

Capítulo 8: Abstrações de dados

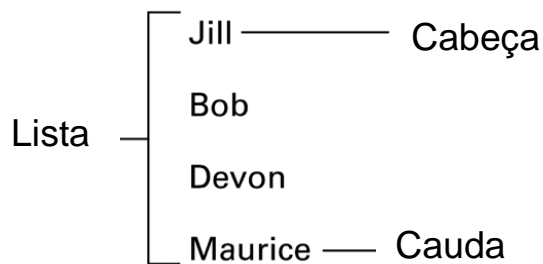
**Ciência da computação: Uma visão abrangente
11a Edition**

**by
J. Glenn Brookshear**

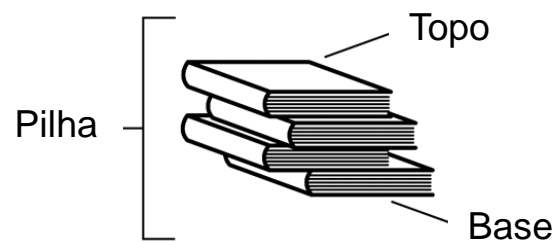
Estruturas de dados básicas

- Matriz homogênea
- Matriz heterogênea
- Lista
 - Pilha
 - Fila
- Árvore

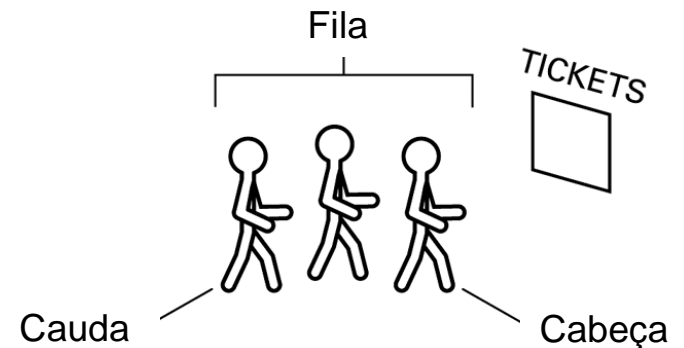
Listas, pilhas e filas



a. Uma lista de nomes



b. Uma Pilha de livros



c. Uma Fila de pessoas

Terminologia para listas

- **Lista:** Uma coleção de dados cujas entradas são arranjadas em seqüência
- **Cabeça:** O início da lista
- **Cauda:** O final da lista

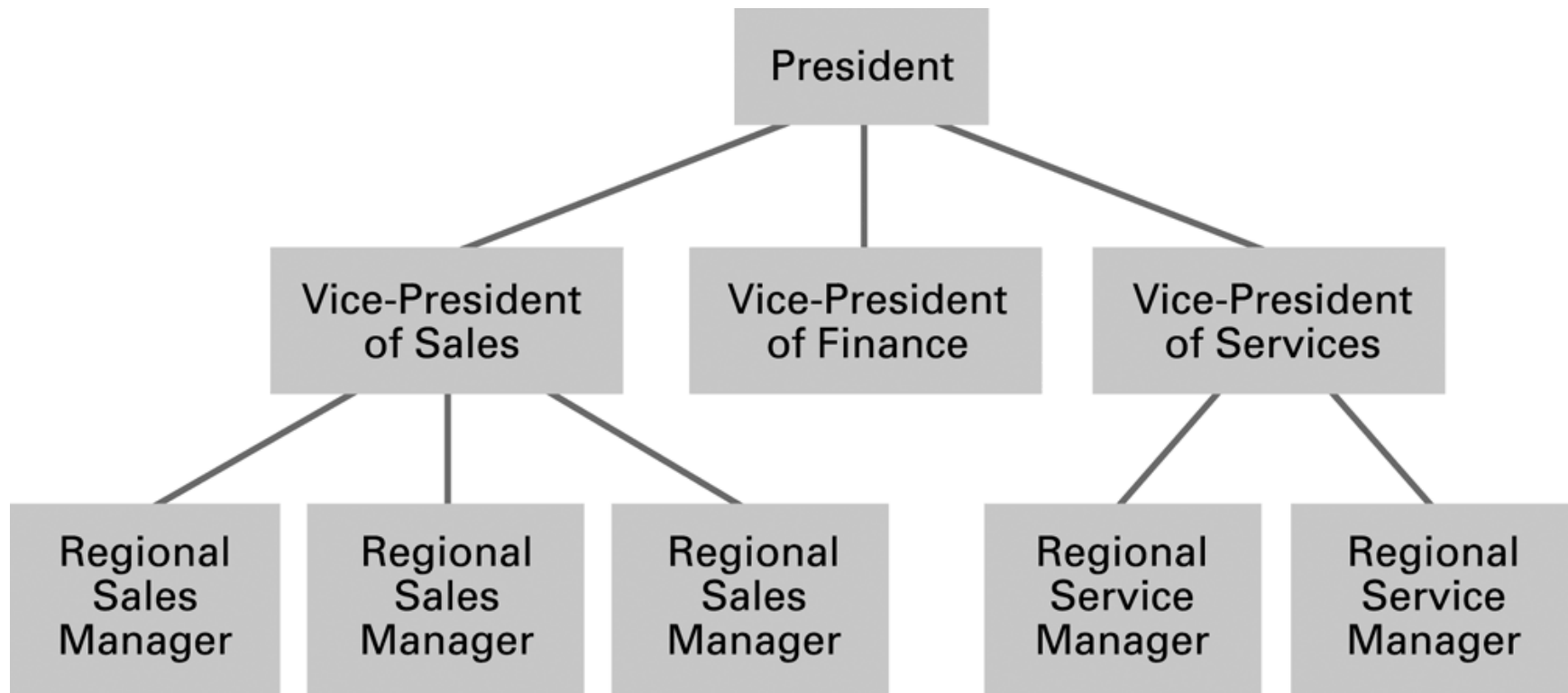
Terminologia para pilhas

- **Pilha:** Uma lista em que as entradas são removidas e inseridas apenas na cabeça
- **LIFO:** Last-in-first-out
- **Topo:** O cabeça de lista (pilha)
- **Bottom ou base:** A cauda da lista (pilha)
- **Pop:** Para remover a entrada na parte superior
- **Push:** Para inserir uma entrada no topo

Terminologia para filas

- **Queue:** A list in which entries are removed at the head and are inserted at the tail
- **FIFO:** First-in-first-out

Um exemplo de um organograma (árvore)



Terminologia para uma árvore

- **Árvore:** Uma coleção de dados cujas entradas têm uma organização hierárquica
- **Nó:** Uma entrada em uma árvore
- **Nó raiz (root):** O nó no topo
- **Terminal ou nó folha :** um nó na parte inferior da árvore

