

TAD Fila

Prof. D.Sc. Saulo Ribeiro

Funções do TAD Fila

- Fila inicFila(void) : cria e retorna uma fila vazia
- void adicFila(Fila f, TipoF elem): adiciona um elemento no final da fila
- void elimFila(Fila f): elimina e retorna o primeiro elemento da fila
- TipoF infoFila(Fila f): retorna o primeiro elemento da fila sem remover
- int vaziaFila(Fila f) : retorna verdadeiro se a fila está vazia e falso caso contrário
- void destruirFila(Fila f): destrói a fila f, desalocando toda memória ocupada

Conceitos de Fila

- Uma fila é uma estrutura linear, na qual os elementos só podem ser adicionados por um dos extremos e eliminados pelo outro extremo.
- O exemplo típico é de uma fila de espera que se faz enquanto se aguarda ser atendido.
- Fila:
 - um novo elemento é inserido no final da fila e um elemento é retirado do início da fila
 - fila = “o primeiro que entra é o primeiro que sai” (FIFO) First-In-First-Out
 - pilha = “o último que entra é o primeiro que sai” (LIFO) Last-In-First-Out

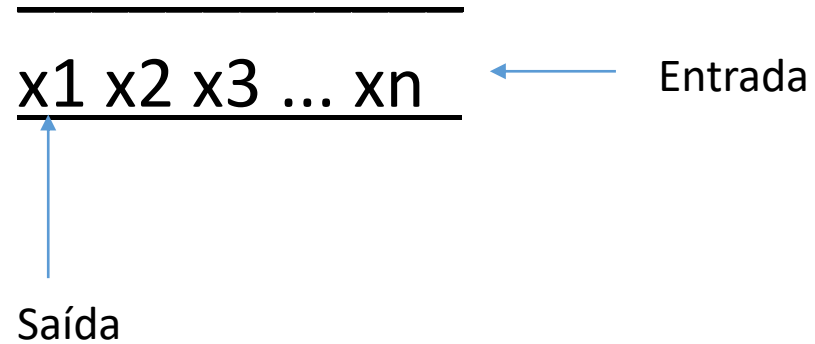
Conceitos de Fila

- As filas são muito utilizadas nos processos de simulação, nos quais se quer determinar o comportamento de um sistema que presta serviço a um conjunto de usuários, os quais esperam por sua vez para serem atendidos.
- Como exemplos destes sistemas, tem-se bancos, aeroportos, os processos de um computador.
- As filas também são usadas em muitos algoritmos de caminhamento em árvores e grafos.
- O único elemento visível de uma fila é o primeiro elemento. Enquanto ele não tiver sido eliminado, não se terá acesso ao elemento seguinte.

Notação



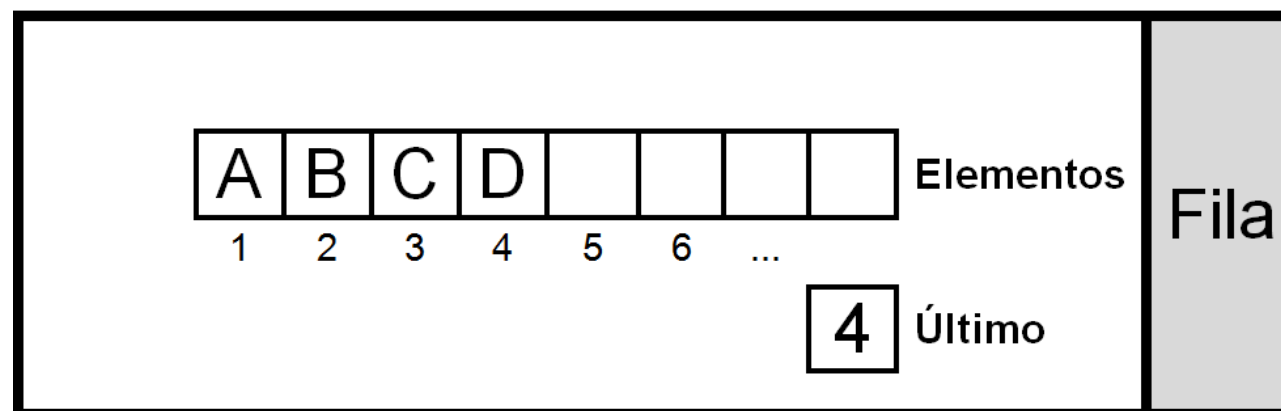
- O formalismo escolhido para fila:



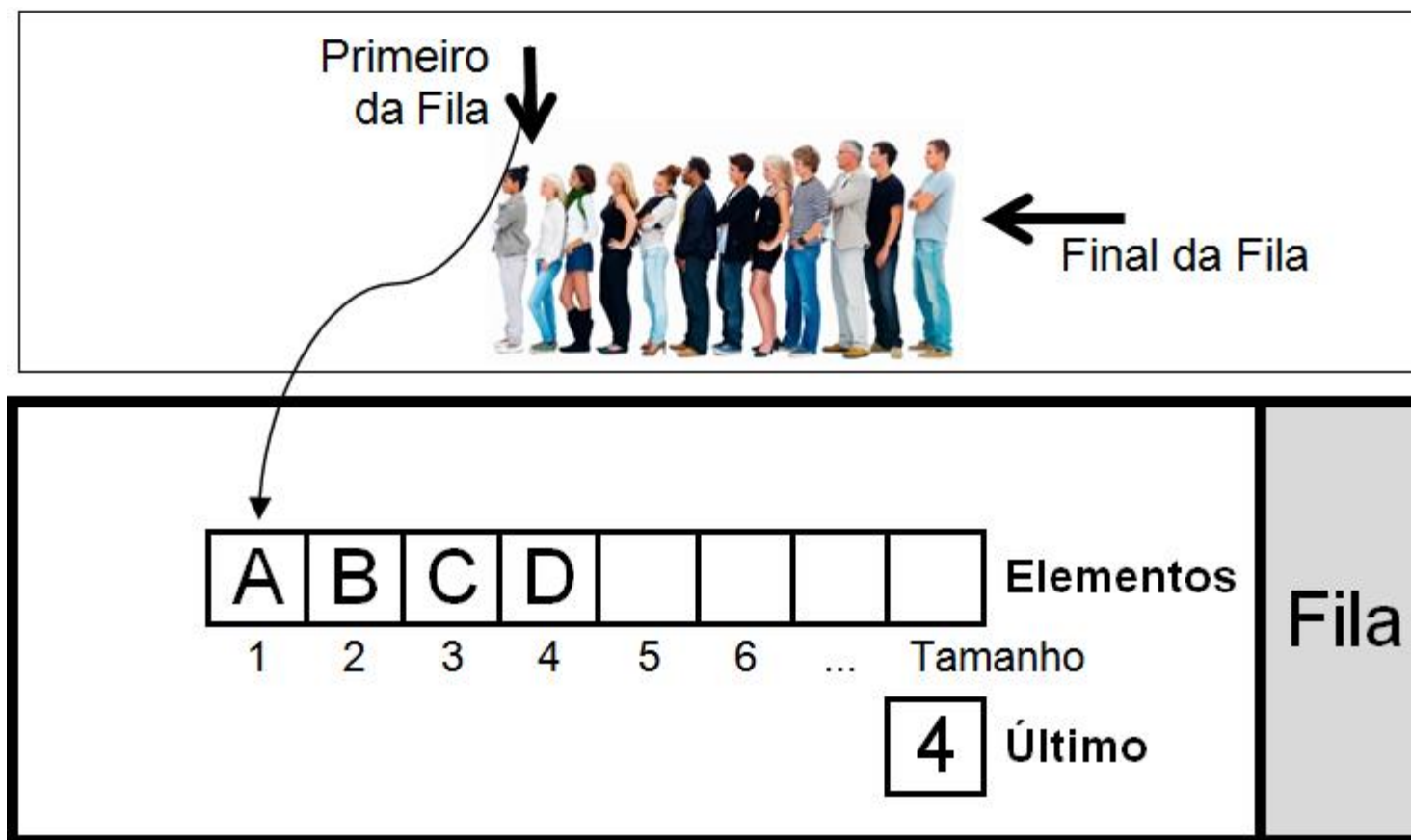
Implementação

- Vetor comum
- Vetor Circular
- Lista
 - Para inserir um elemento na fila, basta sempre por o iterador no último elemento e adicionar(anxLista). Para ler/eliminar o elemento da fila, basta por o iterador no primeiro elemento da lista e fazer as operações: infoLista,
 - Pode-se se usar uma estrutura de lista simplesmente encadeada, com um apontador para o primeiro elemento e outro para o último elemento.

