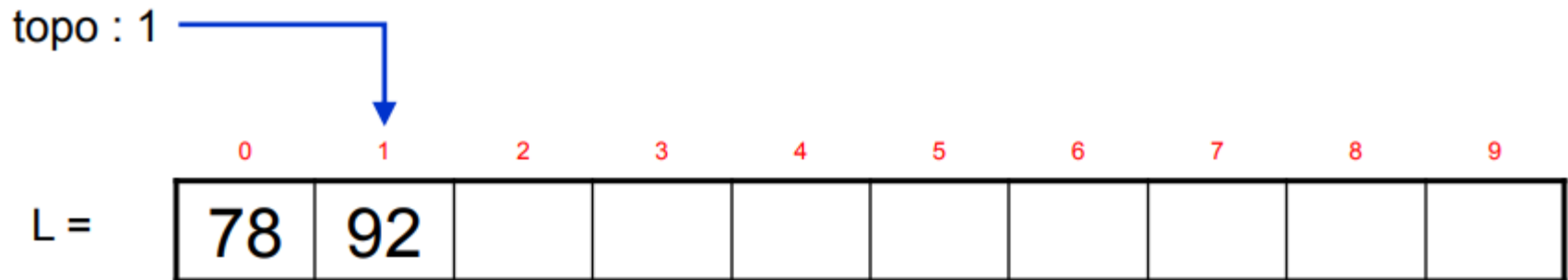
Se a pilha não estiver cheia.

Item a ser inserido: 92

- Incrementar topo de uma unidade.
- • Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



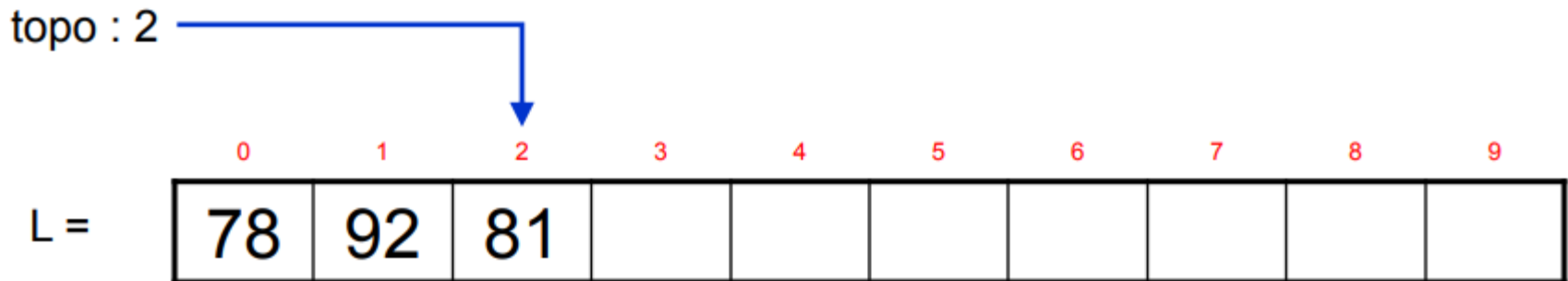
Se a pilha não estiver cheia.

→ Item a ser inserido: 81

- Incrementar topo de uma unidade.
- Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



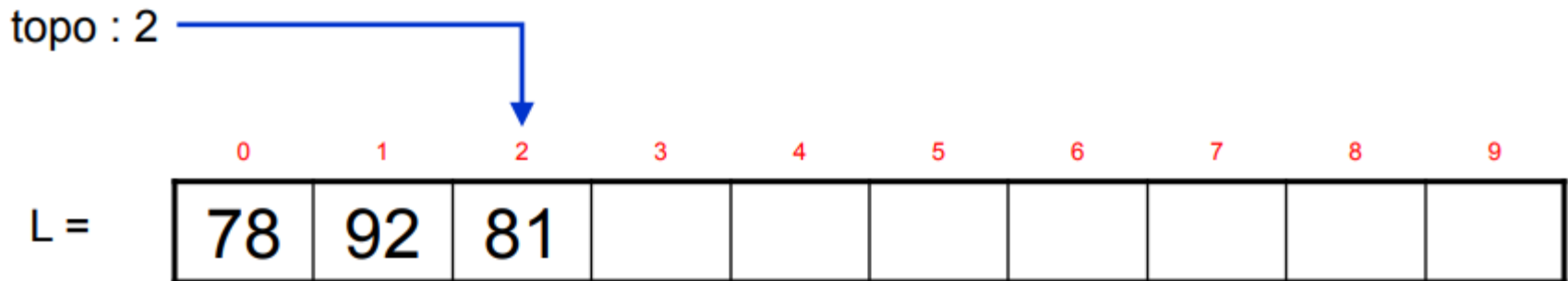
Se a pilha não estiver cheia.

