

# Fila

SCC-202 – Algoritmos e Estruturas de  
Dados I

# Fila

- O que é?
- Para que serve?



# Biblioteca

---

- Quais são as operações possíveis?
  - Entrar na fila
    - Quem entra, entra onde?
  - Sair da fila
    - Quem sai, sai de onde?
  - Outras?

# Fila (*queue*)

---

- O que é?

- *É uma estrutura para armazenar um conjunto de elementos, que funciona da seguinte forma*

- Novos elementos sempre entram no fim da fila
    - O único elemento que se pode retirar da fila em um dado momento é seu primeiro elemento

- Para que serve?

- Modelar situações em que é preciso armazenar um conjunto ordenado de elementos, no qual o primeiro elemento a entrar no conjunto será também o primeiro elemento a sair do conjunto, e assim por diante

- F.I.F.O

- *First In, First Out*

# Aplicações de fila

---

- Biblioteca
  - Lista de espera para livros
- Impressão
  - Documentos a serem impressos
- Aeroporto
  - Lista de espera para vôos
- Outras?

# Operações

---

- $\text{Cria}(F)$ : cria uma fila  $F$  vazia
- $\text{Entra}(F, X)$ :  $X$  entra no fim da fila  $F$
- $\text{Sai}(F, X)$ : o primeiro elemento da fila  $F$  é retirado da fila e atribuído a  $X$
- $Y = \text{IsEmpty}(F)$ : verdade se a fila estiver vazia; caso contrário, falso
- $Y = \text{IsFull}(F)$ : verdade se a fila estiver cheia; caso contrário, falso

# Exemplo

<b>operação</b>	<b>fila</b>	<b>resultado</b>
cria( F )	1º DA FILA -->	
entra( F, a )	1º DA FILA --> a	
entra( F, b )	1º DA FILA --> a, b	
entra( F, c )	1º DA FILA --> a, b, c	
sai( F, X )	1º DA FILA --> b, c	X = a
entra( F, d )	1º DA FILA --> b, c, d	
sai( F, X )	1º DA FILA --> c, d	X = b

# Implementação da fila

---

## ■ Alocação seqüencial

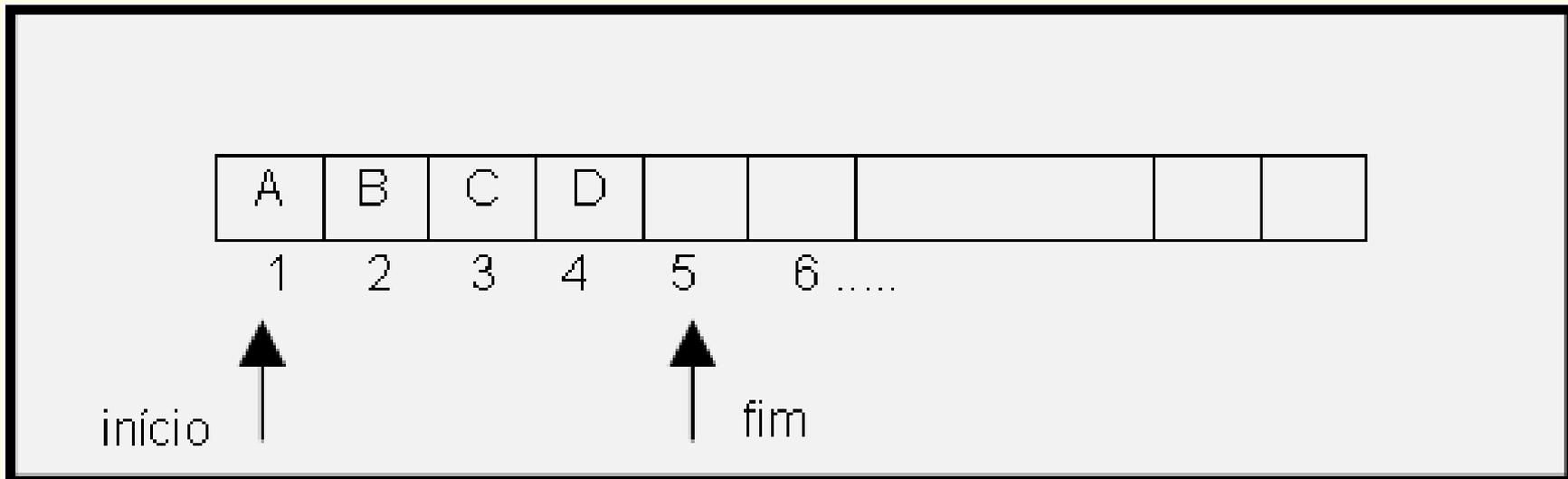
- Os elementos da fila ficam, necessariamente, em seqüência (um ao lado do outro) na memória

## ■ Alocação estática

- Todo o espaço de memória a ser utilizado pela fila é reservado (alocado) em tempo de compilação
- Todo o espaço reservado permanece reservado durante todo o tempo de execução do programa, independentemente de estar sendo efetivamente usado ou não

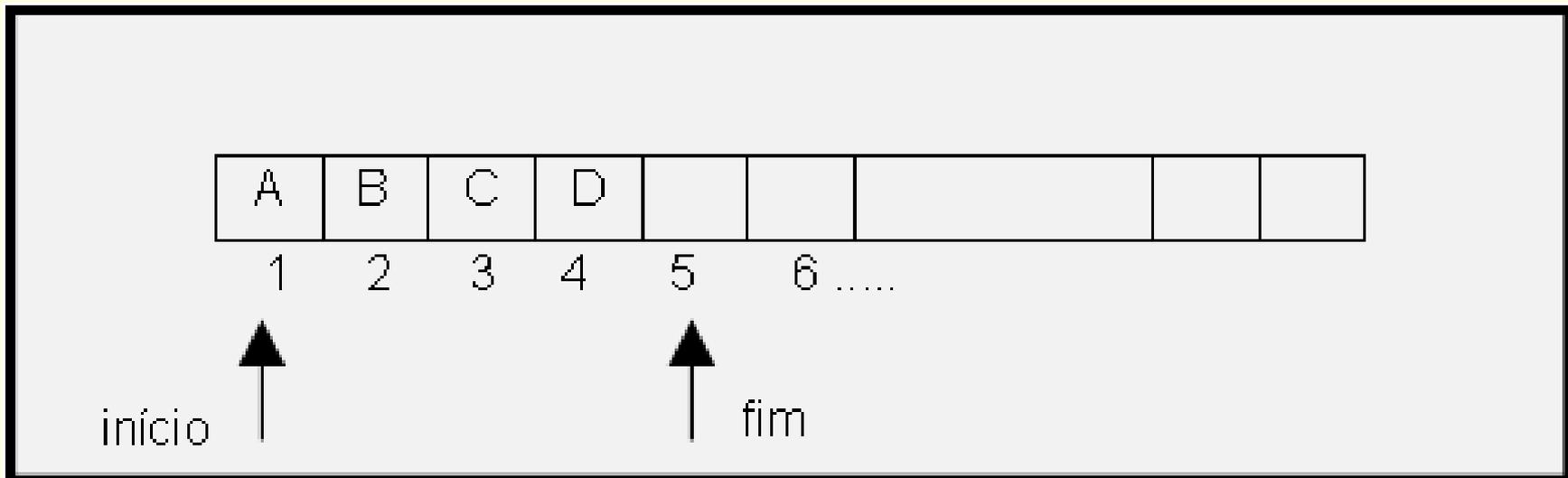
# Implementação da fila

- **Início** aponta para/indica o primeiro da fila, ou seja, o primeiro elemento a sair
- **Fim** aponta para/indica o fim da fila, ou seja, onde o próximo elemento entrará