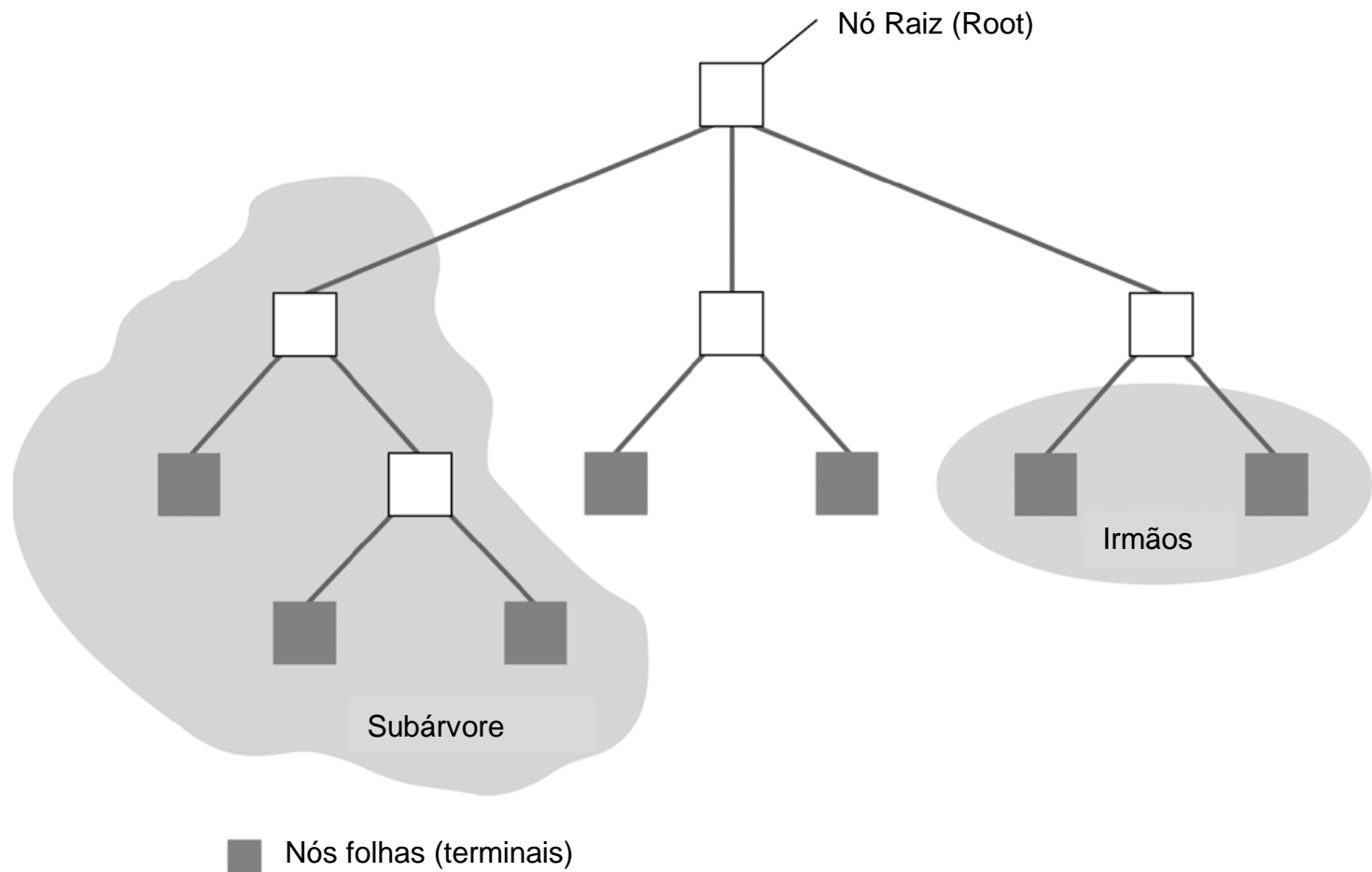
Terminologia para uma árvore (continuação)

- **Pai:** O nó imediatamente acima de um nó especificado
- **Criança:** Um nó imediatamente abaixo de um nó especificado
- **Ancestral:** Pai, pai de pai, etc..
- **Descendente:** Filho, filho de filho, etc.
- **Irmãos:** Nós compartilhando um pai comum

Terminologia para uma árvore (continuação)

- **Árvore binária:** uma árvore em que cada nó tem no máximo dois filhos.
- **Profundidade:** O número de nós no caminho mais longo da raiz até as folhas

Terminologia de árvore



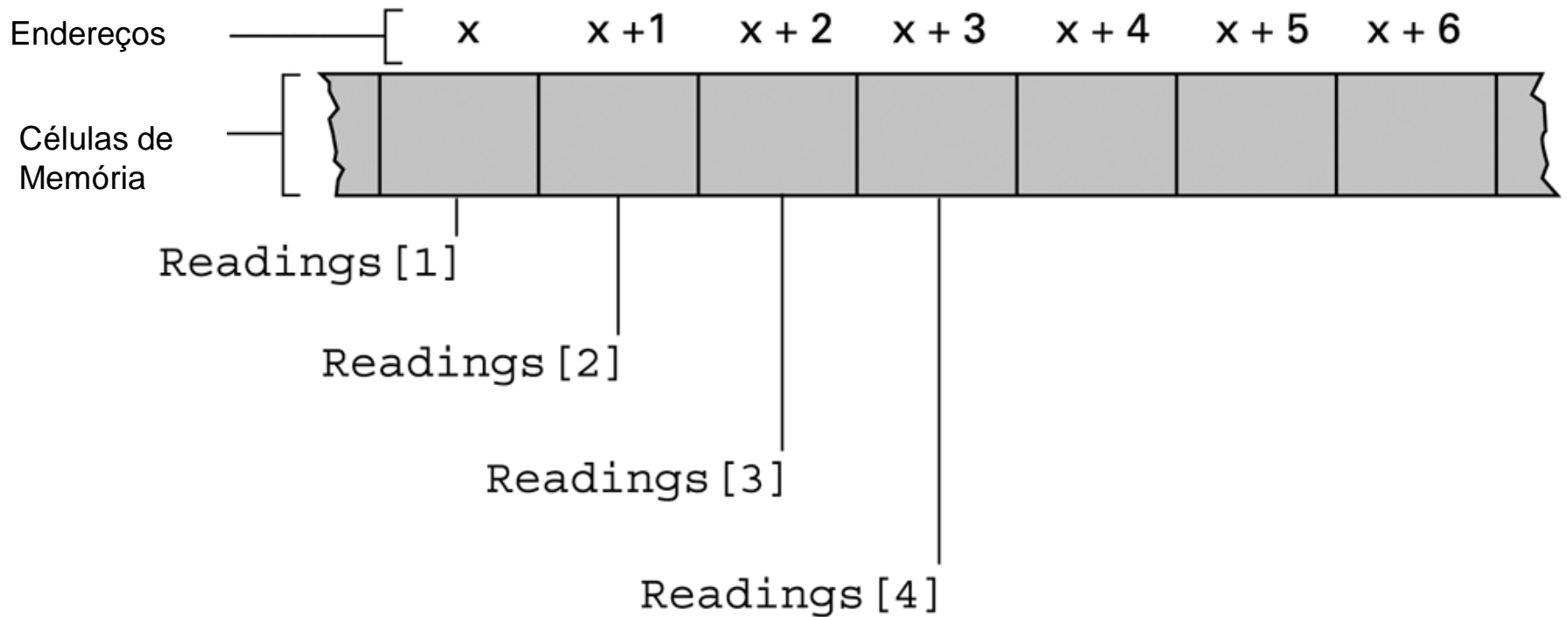
Conceitos adicionais

- Estruturas de dados estáticas: O Tamanho e forma da estrutura de dados não altera
- Estruturas de dados dinâmicas: O Tamanho e forma da estrutura de dados podem mudar
- Ponteiros: Usados para localizar dados

