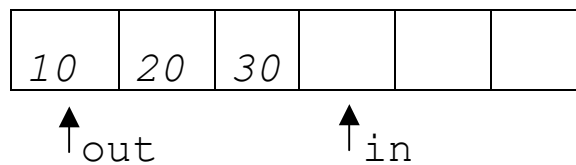


Métodos Parciais e Pré-Condições

Suponhamos uma classe que implementa uma estrutura do tipo fila (First In First Out):

Fila (circular) implementada através de um array,



out – posição do próximo elemento a retirar

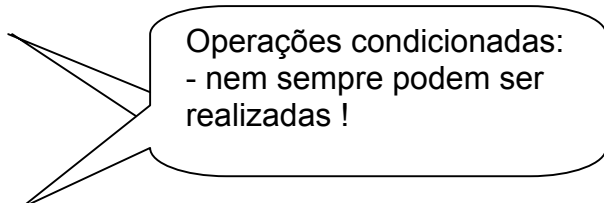
in – posição onde será colocado o próximo elemento

dim – dimensão do array

nElementos – número de elementos da fila

```
public class Fila {
    // variáveis de instância
    private int[] fila;
    private int nElementos, dim, in, out;
    // construtor
    public Fila ( int N ) {
        dim = N;
        fila = new int [dim];
        in = 0;
        out = 0;
    }
}
```

```
// métodos de instância
public void inserir ( int e) {
// se a fila não estiver cheia
    fila[in] = e;
    nElementos ++;
    in = (in +1) % dim;
}
public int retirar () {
// se a fila não estiver vazia
    int e = fila[out];
    nElementos --;
    out = (out + 1) % dim;
    return (e);
}
```



Operações condicionadas:
- nem sempre podem ser realizadas !

Como resolver?

Hipótese 1: - colocar um parâmetro de retorno que indica o sucesso da operação.

```
public boolean inserir (int e){
    If ( nElementos != dim ) {
        ...
        return true;
    } else {
        retrun false;
    }
}
```

Ao invocar o método,

...

```
Fila filaA = new Fila (10);  
boolean sucesso = filaA.inserir (5);  
if (! sucesso )  
    System.out.println ("Erro: fila cheia");
```

O que fazer no caso da operação retirar?

- já temos um resultado
- podemos definir um novo tipo de objecto ...

A solução irá ser introduzir no código a detecção de situações de erro (exceções) e o seu tratamento.

Por agora,

2ª hipótese de solução: - implementar, na classe Fila, métodos que funcionam

como pré condições das operações condicionadas.

```
public boolean cheia () {  
  
    return (nElementos == dim );  
  
}
```

```
public boolean vazia () {  
  
    return (nElementos == 0 );  
  
}
```

Utilização:

```
Fila filaA = new Fila(10);  
  
if ( ! filaA.cheia())  
  
    filaA.inserir(5);  
  
else  
  
    System.out.println ( "Erro - fila cheia");  
  
...  
  
if ( ! filaA.vazia() )  
  
    filaA.retirar();  
  
else  
  
    System.out.println ( "Erro - fila vazia")
```