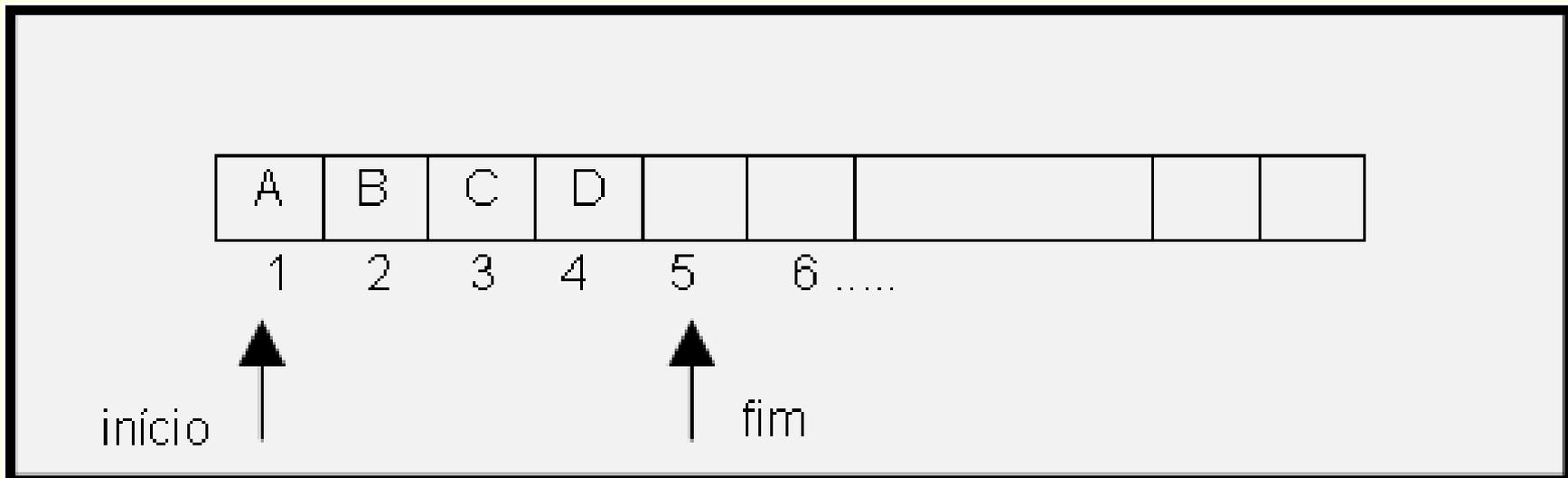
# Implementação da fila

- Qual a condição inicial, quando a fila é criada?
- Qual a condição para fila vazia?
- Qual a condição para fila cheia?



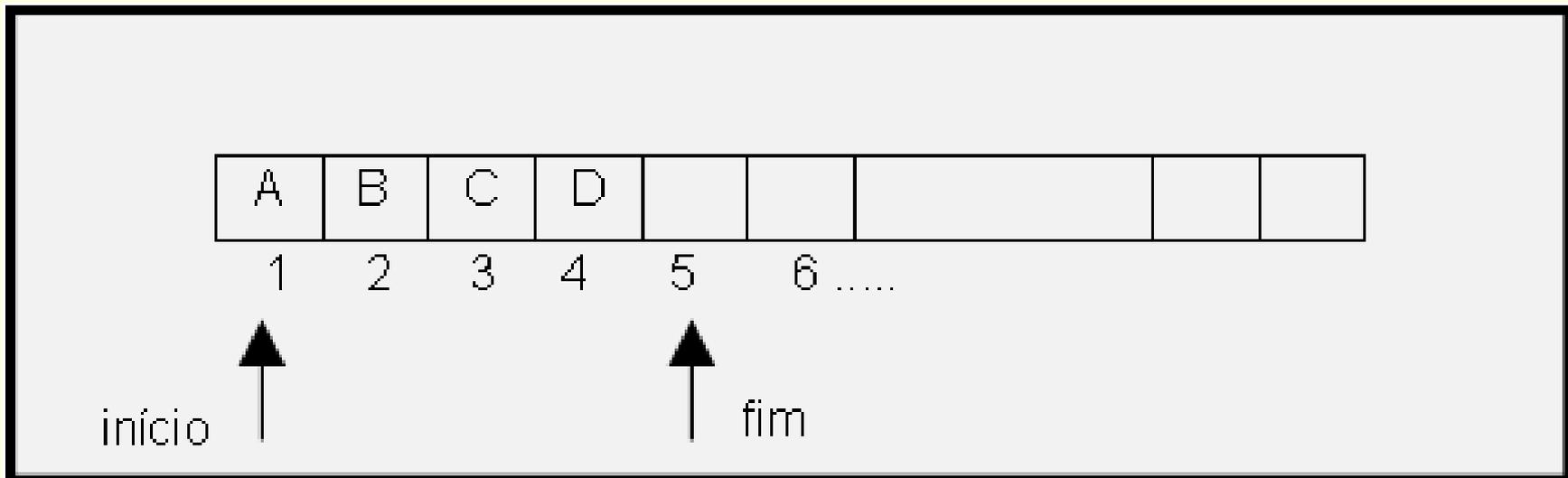
# Implementação da fila

- Qual a condição inicial, quando a fila é criada?
  - Início=0, fim=1
- Qual a condição para fila vazia?
- Qual a condição para fila cheia?