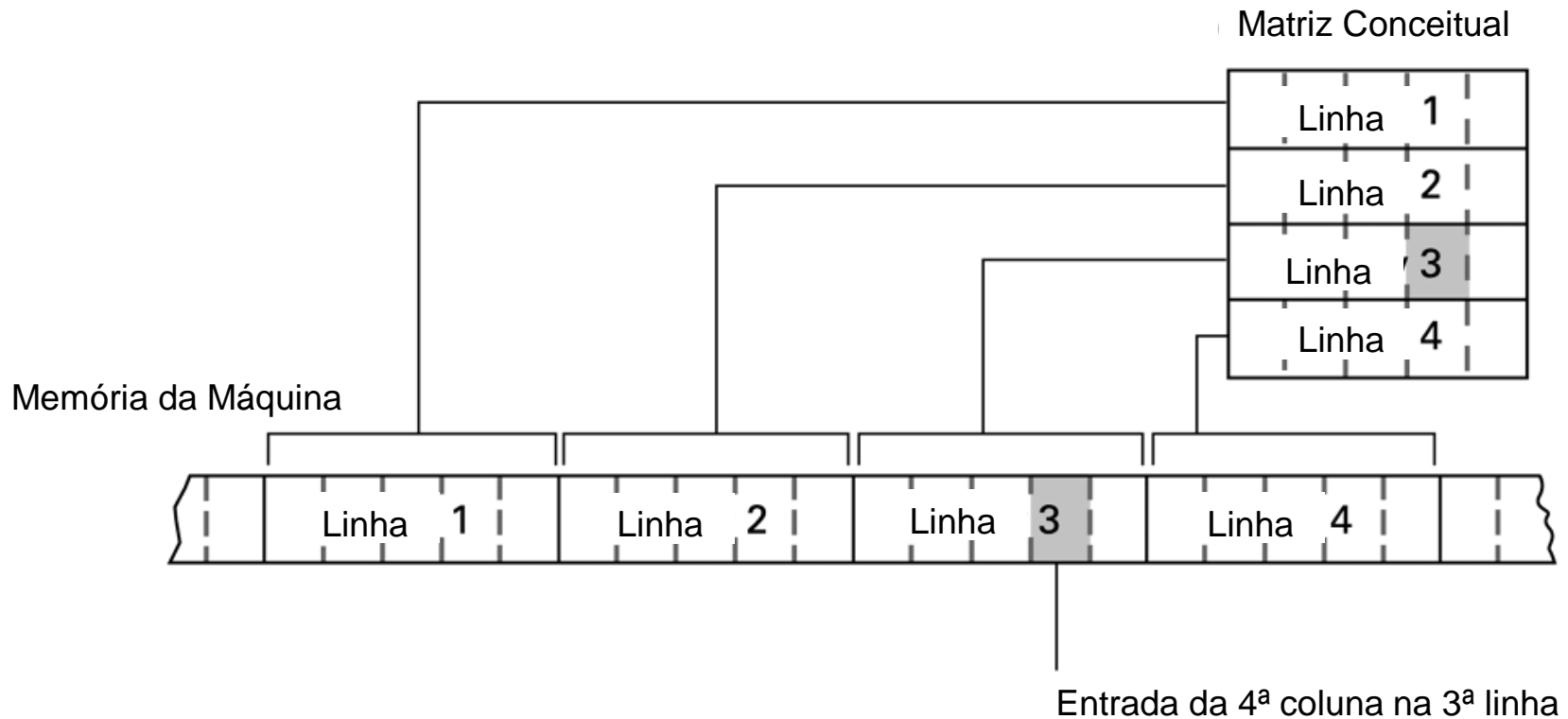
Armazenamento de Matrizes

- Matriz homogênea
 - Ordem principal de linhas x Ordem principal de colunas
 - Endereçamento Polinomial
- Matriz heterogênea
 - Componentes podem ser armazenados um após o outro em um bloco contíguo
 - Componentes podem ser armazenados em locais separados, identificados por ponteiros

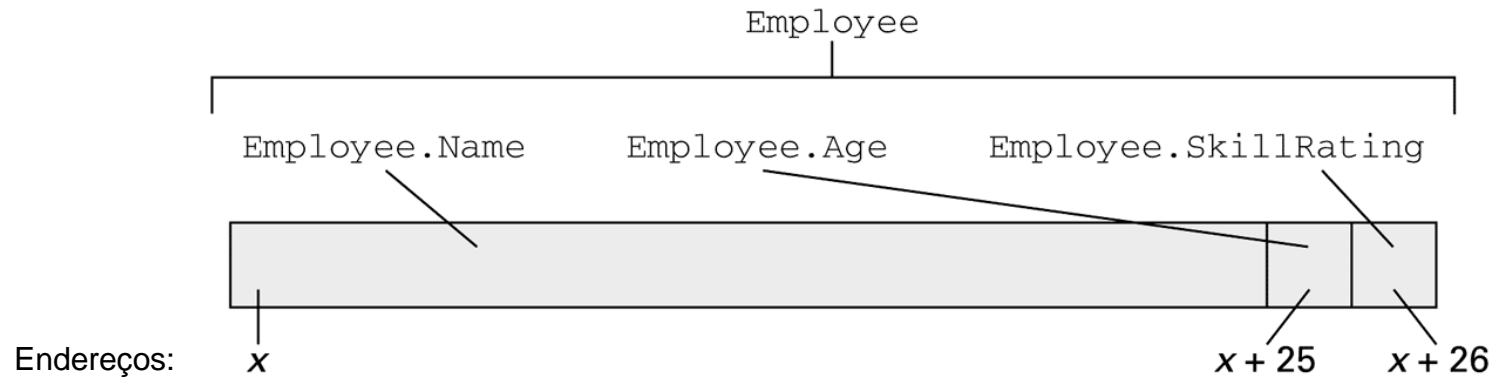
A matriz de leituras de temperatura armazenados na memória, iniciando no endereço x



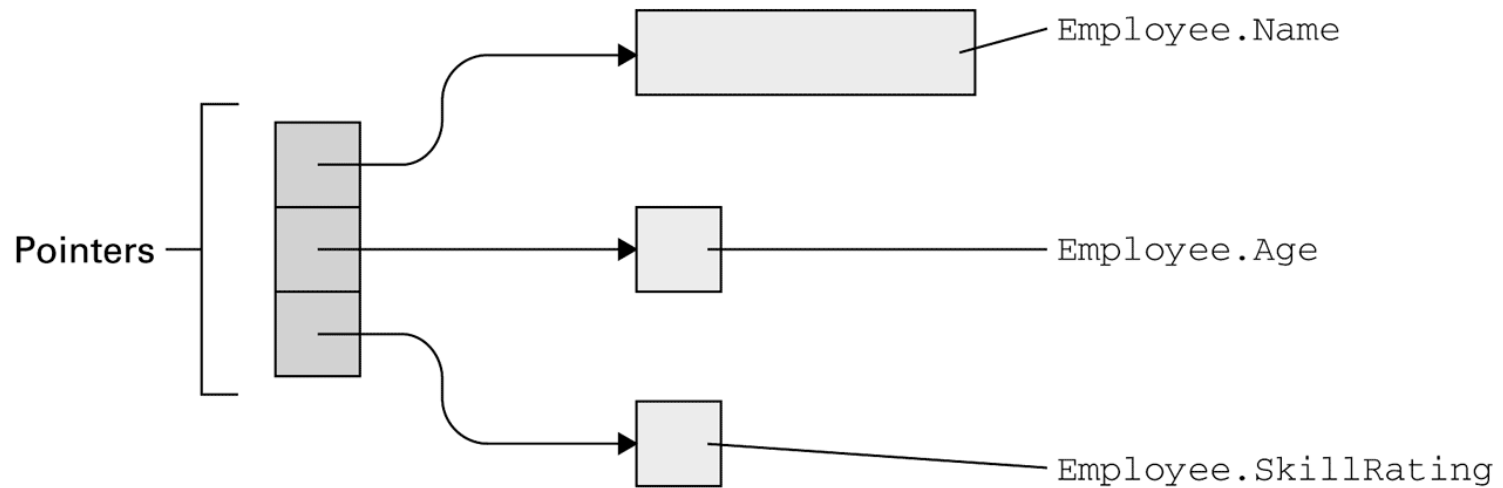
Uma matriz bidimensional com quatro linhas e cinco colunas armazenadas em ordem de linha principal