Item a ser inserido: 81

- Incrementar topo de uma unidade.
- • Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



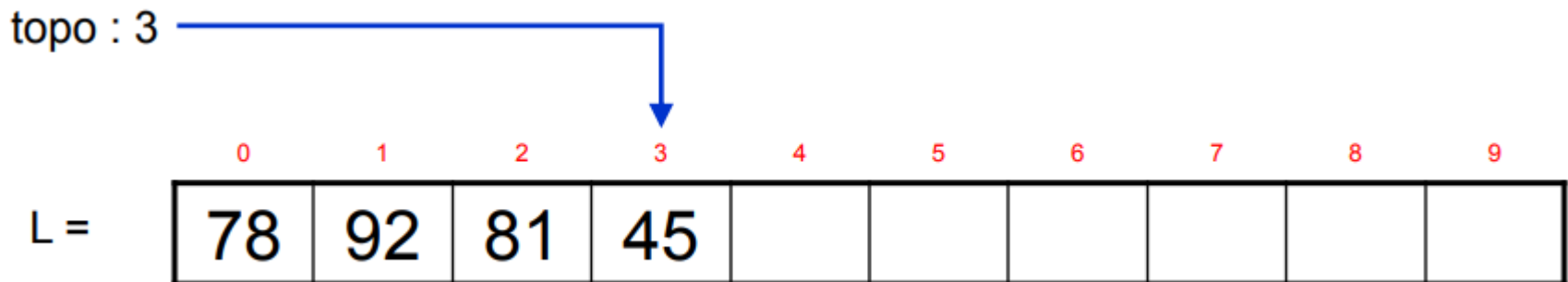
Se a pilha não estiver cheia.

→ Item a ser inserido: 45

- Incrementar topo de uma unidade.
- Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



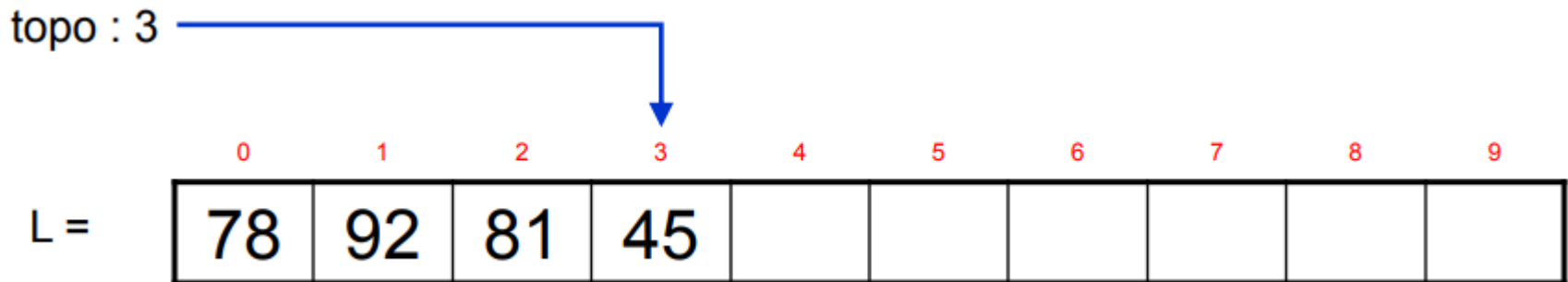
Se a pilha não estiver cheia.

Item a ser inserido: 45

- Incrementar topo de uma unidade.
- • Empilhar o elemento na posição topo.

# Pilhas Estáticas

- 1) Operação: **Inserção** (**Empilhar**)



Se a pilha não estiver cheia.