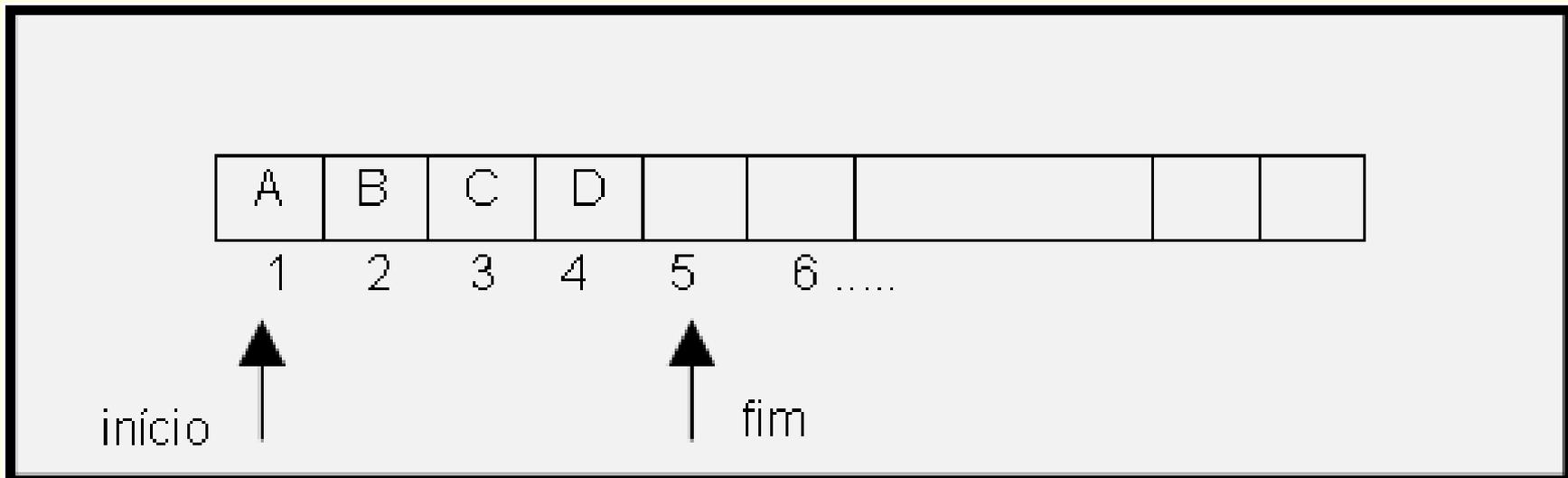
# Implementação da fila

- Qual a condição inicial, quando a fila é criada?
  - Início=0, fim=1
- Qual a condição para fila vazia?
  - Início=0, fim=1
- Qual a condição para fila cheia?



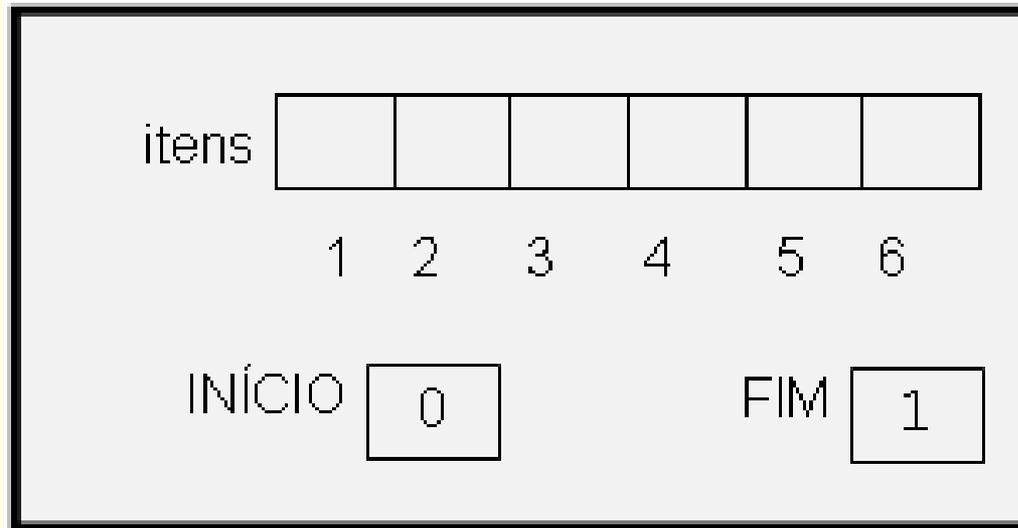
# Implementação da fila

- Qual a condição inicial, quando a fila é criada?
  - Início=0, fim=1
- Qual a condição para fila vazia?
  - Início=0, fim=1
- Qual a condição para fila cheia?
  - Fim=tamanho da pilha+1