Armazenar a matriz heterogênea empregado



a. Matriz armazenada em um bloco contíguo

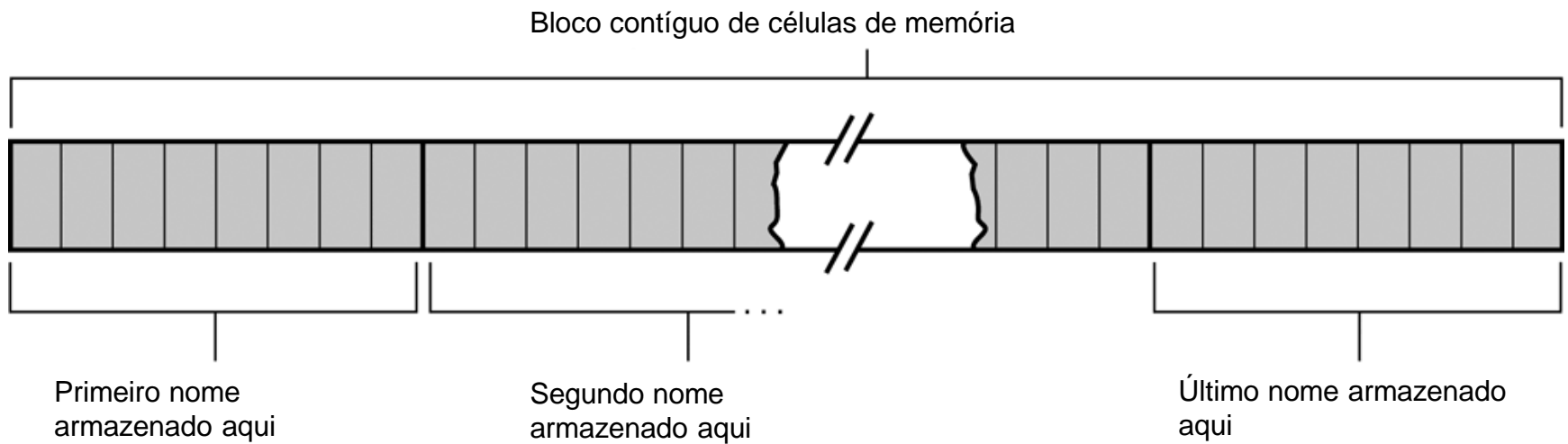


b. Componentes da Matriz armazenados em locais separados

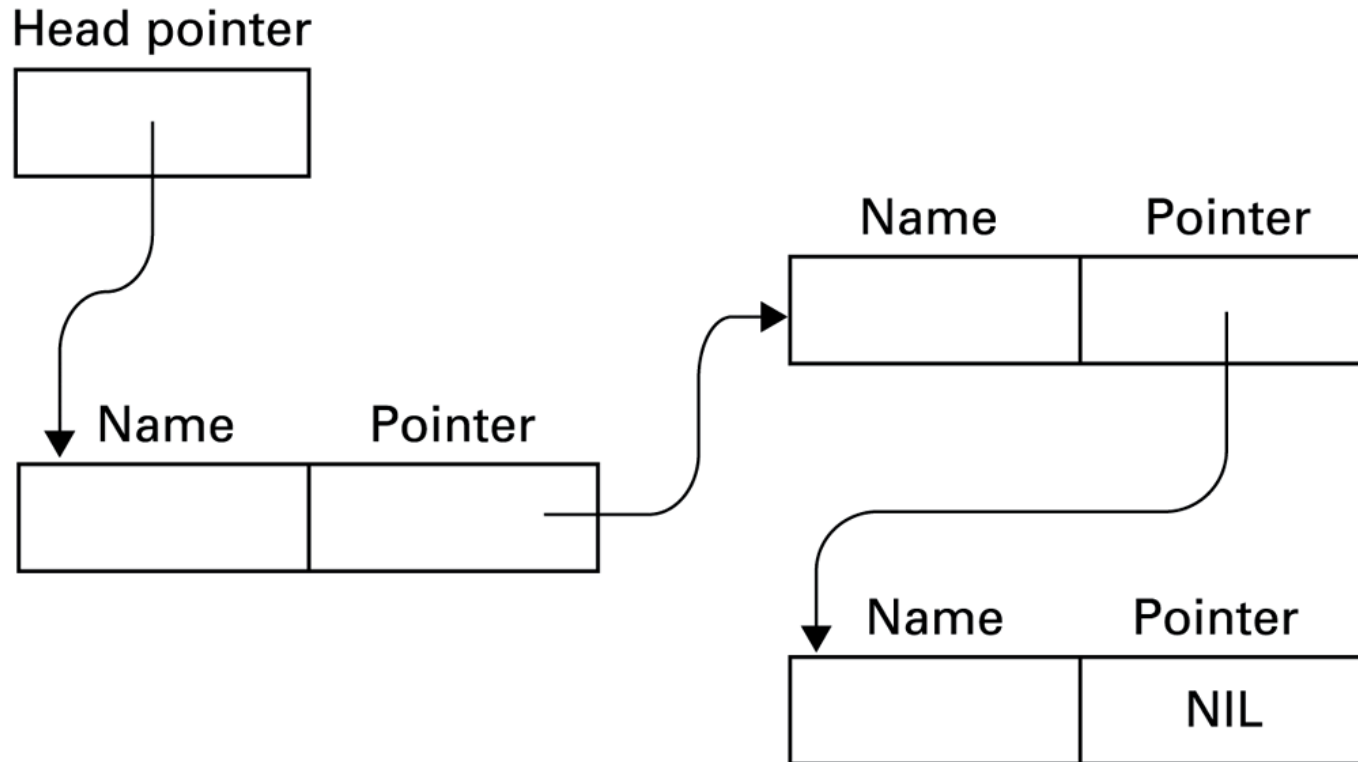
Armazenar listas

- **Lista contígua:** lista armazenada em uma matriz homogênea
- **Lista encadeada:** lista em que cada uma das entradas são ligadas através de ponteiros
 - **Head pointer:** Ponteiro para a primeira entrada na lista
 - **NIL (NULL) pointer:** Um valor padrão de "não-ponteiro" usado para indicar o final da lista

