


 Instituto Superior de Economia e Gestão  
 UNIVERSIDADE TÉCNICA DE LISBOA

---

## Mestrado Decisão Económica e Empresarial COMPUTAÇÃO

Sumário:

Pilhas, Filas e Listas.

## Filas

---

Pilha – exemplo tubo com bolas de ténis. (LIFO)  
(vimos outro exemplo na aula passada)

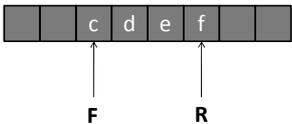
Fila – fila para impressão numa impressora partilhada por várias máquinas;  
fila de trânsito ...  
(FIFO)

DEE- Computação 2010/11 Lspinto 2

## Implementação de fila com um vector

---

Vector Q



Ponteiros F e R indicam a posição da frente e retaguarda

F é a posição do próximo elemento a ser removido da fila  
R + 1 é a posição onde deve ser inserido o próximo elemento

DEE- Computação 2010/11 Lspinto 3

## Operações básicas

---

- Criação de uma fila **CriaF**
- Inserir um elemento (no final) **Finsere**
- Retirar um elemento (do início) **Fapaga**
- Verificar se está vazia **EstaVazia**
- Obter o valor do primeiro elemento da fila sem o remover **InicioF**

DEE- Computação 2010/11 Lspinto 4

## Operações básicas

---

- Criação de uma fila **CriaF**  
Dimensiona o vector da fila e inicializa F e P em zero
- Inserir o elemento Y na fila Q com inicio F e fim R  
**Procedure Finsere(Q,F,R,Y)**
  1. Testar a capacidade da fila
    - If  $R \geq N$  Then
    - Mensagem (excedeu a dimensão da fila)
    - Return
  2. Incrementar o ponteiro da retaguarda
    - $R=R+1$
  3. Inserir o elemento
    - $Q(R)=Y$
  4. Verificar o ponteiro da frente
    - If  $F=0$  Then
    - $F=1$
    - Return

DEE- Computação 2010/11 Lspinto 5

## Operações básicas

---

- Retirar o primeiro elemento da fila Q com inicio F e fim R  
**Function Fapaga(Q,R,F)**
  1. Verificar se a fila contém elementos
    - If  $F=0$  Then
    - Mensagem (A fila está vazia)
    - Return
  2. Armazenar o elemento a eliminar
    - $Y=Q(F)$
  3. Testar se a fila fica vazia
    - If  $F=R$  Then
      - $F=0$
      - $R=0$
    - Else  $F=F+1$
  4. Devolver o elemento
    - Return(Y)

DEE- Computação 2010/11 Lspinto 6

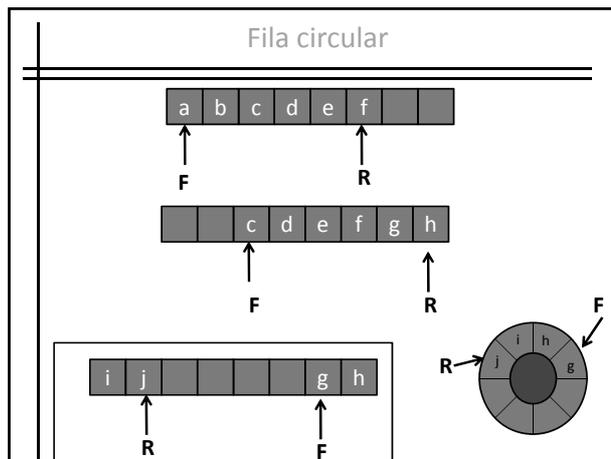
### Operações básicas

---

**Verificar se está vazia** **Function EstaVazia(R,F)**

1. Testar se existem elementos na fila
  - If F=0 Then
  - Mensagem( Fila vazia)
  - Return(true)
  - Else
  - Return(false)
- Obter o primeiro elemento da fila Q com inicio F sem o remover **Function InicioF(Q,F)**
  1. Verificar se a fila contém elementos
    - If F=0 Then
    - Mensagem (A fila está vazia)
    - Return
  2. Devolver o elemento da frente da fila
    - Y=Q(F)

DEE- Computação 2010/11 Lspinto 7



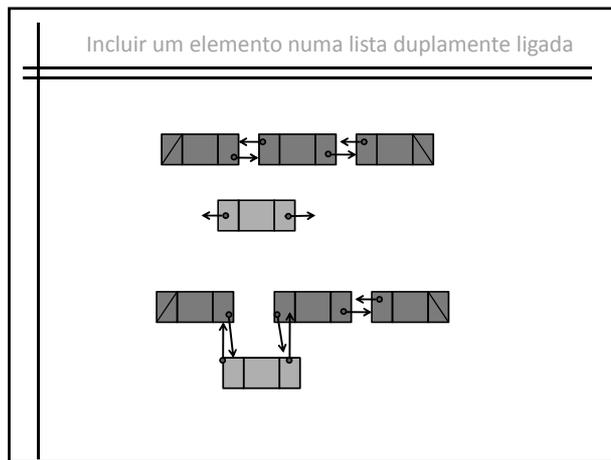
### Tipos de Listas

---

1. **Lista duplamente encadeada**

Cada elemento é um objeto com um campo chave e dois ponteiros: ant e prox .
2. **Lista simplesmente encadeada**

Omitimos em cada elemento o ponteiro ant .



### Exercício - aplicação de filas

---

Considere 6 cidades numeradas de 0 a 5 e interligadas por estradas de sentido único. As ligações entre as cidades são representadas por uma matriz:

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
|   | 0 | 1 | 2 | 3 | 4 | 5 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 0 | 1 | 0 |
| 4 | 1 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 1 | 0 | 0 | 0 | 0 |

$A[i][j]$  vale 1 se existe estrada da cidade  $i$  para a cidade  $j$  o.c.c.

A distância de uma cidade  $c$  a uma outra  $j$  é o menor número de estradas a percorrer para ir de  $c$  a  $j$ . Dada uma cidade  $c$ , determinar a distância de  $c$  a cada uma das restantes cidades.

**algoritmo :**

- uma cidade é considerada *ativa* se já foi visitada mas as estradas que nela começam ainda não foram exploradas;
- mantenha uma fila das cidades ativas;
- em cada iteração, remova da fila uma cidade  $i$  e insira na fila todas as cidades vizinhas de  $i$  que ainda não foram visitadas.

Cidade inicial 3, solução:

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
|   | 0 | 1 | 2 | 3 | 4 | 5 |
| 2 | 2 | 2 | 1 | 0 | 1 | 6 |

DEE- Computação 2010/11 Lspinto 13