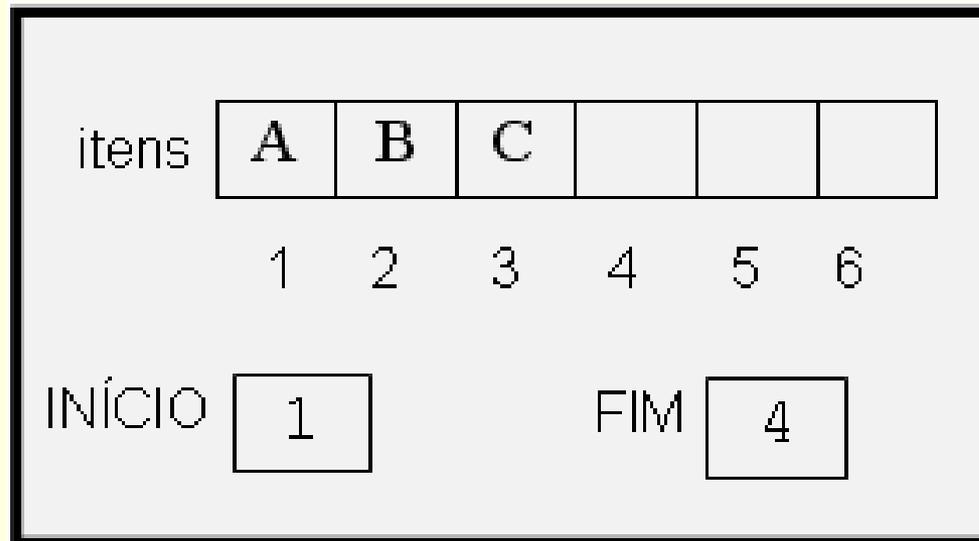
# Exemplo de uso da fila

- Criação da fila



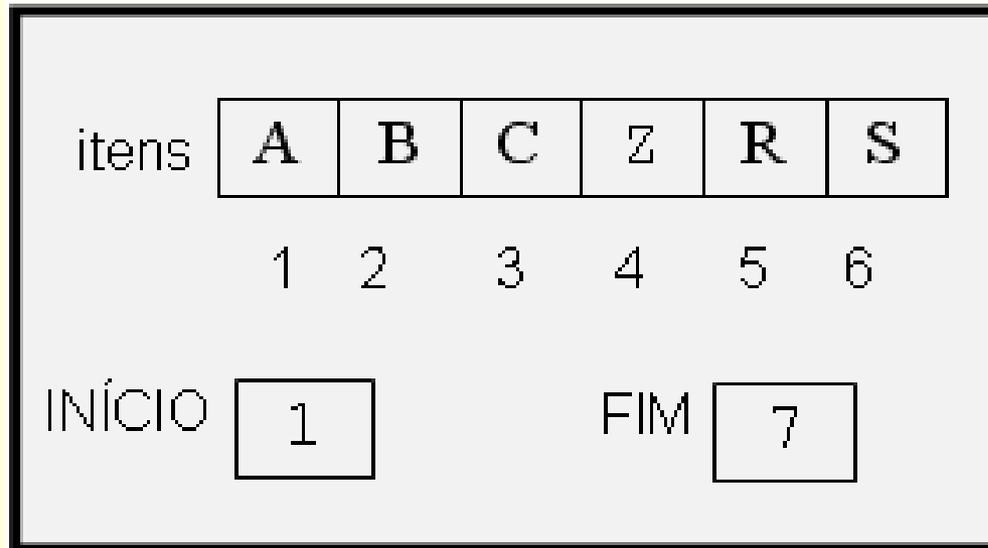
# Exemplo de uso da fila

- `entra(F,A)`, `entra(F,B)`, `entra(F,C)`



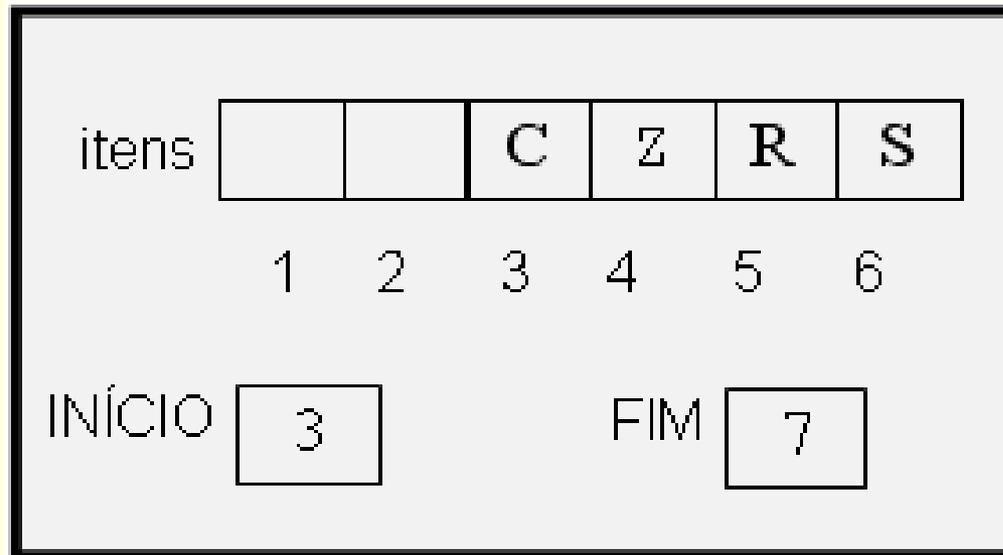
# Exemplo de uso da fila

- entra(F,Z), entra(F,R), entra(F,S)
  - IsFull=TRUE



# Exemplo de uso da fila

- `sai(F,X), sai(F,X)`
  - `IsFull=FALSE`



Como inserir mais elementos?

Qual o problema com a fila?

# Fila

---

- Como reutilizar os espaços do início da fila?

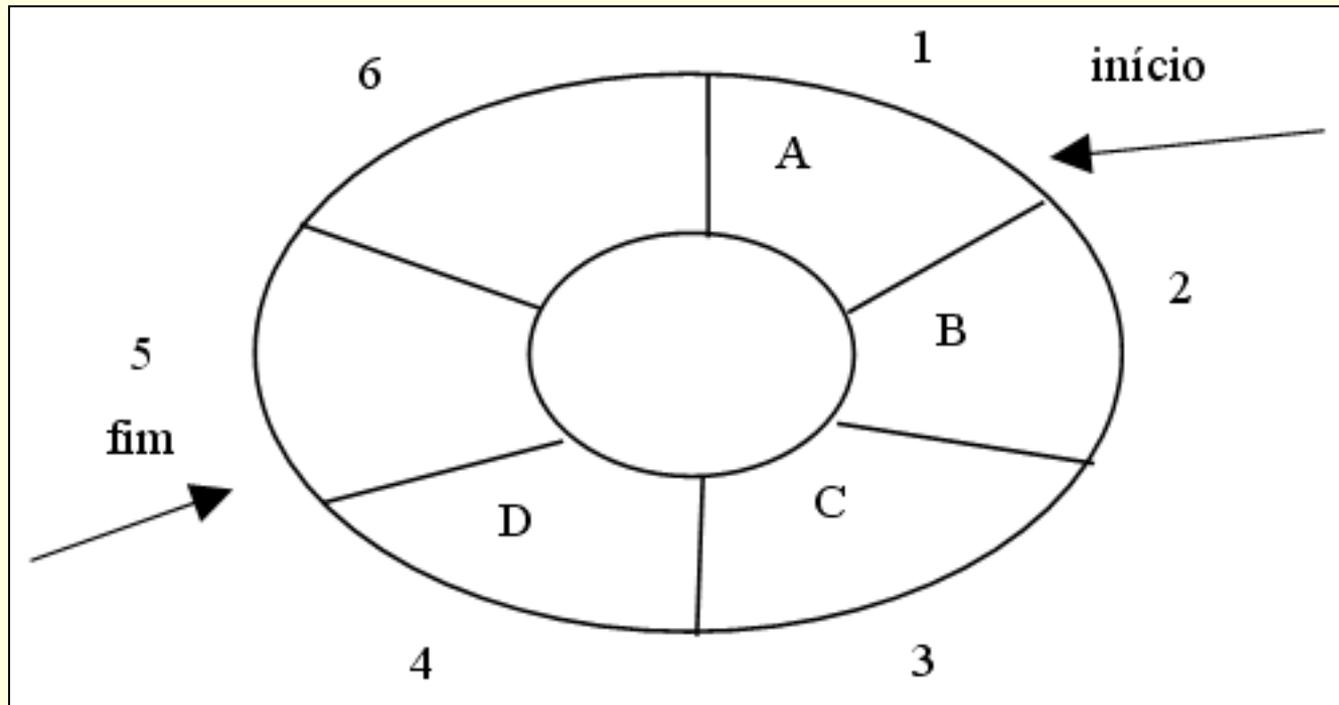
# Fila

---

- Como reutilizar os espaços do início da fila?
  - Outra forma de implementação
  - Melhor aproveitamento da representação utilizada

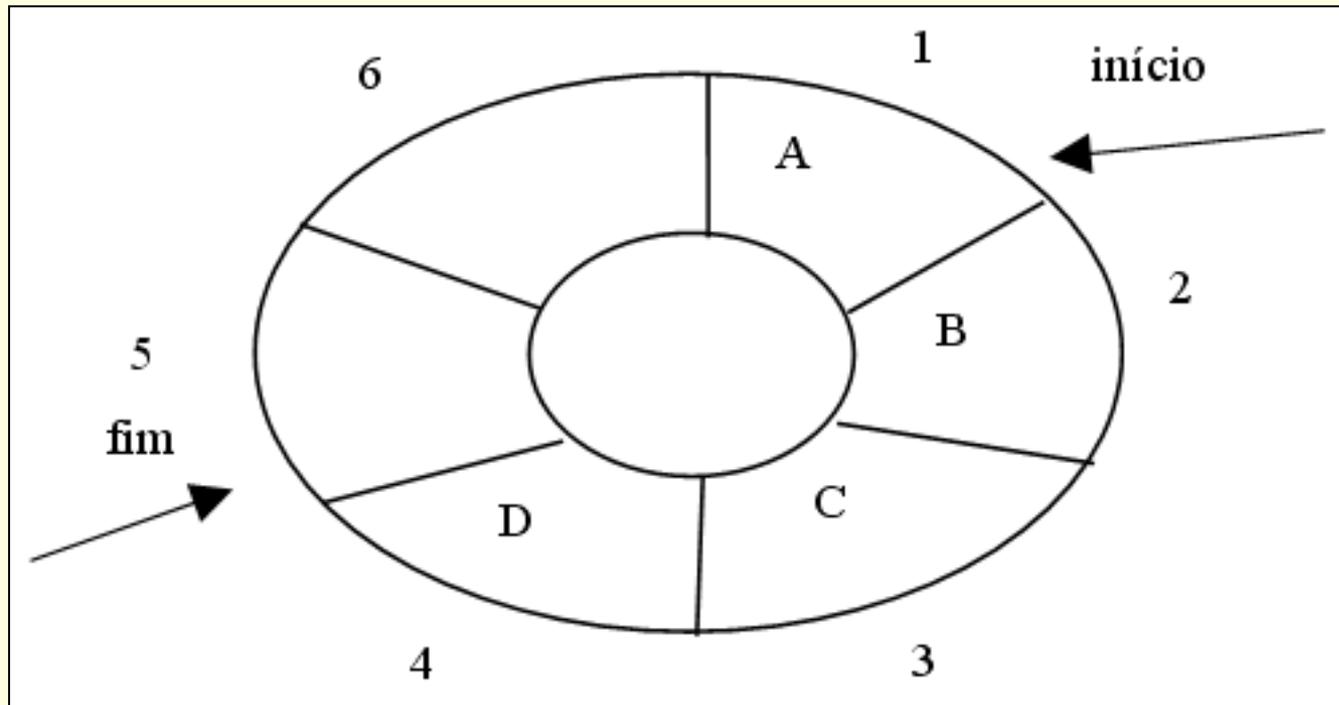
Fila em vetor circular!