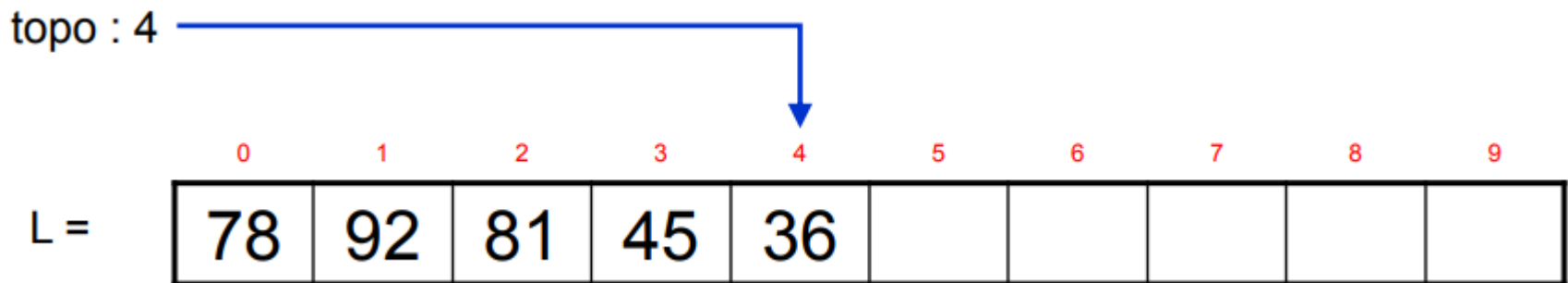
→ Item a ser inserido: 36

- Incrementar topo de uma unidade.
- Empilhar o elemento na posição topo.



# Pilhas Estáticas

- 1) Operação: **Inserção** (Empilhar)



Se a pilha não estiver cheia.

Item a ser inserido: 36

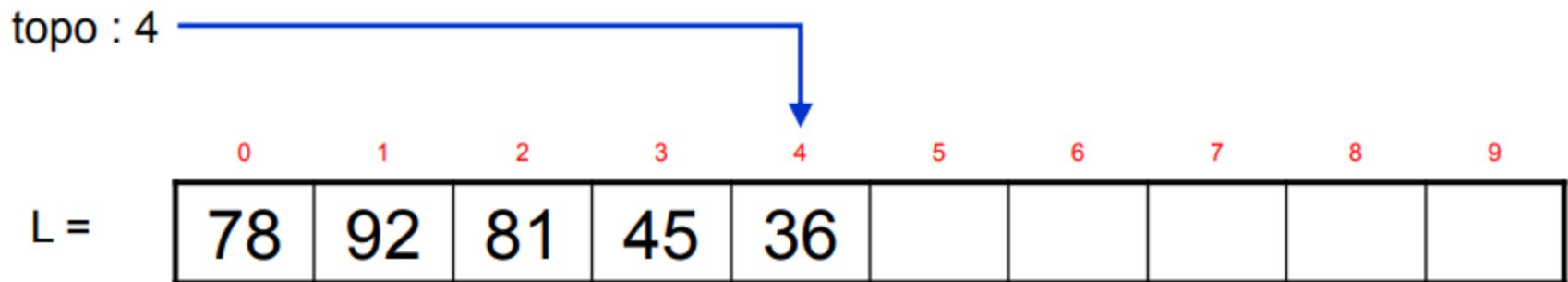
- Incrementar topo de uma unidade.
- • Empilhar o elemento na posição topo.

# Pilhas Estáticas

- Empilhar (Lembrete)
  - Caso Geral
    - Se a Pilha Não Estiver Cheia
      - Incrementar o topo
      - Empilhar
  - Como saber se a pilha está cheia:
    - Se topo for igual ao tamanho do vetor
      - Então a pilha está cheia

# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)