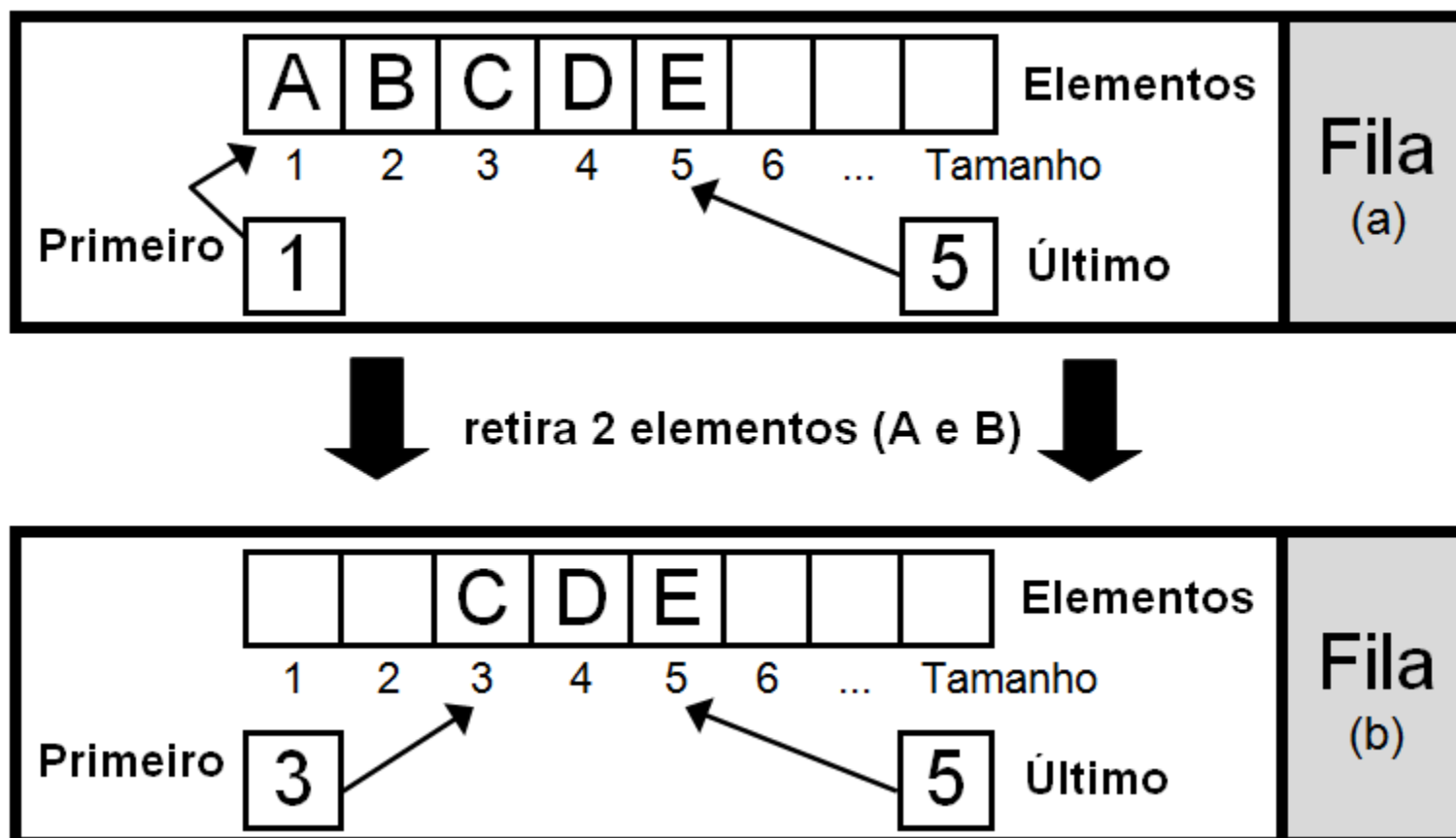
Implementação: Vetor Comum



Implementação: Vetor Comum



Implementação: Vetor Comum



Implementação: Vetor Circular

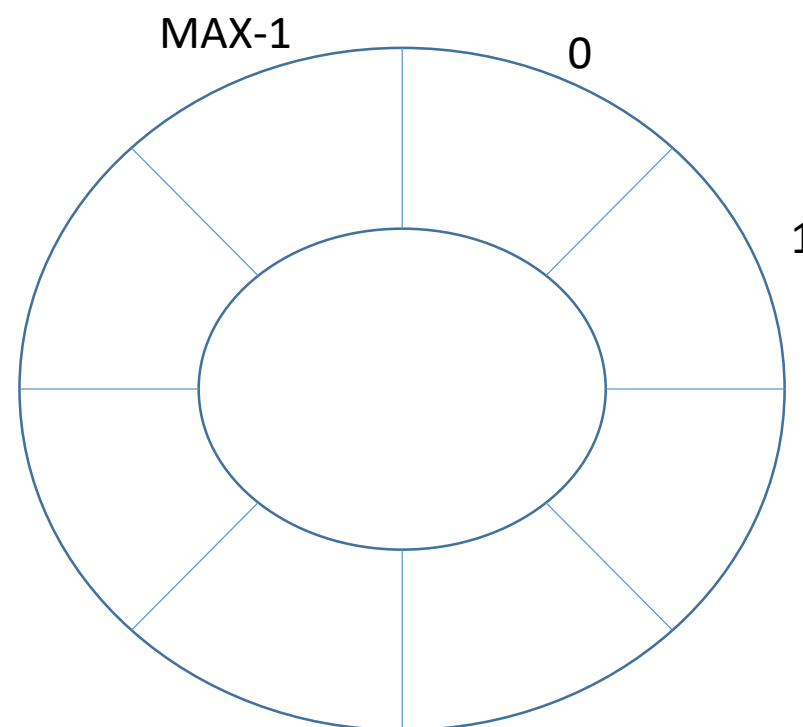
```
typedef int TipoF;
```

```
typedef struct{  
    TipoF info[MAX]; // vetor circular  
    int primeiro;     // posição do primeiro elemento  
    int ultimo;       // posição do último elemento  
} Tfila, *Fila
```

Implementação: Vetor Circular

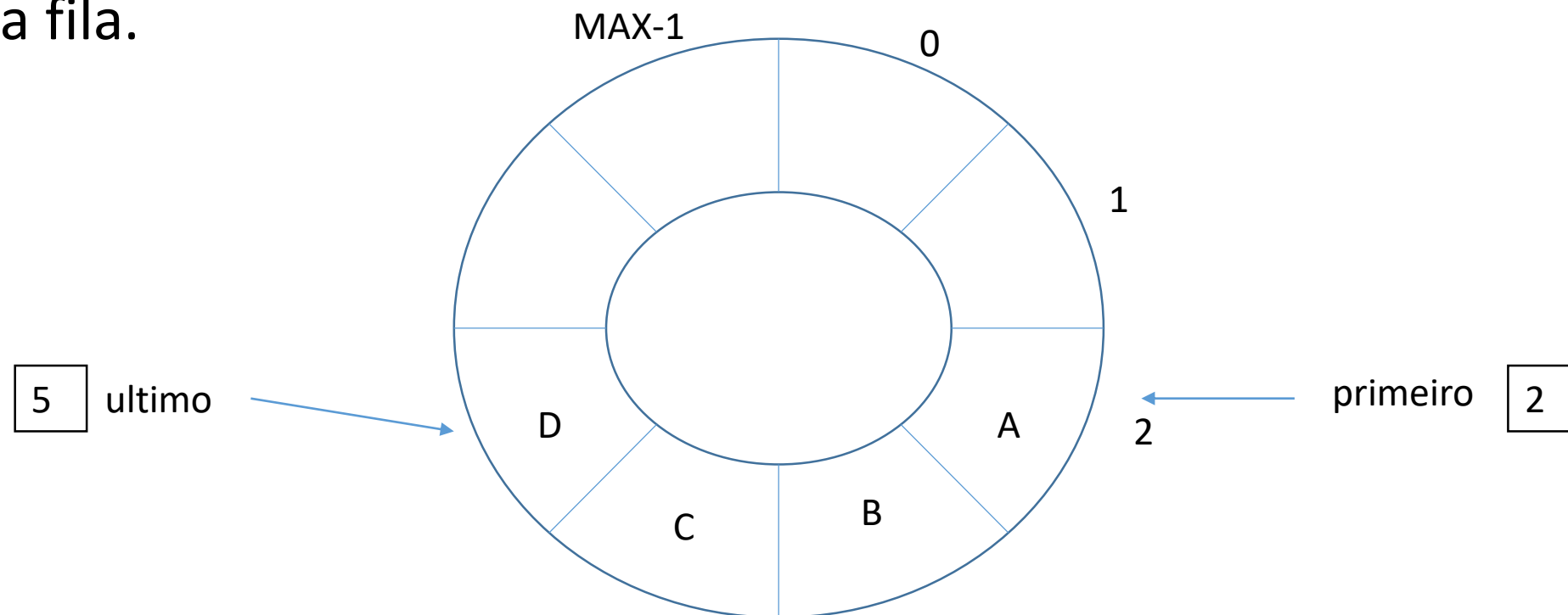
- A fila vazia se representa com valores especiais para os campos de primeiro e ultimo: por ex. pode-se setar -1.

primeiro
ultimo



Implementação: Vetor Circular

- Uma fila semi-cheia, tem o campo primeiro com a posição do primeiro elemento da fila, e o campo ultimo com a posição do último elemento da fila.



Implementação: Vetor Circular

- Uma fila completamente cheia sempre tem o campo primeiro uma posição depois do campo ultimo.
- Isto é, $((ultimo + 1) \% MAX) == primeiro$, onde % representa o operador módulo.

