



SÃO
PAULO
TECH
SCHOOL



ED

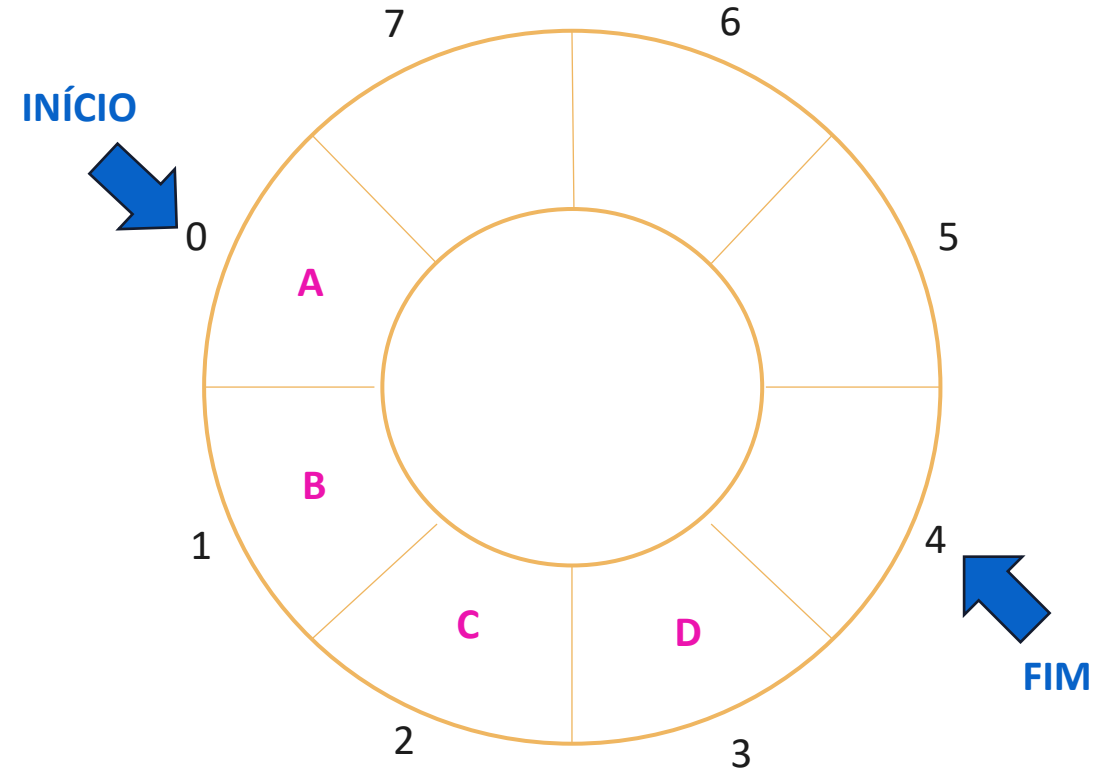
Estrutura de Dados e Armazenamento

Fila Circular

Profa. Célia Taniwaki

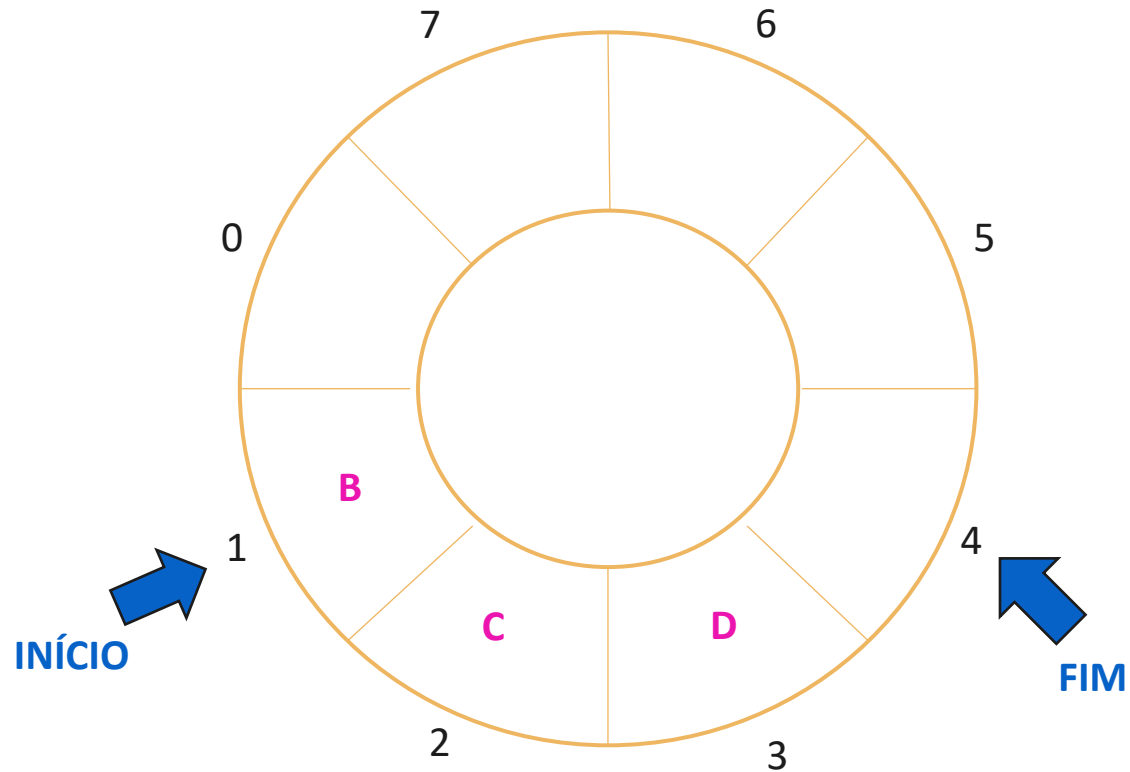
Fila Circular

- E se não quisermos deslocar todos os elementos da fila, na hora de remover um elemento?
- A solução é utilizarmos uma **fila circular**
- Nesse caso, é como se o final do vetor se “emendasse” com o início do vetor, formando um círculo
- Considera-se que a posição 0 (zero) do vetor seja a posição seguinte à posição $n-1$ do vetor (n = tamanho do vetor = `vetor.length`)

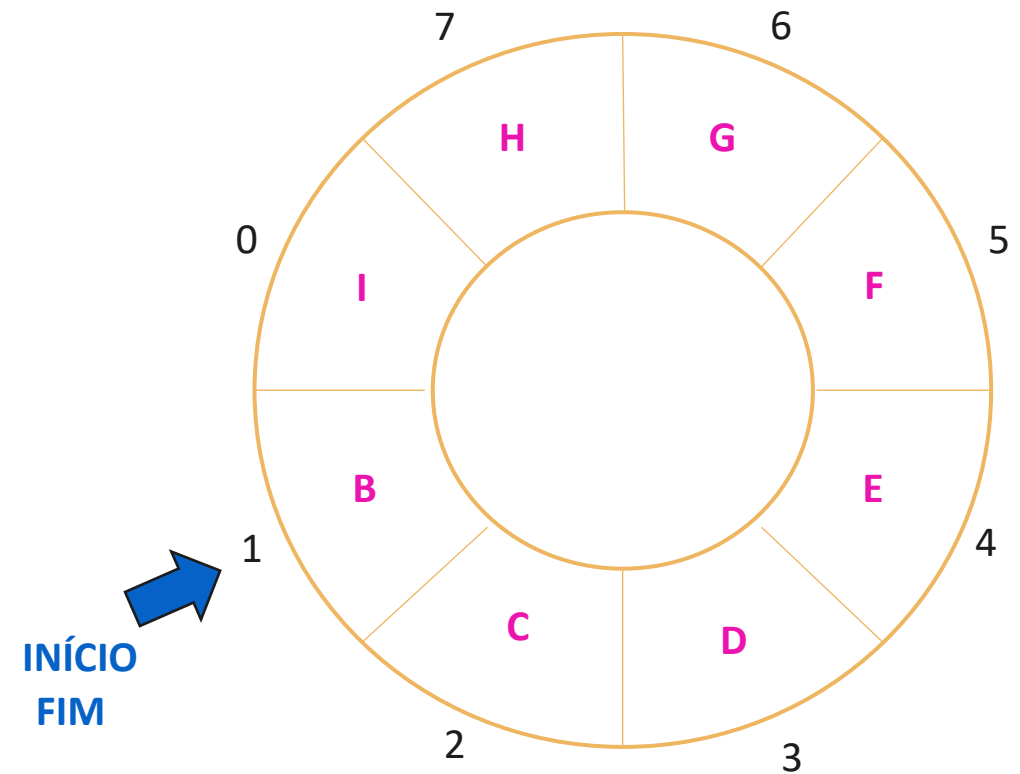


Fila Circular

Após a remoção de um elemento:

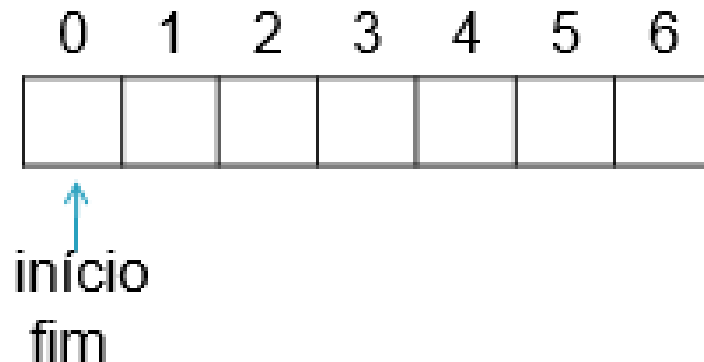


Quando a fila ficar cheia:



Implementação da fila circular usando vetor

- Criar a classe FilaCircular, utilizando a fila como vetor
 - Atributos:
 - `int tamanho` /* tamanho da fila */
 - `String[] fila` /* vetor que representa a fila */
 - `int inicio` /* início da fila */
 - `int fim` /* fim da fila */
 - Construtor, que recebe a capacidade da fila, cria o array para a fila, inicializa tamanho, inicio e fim com zero



Implementação da classe FilaCircular

Métodos da classe FilaCircular (n é a capacidade da fila = fila.length)

- **boolean isEmpty()**: devolve true caso a fila esteja vazia
- **boolean isFull()**: devolve true caso a fila esteja cheia
- **void insert(String info)**: se fila não está cheia, então insere info em fila[fim], faz fim = (fim+1) % fila.length e incrementa tamanho
- **String peek()**: retorna fila[inicio]
- **String poll()**: se fila não está vazia, salva fila[inicio] numa variável, faz inicio = (inicio+1) % fila.length, decrementa tamanho e retorna o valor salvo na variável
- **void exhibe()**: se fila está vazia, exhibe "Fila vazia", senão exhibe os elementos da fila

Implementação da classe FilaCircular

- Explicação sobre o “incremento” de fim (o mesmo vale para início)
 - A ideia é fazer com que fim tenha o valor dos índices do vetor fila, mas, ao chegar ao último índice, fim volte a ser zero, para que a fila seja tratada como circular
 - Por exemplo, consideremos uma fila com capacidade 5:



- Nesse caso, `fila.length = 5`
- `fim` é inicializado com zero
- Ao incrementarmos `fim`, ao invés de fazermos `fim++`, costumamos fazer

`fim = (fim + 1) % fila.length`

Implementação da classe `FilaCircular`

- Explicação sobre o “incremento” de fim (o mesmo vale para início)
 - Dessa forma, quando fim for zero, $(fim+1) \% fila.length = 1$
 - Quando fim for 1, $(fim+1) \% fila.length = 2$ (resto de 2 por 5)
 - Quando fim for 2, $(fim+1) \% fila.length = 3$ (resto de 3 por 5)
 - Quando fim for 3, $(fim+1) \% fila.length = 4$ (resto de 4 por 5)
 - Quando fim for 4, $(fim+1) \% fila.length = 0$ (resto de 5 por 5)



`fila.length = 5` (neste exemplo)

Implementação da classe FilaCircular

- Explicação sobre o incremento de fim (o mesmo vale para inicio)
- Esse “incremento” pode ser feito de várias formas:

```
fim = (fim + 1) % fila.length;
```

OU

```
fim++;
```

```
fim = fim % fila.length;
```

OU

```
fim++;
```

```
if (fim == fila.length) {
```

```
    fim = 0;
```

```
}
```

Implementação da classe `FilaCircular`

Explicação sobre o método `exibe()`:

- Deve ter um loop que usa 2 contadores:
 - O índice para percorrer o vetor fila, que deve começar com o valor de início e ser “incrementado” da mesma forma que o fim e o início, ou seja, quando esse índice for o último índice do vetor fila, tem que virar zero.
 - Um contador que começa valendo zero, e que controla quantas vezes vai printar um valor do vetor. Esse contador é que vai controlar o loop, enquanto contador for menor do que tamanho.
 - Obs: não é possível comparar o índice com fim, para controlar o loop, pois na nossa implementação, início pode ser igual a fim quando a fila está vazia ou quando a fila está cheia.

Agradeço
a sua atenção!



SÃO
PAULO
TECH
SCHOOL