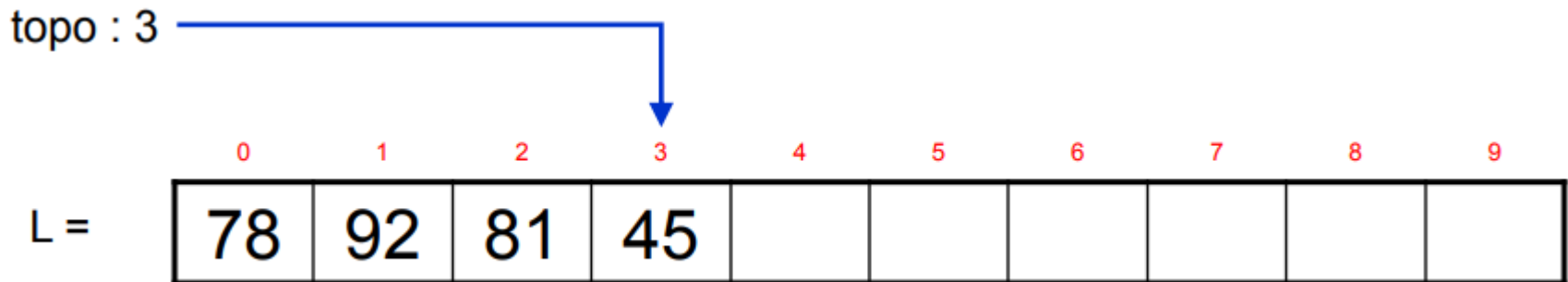
Se a pilha não estiver vazia

→ Item a ser removido: ???

- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado

# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)



Item removido:

36

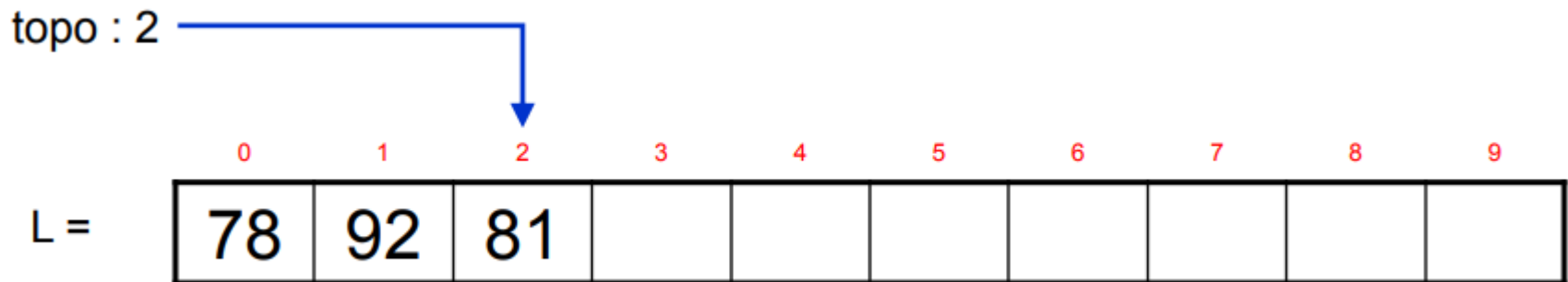
Se a pilha não estiver vazia

→ Próximo Item a ser removido: ???

- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado

# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)



Item removido:

45

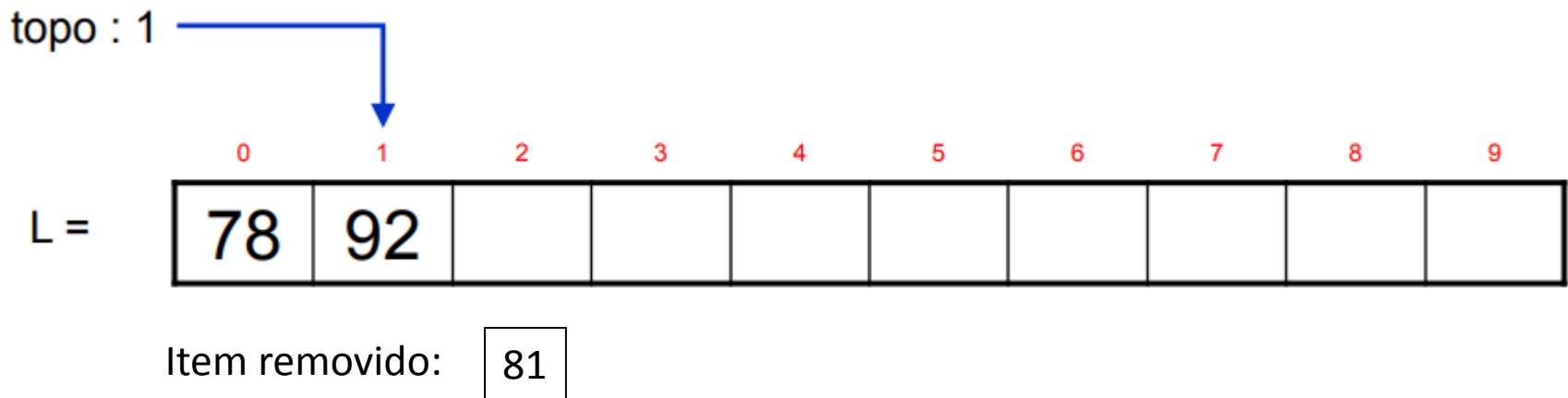
Se a pilha não estiver vazia

→ Próximo Item a ser removido: ???

- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado

# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)



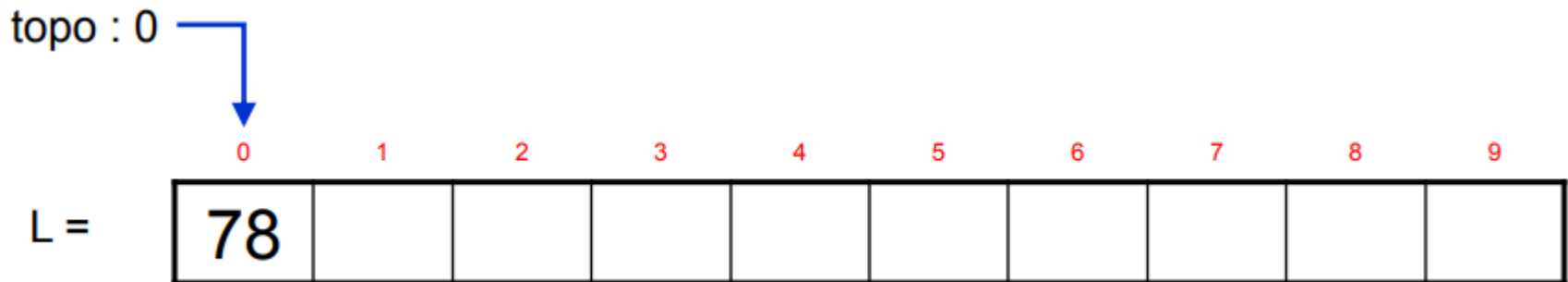
Se a pilha não estiver vazia

→ Próximo Item a ser removido: ???

- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado

# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)



Item removido: 92


Se a pilha não estiver vazia

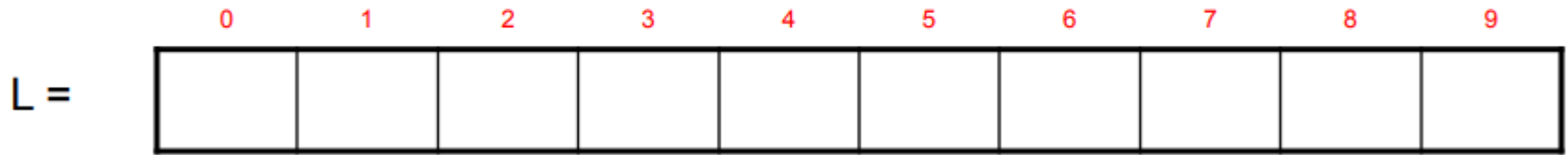
→ Próximo Item a ser removido: ???

- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado

# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)

topo : -1 



Item removido: 78

Se a pilha não estiver vazia


→ Próximo Item a ser removido: ???

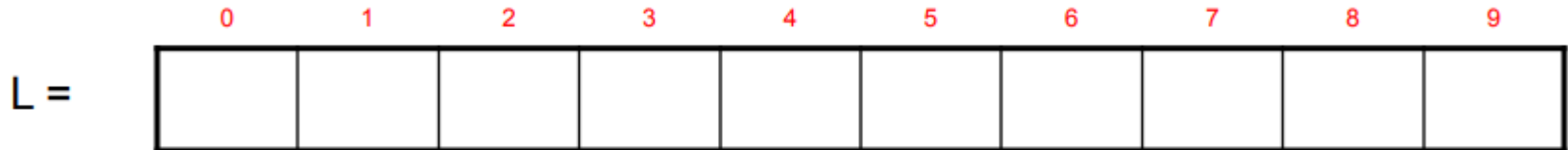
- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado



# Pilhas Estáticas

- 2) Operação: **Remoção** (Desempilhar)

topo : -1 



→ Item a ser removido: ???