# Fila em vetor circular



- Qual a condição para fila vazia?
- Qual a condição para fila cheia?
- Qual a condição inicial (quando a fila é criada)?

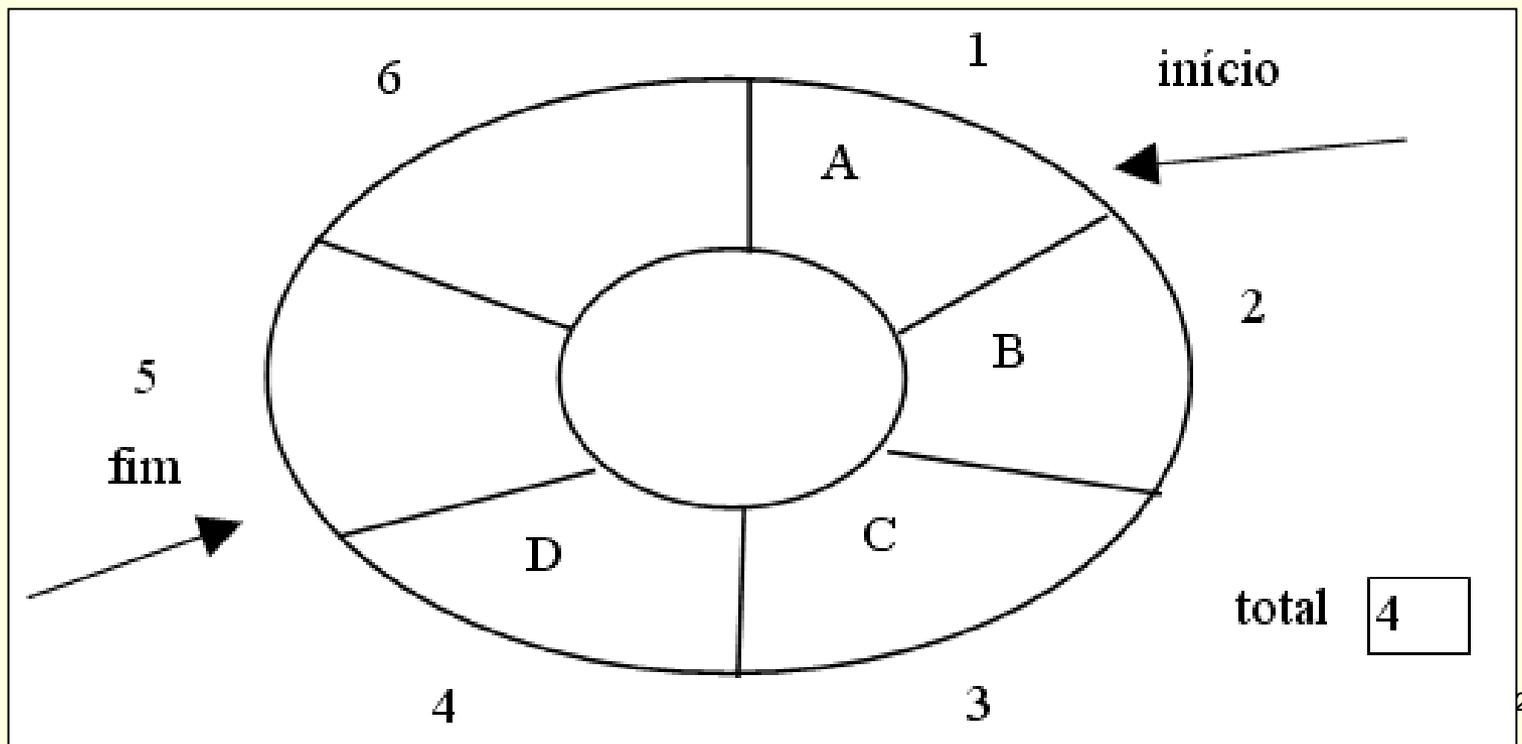
# Fila em vetor circular



- Qual a condição para fila vazia?
- Qual a condição para fila cheia?
- Qual a condição inicial (quando a fila é criada)?

# Fila em vetor circular

- **Solução:** campo extra para guardar número de elementos



# Fila em vetor circular

---

- Qual a condição para fila vazia?
- Qual a condição para fila cheia?
- Qual a condição inicial (quando a fila é criada)?

# Fila em vetor circular

---

- Qual a condição para fila vazia?
  - Total=0
- Qual a condição para fila cheia?
- Qual a condição inicial (quando a fila é criada)?

# Fila em vetor circular

---

- Qual a condição para fila vazia?
  - Total=0
- Qual a condição para fila cheia?
  - Total=tamanho da fila
- Qual a condição inicial (quando a fila é criada)?

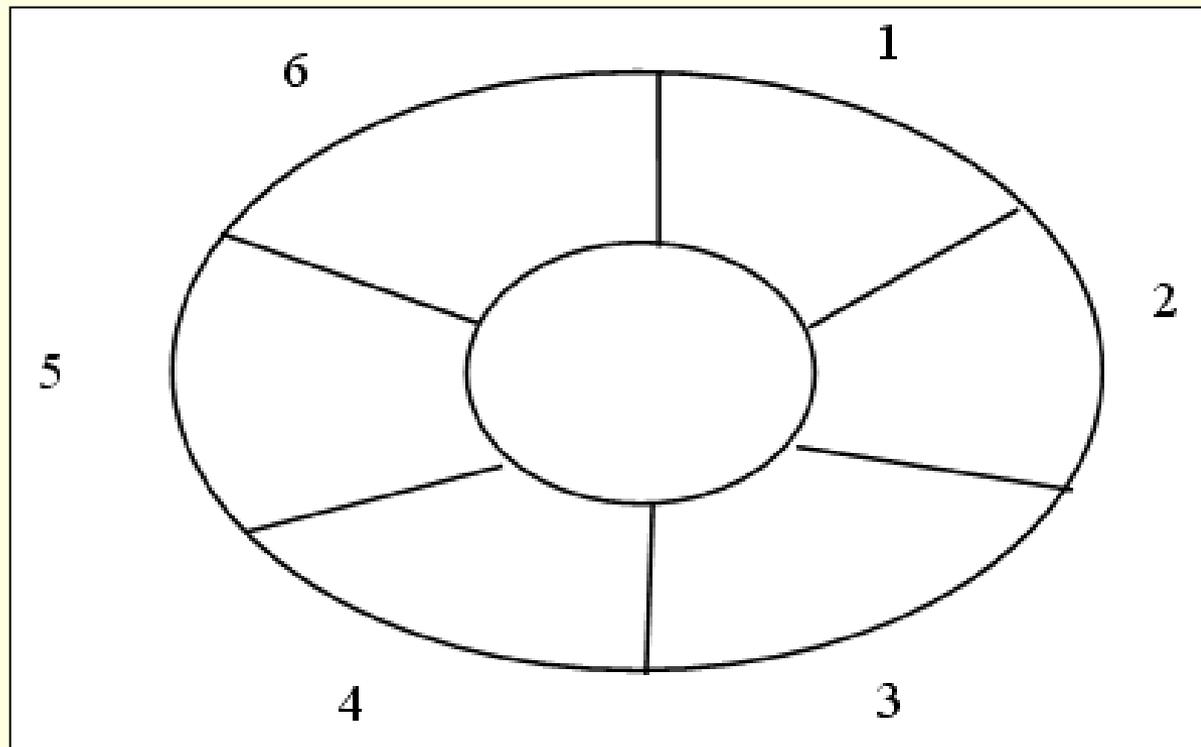
# Fila em vetor circular

---

- Qual a condição para fila vazia?
  - Total=0
- Qual a condição para fila cheia?
  - Total=tamanho da fila
- Qual a condição inicial (quando a fila é criada)?
  - Total=0, início=1, fim=1

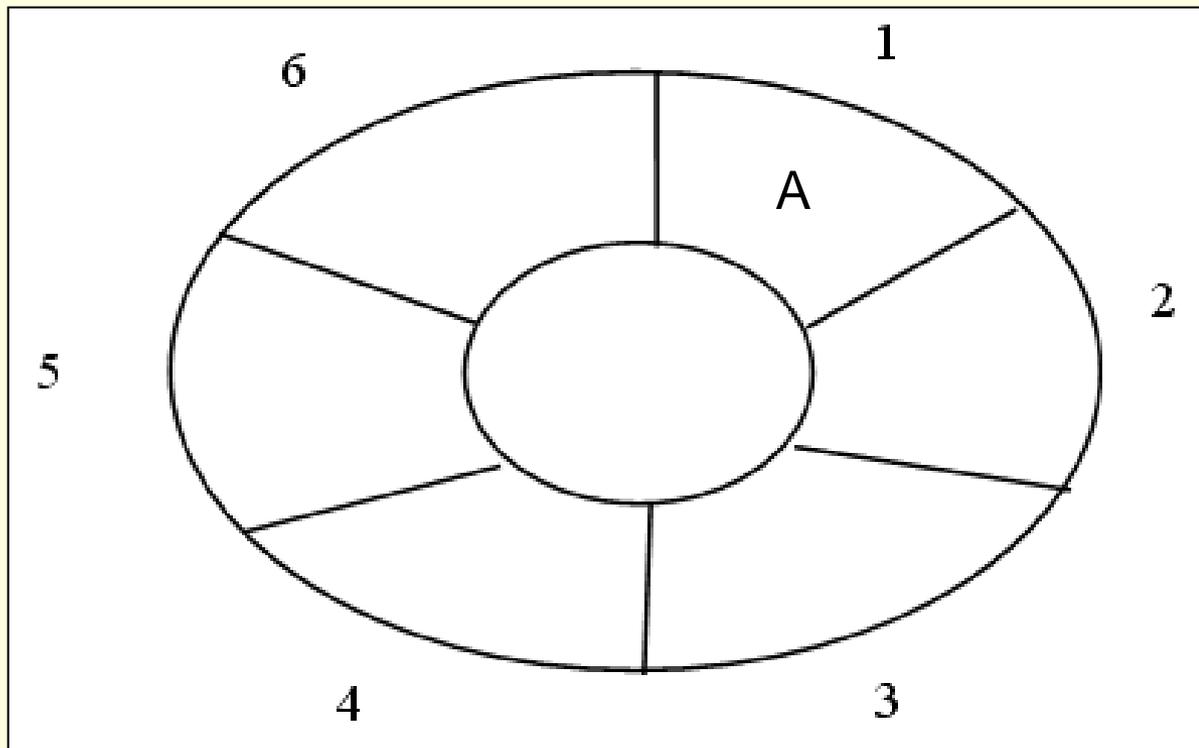
# Exemplo

- Fila criada
  - Início=1, fim=1, total=0



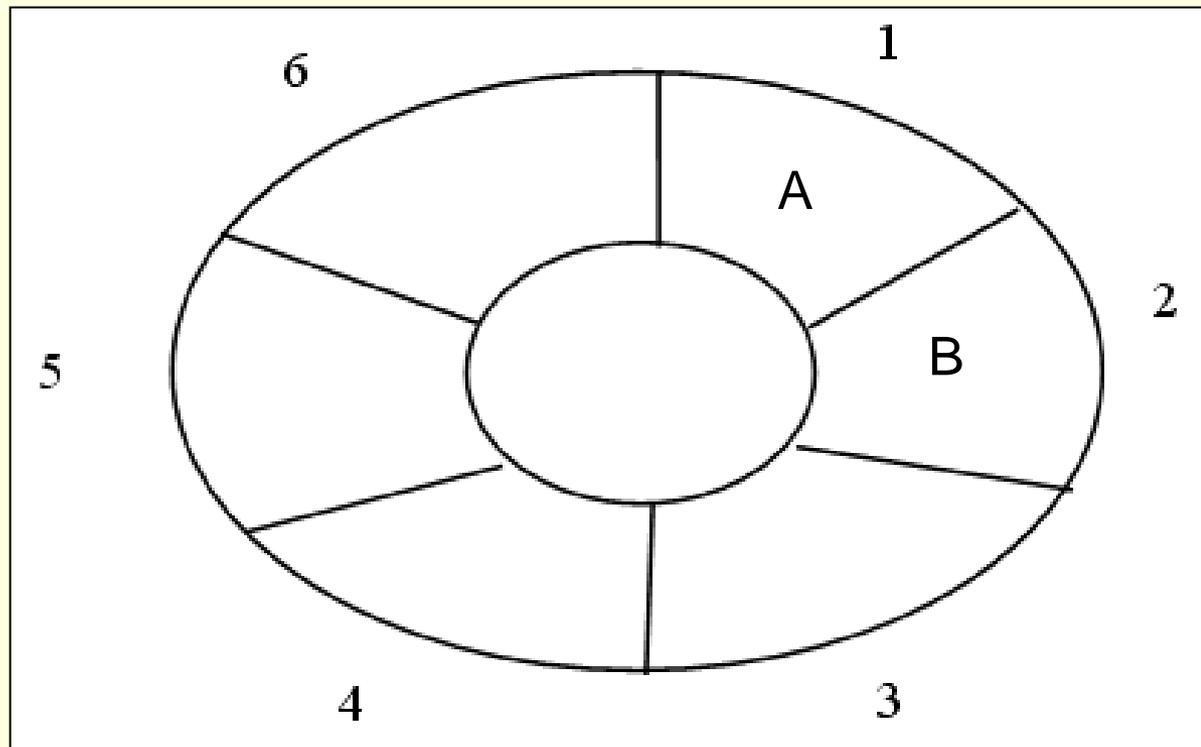
# Exemplo

- Entra A
  - Início=1, fim=2, total=1



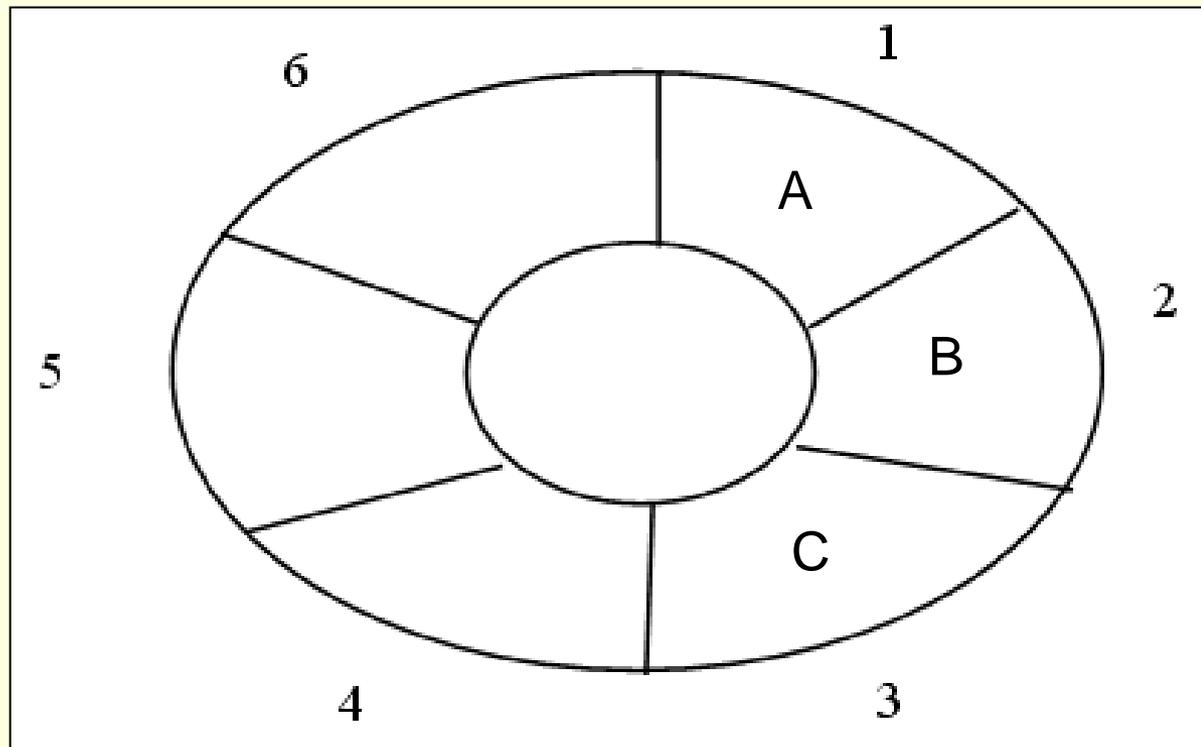
# Exemplo

- Entra B
  - Início=1, fim=3, total=2



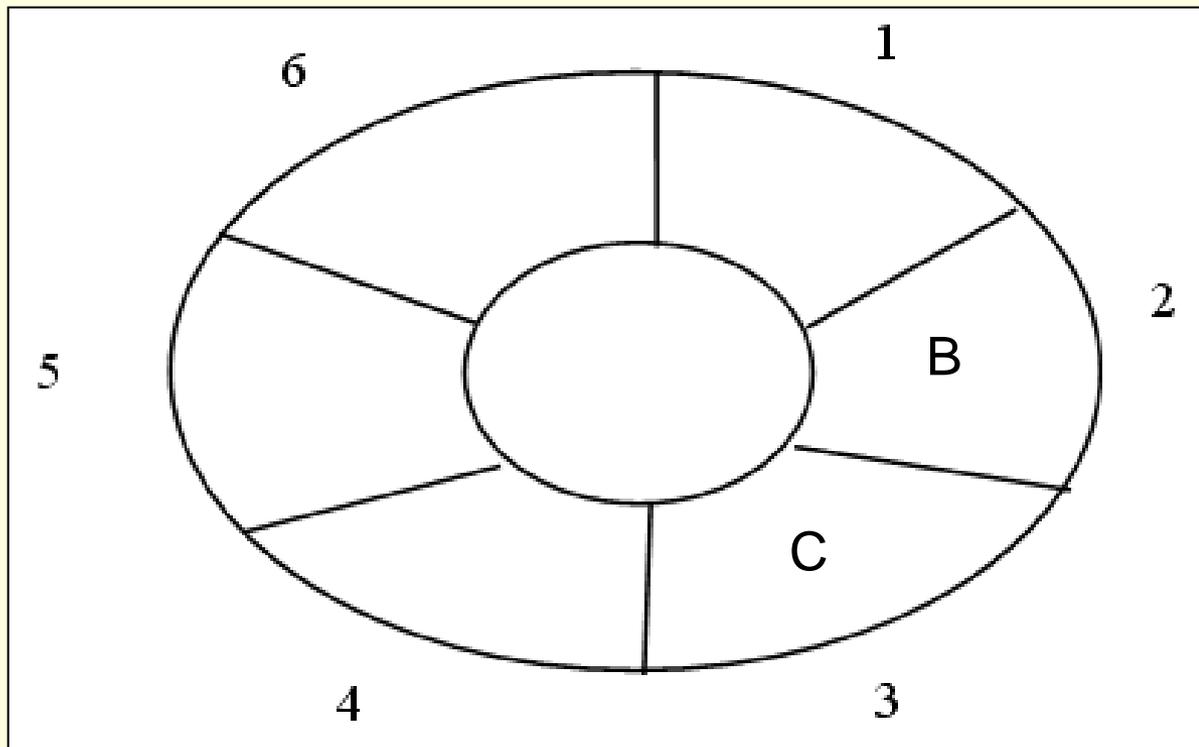
# Exemplo

- Entra C
  - Início=1, fim=4, total=3



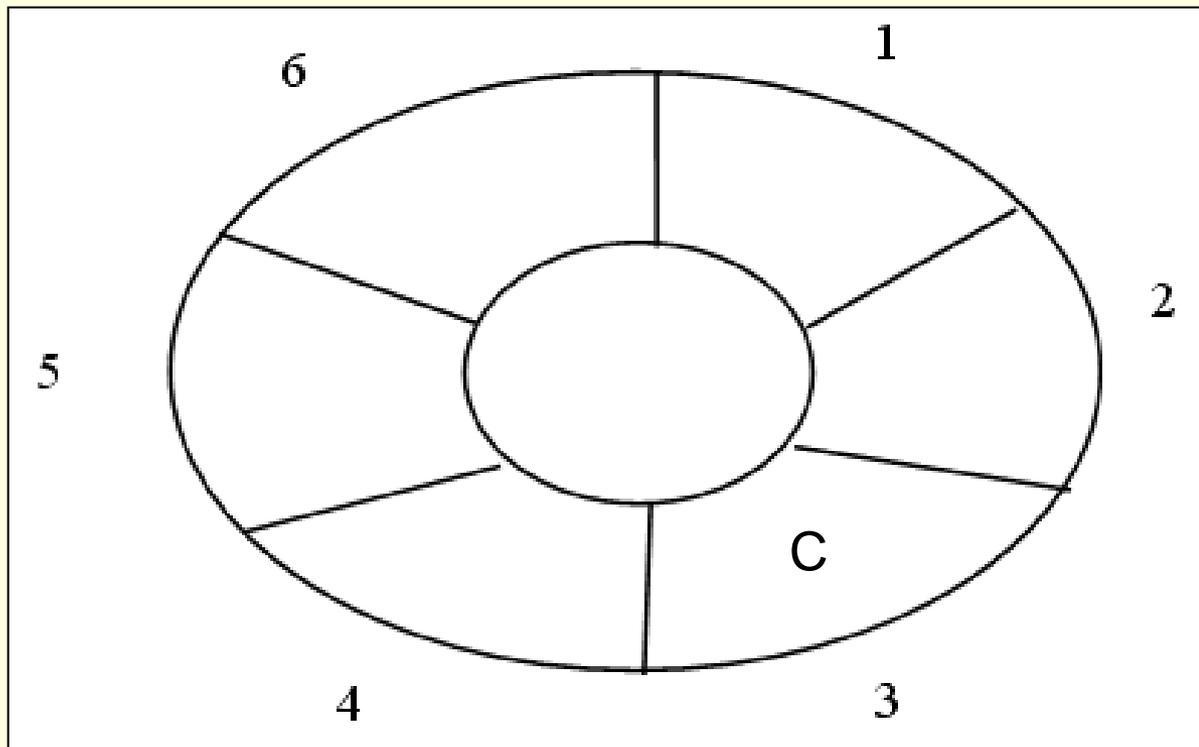
# Exemplo

- Sai primeiro
  - Início=2, fim=4, total=2



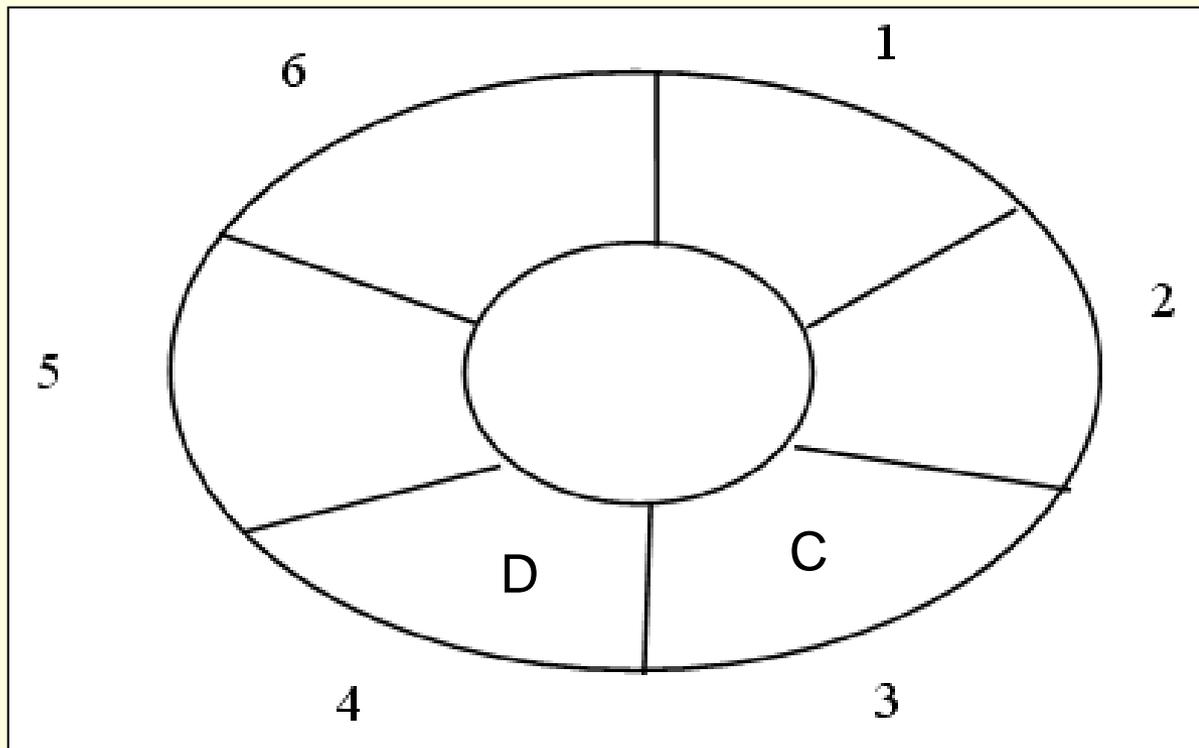
# Exemplo

- Sai primeiro
  - Início=3, fim=4, total=1



# Exemplo

- Entra D
  - Início=3, fim=5, total=2



# Passo a passo para *Entra* e *Sai*

---

- **Entra elemento no fim da fila**
  - `vetor[fim]=elemento`
  - avança fim (“fazer a curva”, se preciso)
  - atualiza total
  
- **Sai primeiro elemento**
  - `elemento=vetor[início]`
  - avança início (“fazer a curva”, se preciso)
  - atualiza total

# Implementação da fila

---

- Declaração em C

?

# Implementação da fila

---

## ■ Declaração em C

```
#define TamFila 100

typedef int elem;

typedef struct {
    int inicio, fim, total;
    elem itens[TamFila];
} Fila;

Fila F;
```

# Operações sobre a fila

---

- Implementar as operações
  - Create
  - Empty
  - IsEmpty
  - IsFull
  - Entra
  - Sai
  
- Atenção: considerações sobre TAD
  - Arquivos .c e .h, parâmetros, mensagens de erro

# Exercício

---

- Faça uma rotina para imprimir os elementos de uma fila

# Exercício

---

- Faça uma rotina para verificar se os elementos de uma fila estão ordenados de forma crescente

# Exercício

---

- Faça uma rotina que inverta uma fila F1, criando-se uma nova fila F2

# Exercício

---

- Desafio: como criar uma fila “mais genérica” que possa guardar tipos diferentes?
  - TAD ainda melhor!

# Exercício para casa

---

- Implemente um sistema para a biblioteca usando o TAD fila
  - Cada livro deve ser representado por um registro (struct)
    - Nome do livro, disponibilidade, fila de espera
  - Ao requisitar um livro, a pessoa entra na fila de espera se o livro não estiver disponível
  - Quando um livro fica disponível, o primeiro da respectiva fila de espera deve receber o livro
  - Implemente as demais funcionalidades (cadastra livro, retira livro etc.) que julgar necessárias

# Créditos

---

- *Material gentilmente cedido pelo Prof. Thiago A. S. Pardo*