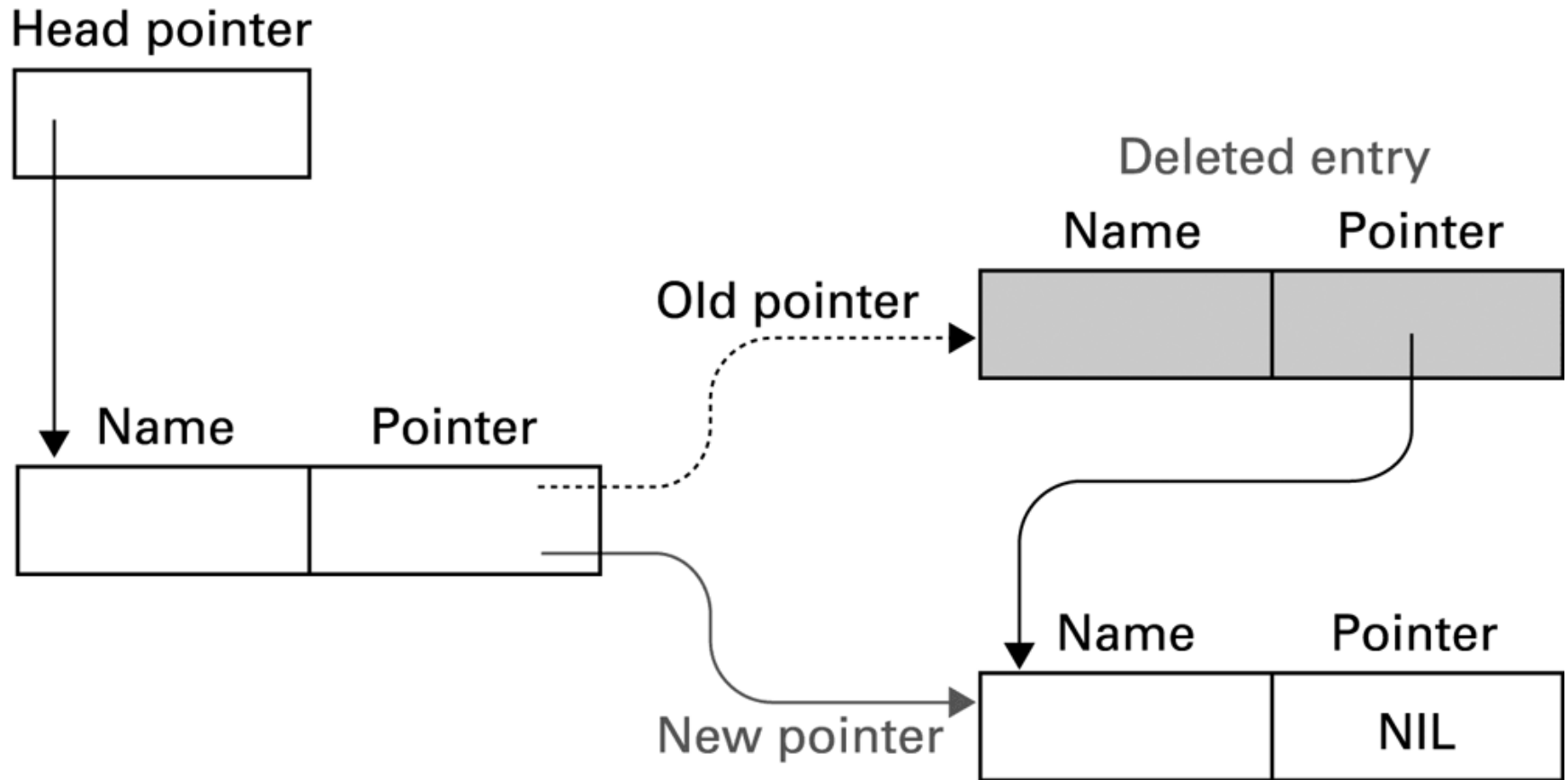
Nomes armazenados na memória como uma lista contígua



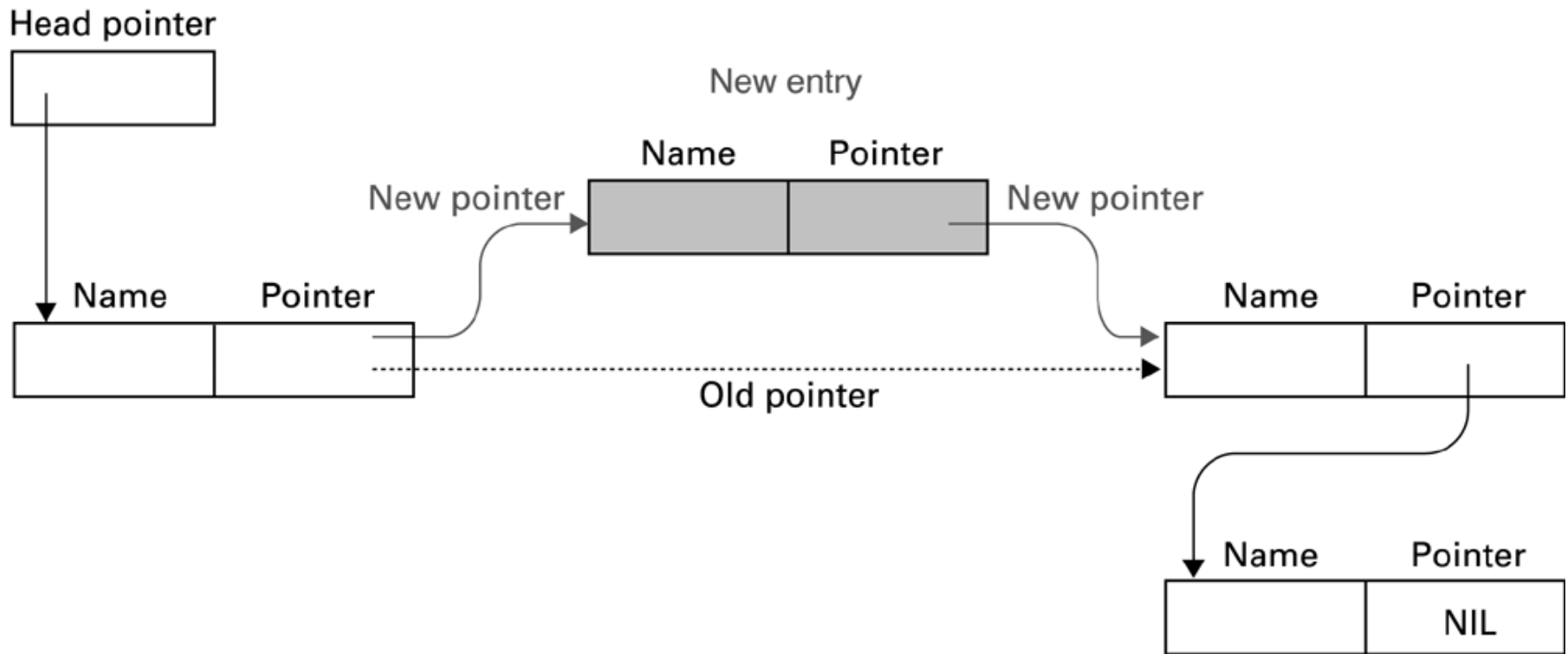
A estrutura de uma lista encadeada



Excluir uma entrada de uma lista encadeada



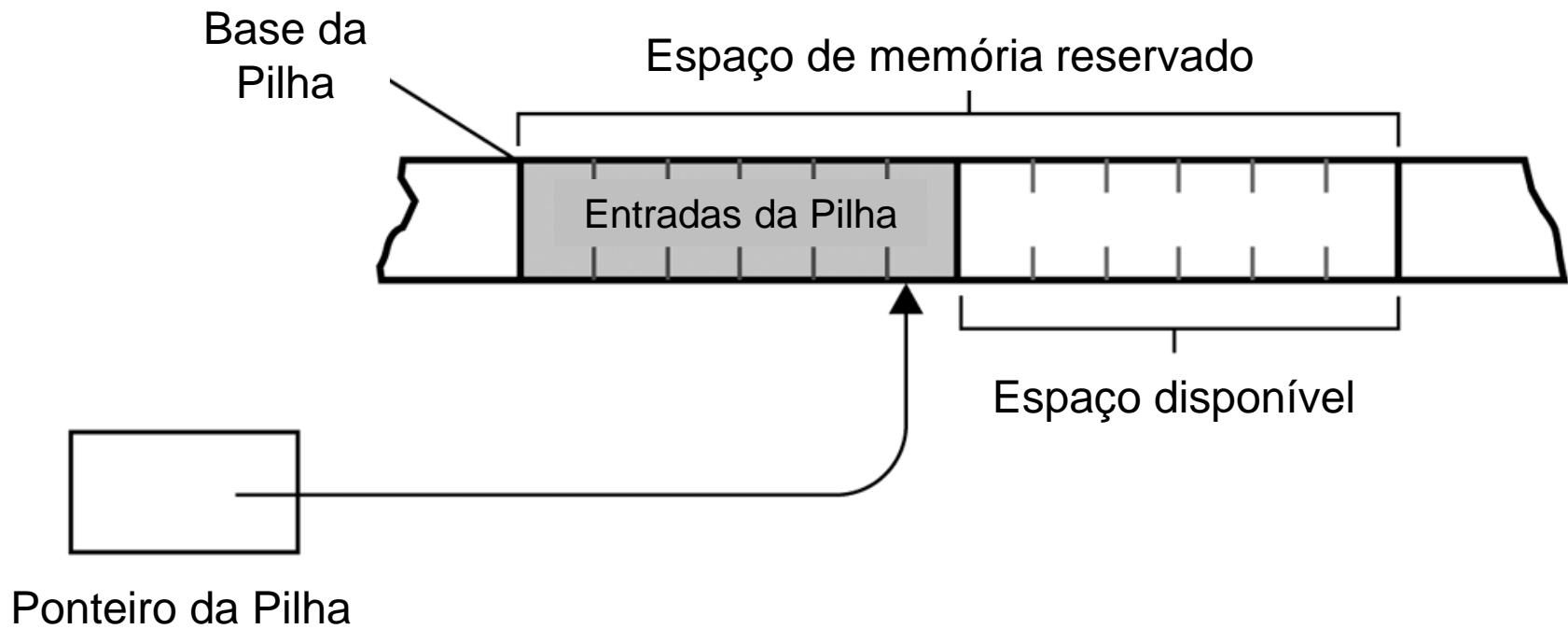
Inserindo uma entrada em uma lista vinculada



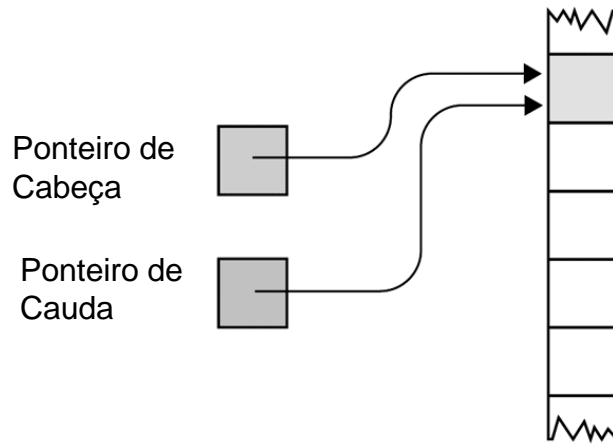
Armazenando Filas e Pilhas

- Pilhas normalmente são armazenadas como listas contíguas
- Filas geralmente são armazenadas como filas circulares
 - Armazenados em um bloco contíguo no qual a primeira entrada é considerada a seguir a última entrada
 - Evita que a fila estoure o espaço a ela alocado

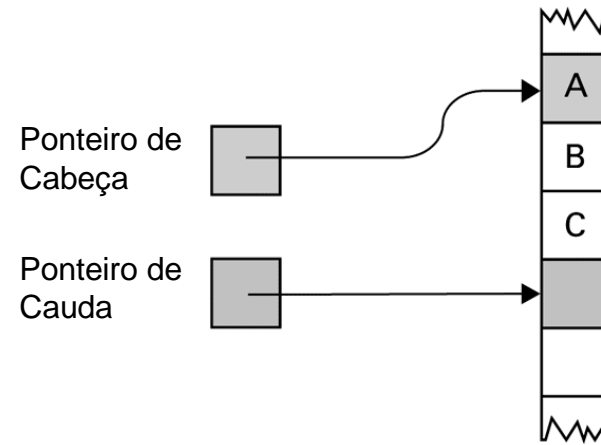
Uma pilha na memória



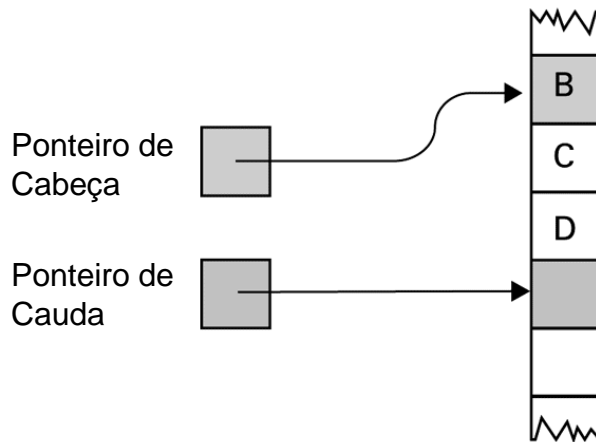
Uma implementação de fila com ponteiros de início (cabeça) e fim (cauda)



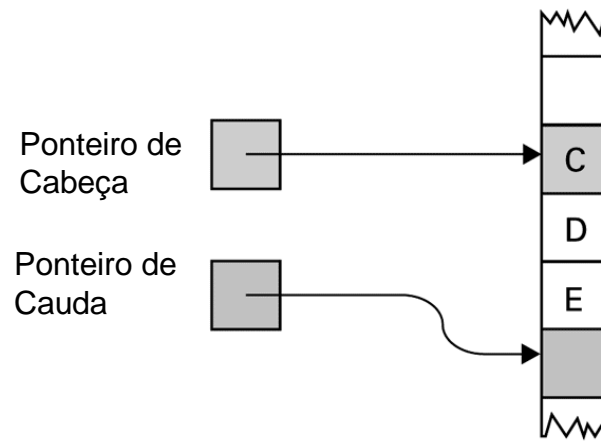
a. Fila Vazia



b. Após a inserção de A, B e C



c. Após remover A e inserir D

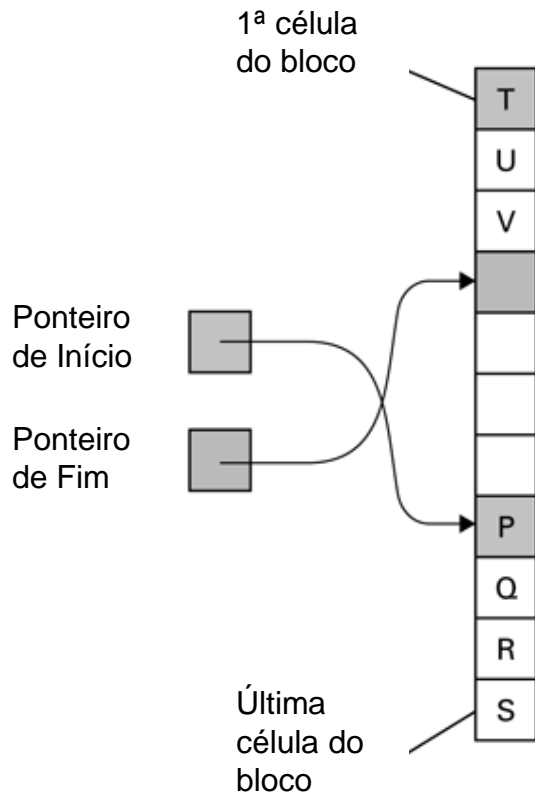


d. Após remover B e inserir E

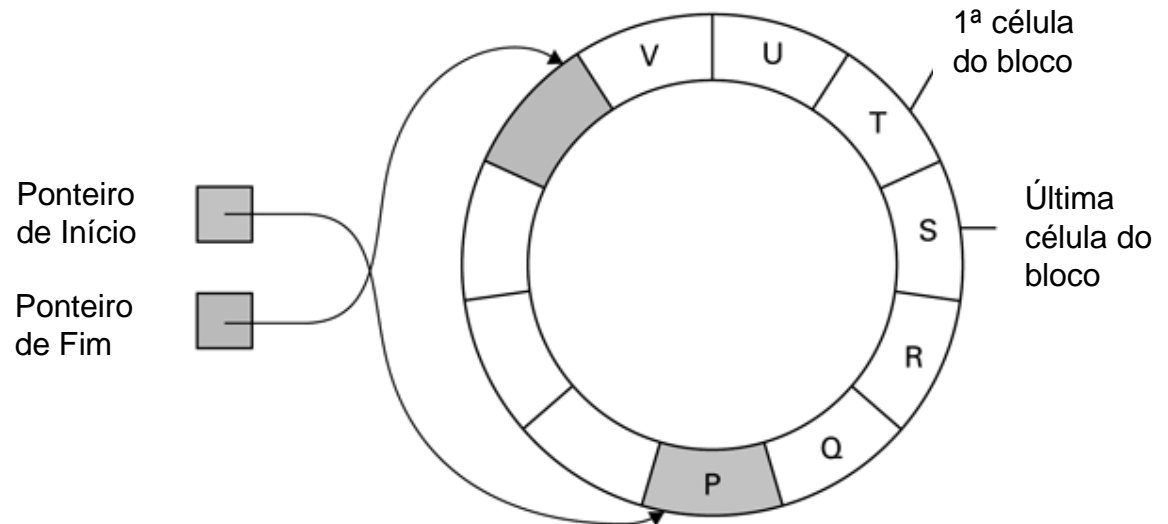
Armazenamento de árvores binárias

- Estrutura Encadeada
 - Cada nó = células de dados + dois ponteiros de filhos
 - Acessado através de um ponteiro para o nó raiz
- Estrutura de matriz contígua
 - $A[1]$ = nó raiz
 - $A[2], A[3]$ = filhos de $A[1]$
 - $A[4], A[5], A[6], A[7]$ = filhos de $A[2]$ e $A[3]$

Uma fila circular que contém as letras P a V

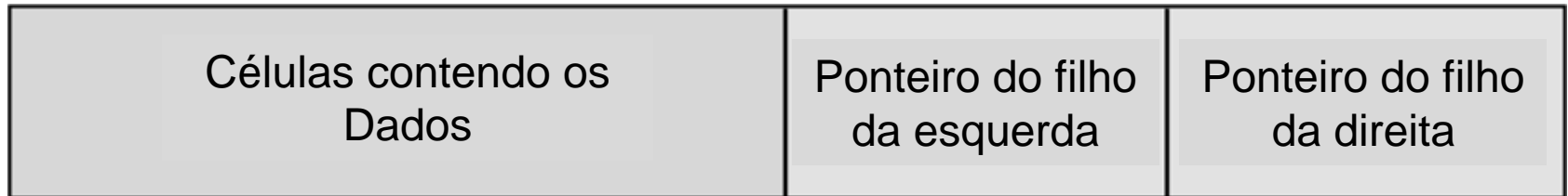


a. A fila como está realmente armazenada



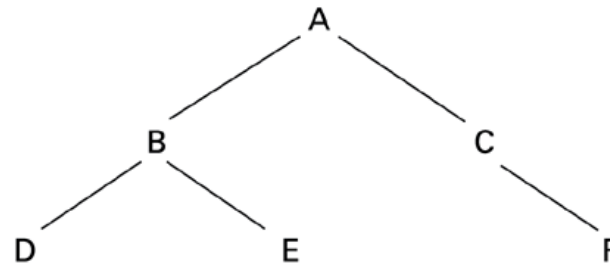
b. O armazenamento conceitual, com a última célula “adjacente” a primeira célula

A estrutura de um nó em uma árvore binária



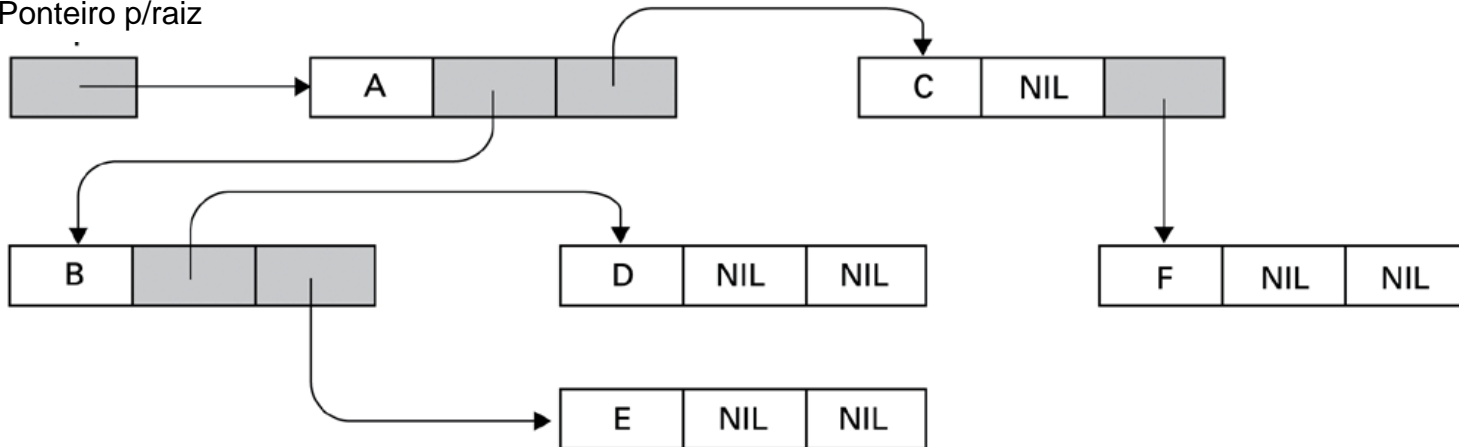
A organização conceitual e real de uma árvore binária, usando um sistema de armazenamento encadeado

Árvore Conceitual



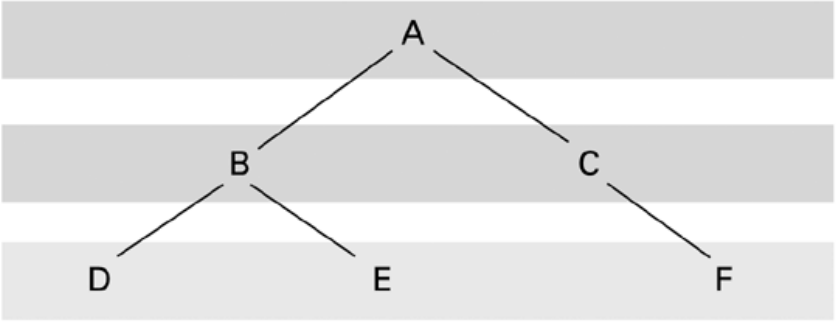
Organização de armazenamento real

Ponteiro p/raiz

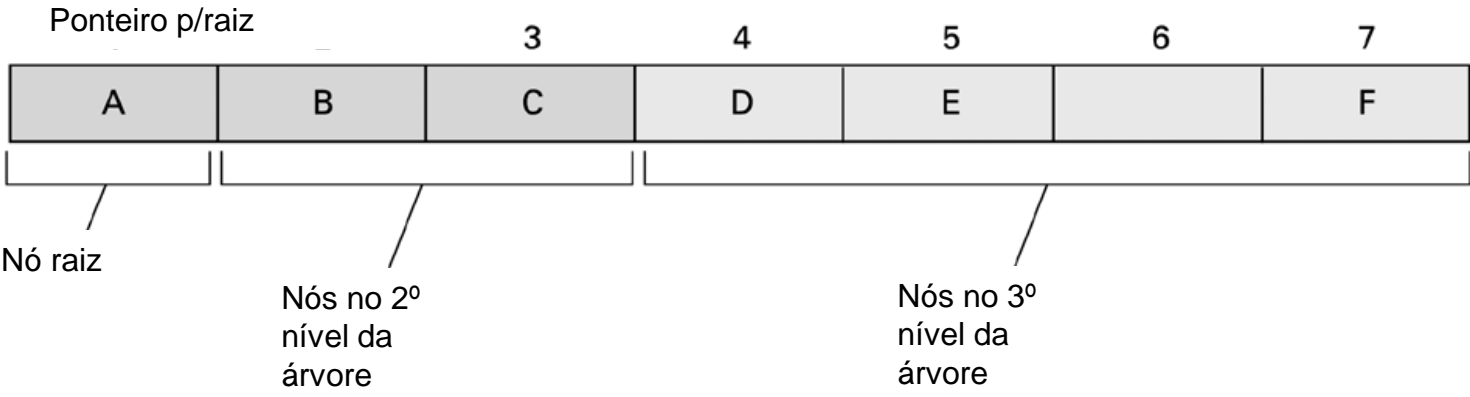


Uma árvore armazenada sem ponteiros

Árvore Conceitual

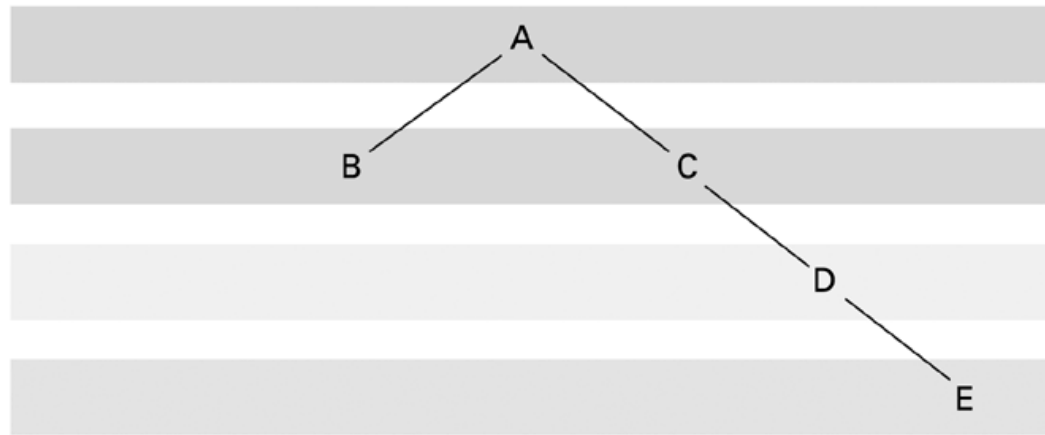


Organização de armazenamento real

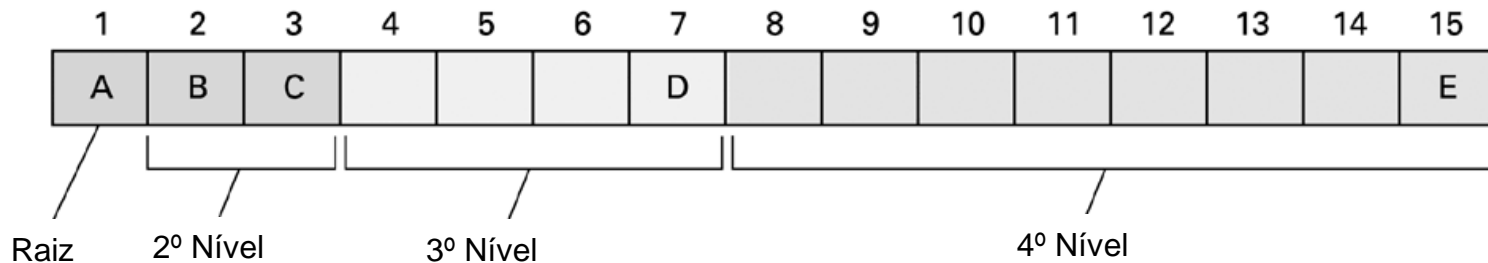


Uma árvore esparsa, desbalanceada, mostrada em sua forma conceitual e como ele seria armazenada sem ponteiros

Árvore Conceitual



Organização de armazenamento real



Manipulação de estruturas de dados

- Idealmente, uma estrutura de dados deve ser manipulada unicamente por procedimentos pré-definidos.
 - Exemplo: Uma pilha tipicamente precisa de pelo menos os procedimentos `push` e `pop`
 - A estrutura de dados juntamente com esses procedimentos constitui uma ferramenta abstrata completa.

Tipo de dados definido pelo usuário

- Um modelo para uma estrutura heterogênea
- Exemplo:

```
define type EmployeeType to be  
{char      Name [25];  
  int      Age;  
  real     SkillRating;  
}
```