**PILHA VAZIA!!!**

Stack Underflow

Se a pilha não estiver vazia.

Portanto:

- Guardar o elemento que está em topo.
- Decrementar topo de uma unidade.
- Devolver o valor guardado

# Pilhas Estáticas

- Desempilhar (Lembrete)
  - Caso Geral
    - Se a pilha não estiver vazia, então
      - Guardar o item que está no topo
      - Decrementar o topo de uma unidade
      - Devolver o item guardado
  - Como saber que a Pilha está vazia
    - Se o topo for igual a -1, a pilha está vazia.

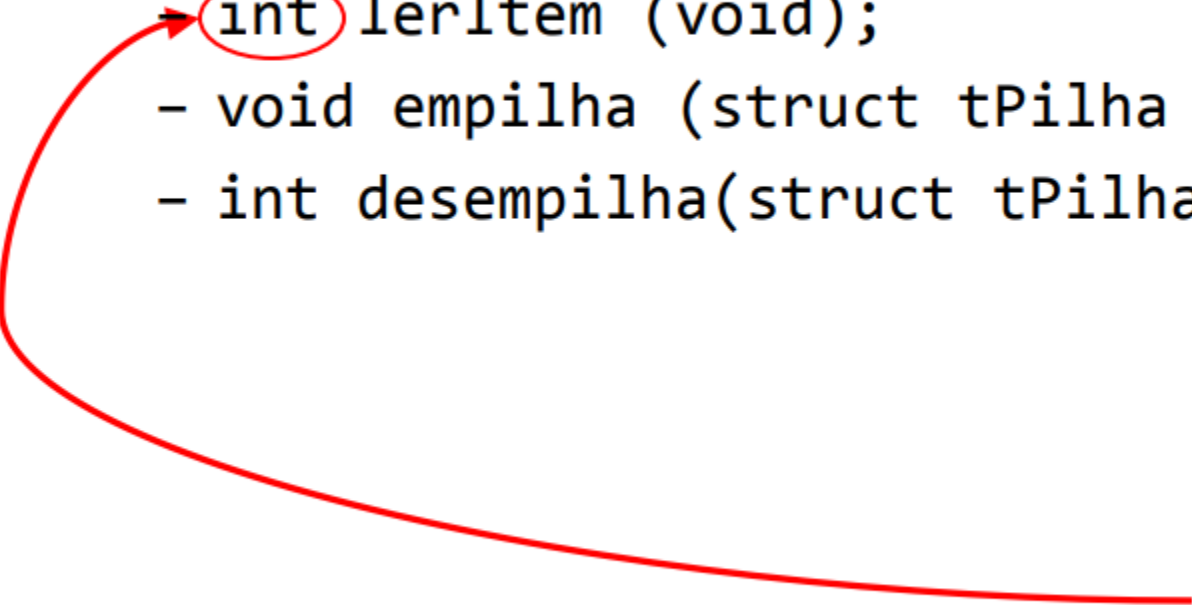
# Pilhas Estáticas

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

# Pilhas Estáticas

- Funções:

- void iniciaPilha (struct tPilha \*, int);
- int lerItem (void);
- void empilha (struct tPilha \*, int);
- int desempilha(struct tPilha \*);



Por simplificação do exemplo nossos itens serão inteiros. Em casos mais complexos teríamos estruturas nesses valores.

# Pilhas Estáticas

- Implementação da Estrutura Pilha

```
1. struct tPilha
2. {
3.     int *itens;
4.     int topo, tamanho;
5. };
```

Por simplificação os  
itens são um vetor de  
inteiros.

- Inicialização da Pilha

```
1. void iniciaPilha(struct tPilha *pilha, int n)
2. {
3.     pilha->itens = (int*) malloc(n*sizeof(int));
4.     pilha->topo = -1;
5.     pilha->tamanho = n;
6. }
```

# Pilhas Múltiplas

- Idéia:

