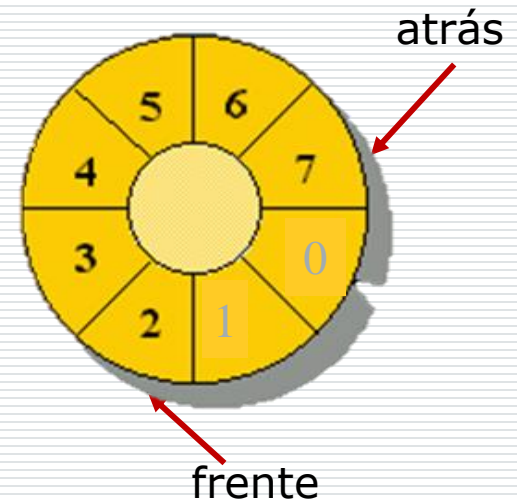


# Tema 3 . Estructuras lineales estáticas y dinámicas

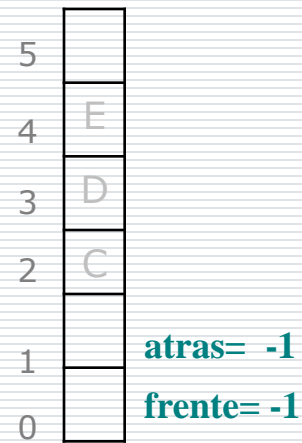
---

## 3.2 Filas o Colas

COLA CIRCULAR



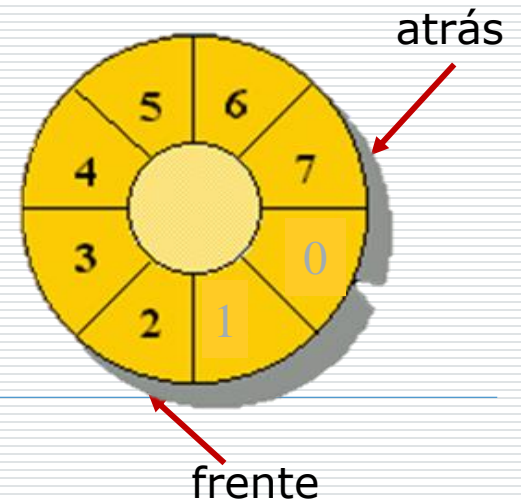
# Fila linear

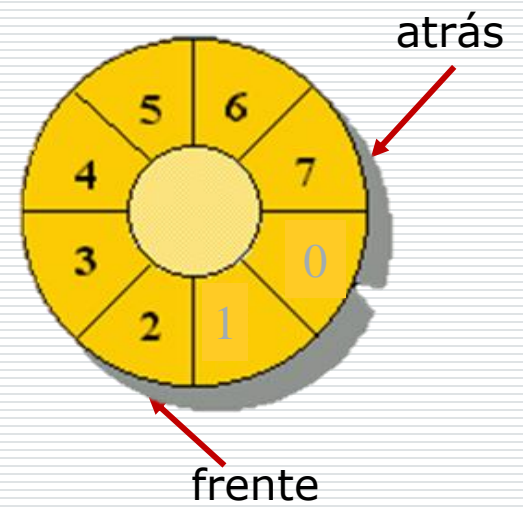
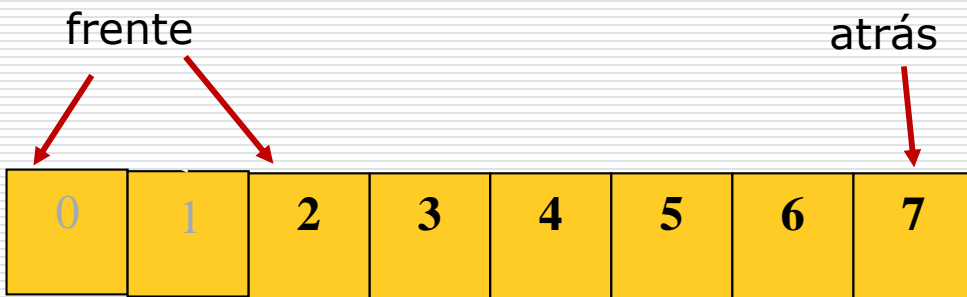


---

## □ COLA CIRCULAR

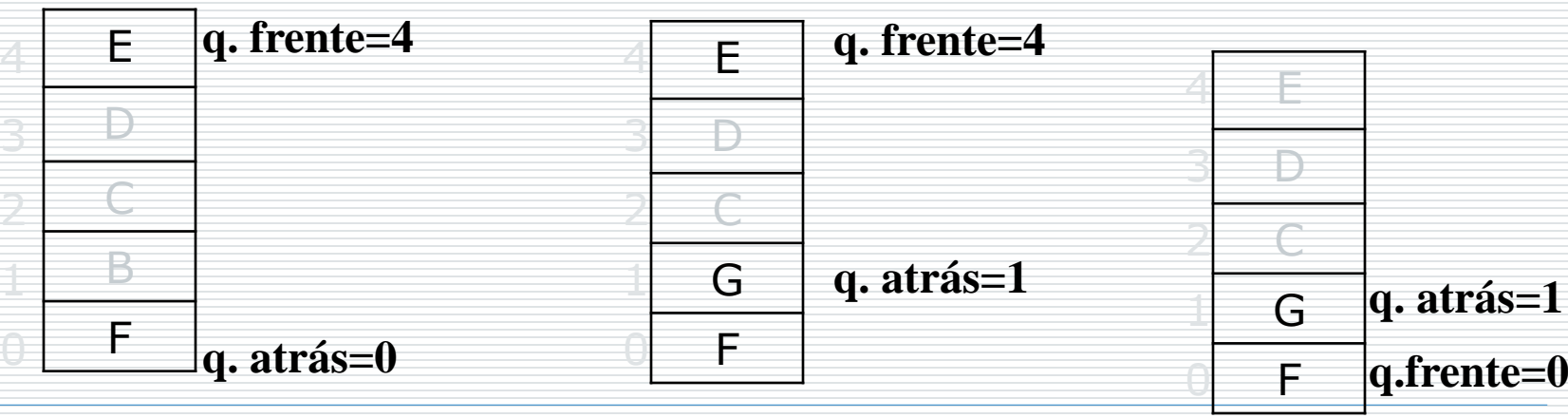
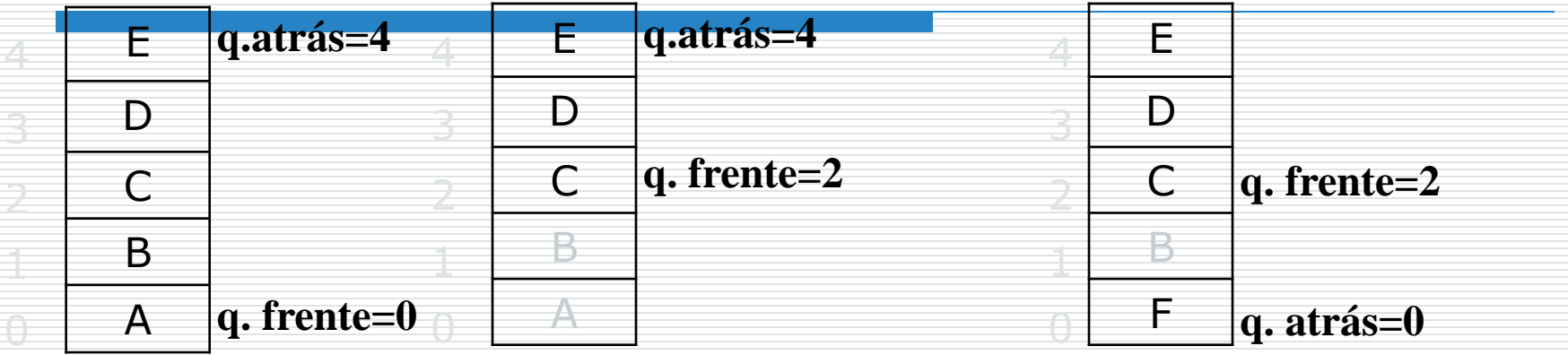
La cola circular es una forma eficiente de aprovechar el espacio que no puede ser aprovechado en las colas lineales.



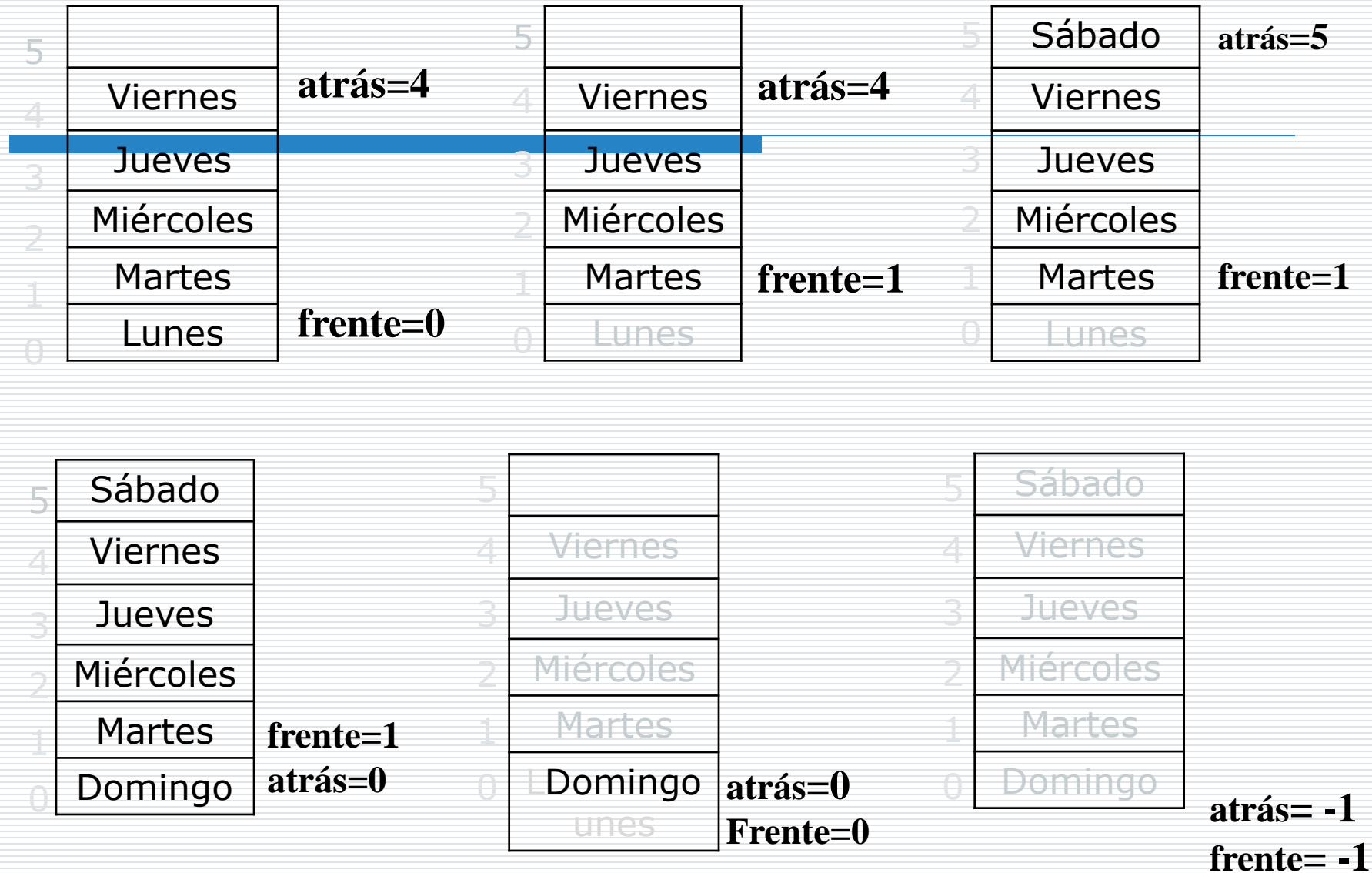


# Cola circular

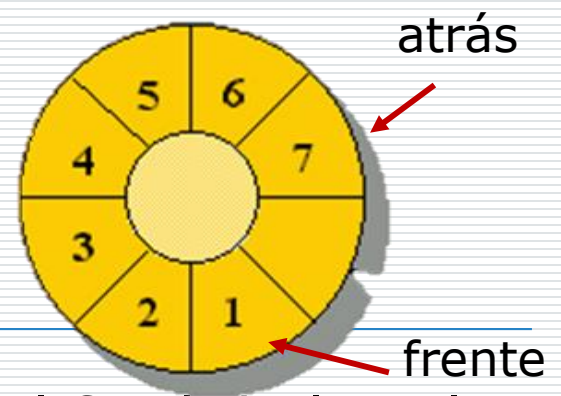
q.elems



# Cola circular



# Métodos



- `insertacircular(x)` --> Agrega un elemento al final de la cola.
- `eliminacircular( )` --> Elimina el primer elemento de la cola.
- `info( )` --> Retorna el primer elemento en la cola.
- `colavacia( )` --> Retorna true si no existen elementos, false en caso contrario.
- `colallena( )` --> Retorna true cuando la cola esta llena false en caso contrario.
- `vaciarcola( )` --> Elimina todos los elementos de la cola.
- `busca(x)` --> Regresa true si encuentra al elemento x en la cola.
- `imprimecola( )` --> Muestra el contenido de la cola.
- `total( )` --> Retorna la cantidad de elementos en la cola

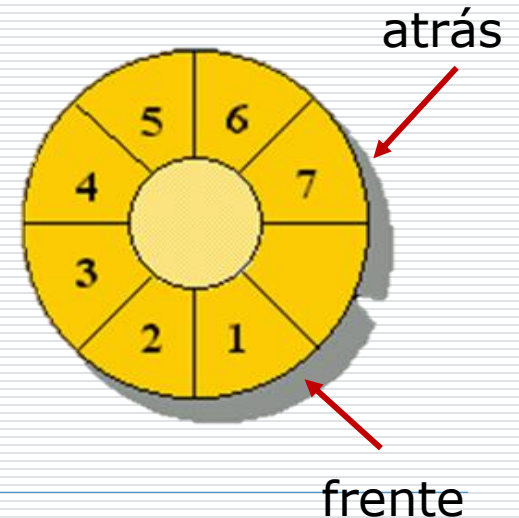
# Métodos

---

busca(x) --> Regresa true si encuentra al elemento x en la cola.

imprimecola( ) --> Muestra el contenido de la cola.

total( ) --> Retorna la cantidad de elementos en la cola.





---

# □ Diseño de algoritmos

---

# Algoritmo colallena()

---

1. Inicio
2. Si  $((\text{atrás} == \text{max}-1) \text{ y } (\text{frente} == 0))$  o  $((\text{atrás}+1) == \text{frente})$   
entonces  
    retorna verdadero  
si no  
    retorna falso
3. Fin condicional\_paso\_2
4. Fin

4	E	<b>atrás=4</b>
3	D	
2	C	
1	B	
0	A	<b>frente=0</b>

5	Sábado	
4	Viernes	
3	Jueves	
2	Miércoles	
1	Martes	<b>frente=1</b>
0	Domingo	<b>atrás=0</b>

# Algoritmo colavacia( )

---

1. Inicio
2. Si frente== -1 y atrás== -1
  - 2.1 Regresa verdadero;
  - 2.2 Si no
  - 2.3 Regresa falso
3. Fin del condicional 2
4. Fin

5	Sábado
4	Viernes
3	Jueves
2	Miércoles
1	Martes
0	Domingo

**atrás= -1**  
**frente= -1**

# Insertacircular( dato)

{Este algoritmo inserta el elemento dato al final de la Colacir. Frente y final son los punteros que indican respectivamente el inicio y fin}

1. Inicio (Si colallena())

2. Si (atrás == max-1) y (frente == 0)  
o ((atrás+1) == frente)

entonces

escribe "Desbordamiento"

si no

2.1 si atrás == max-1

entonces atrás = 0

si no atrás = atrás + 1

2.2 fin del condicional\_2.1

elems[atrás] = dato

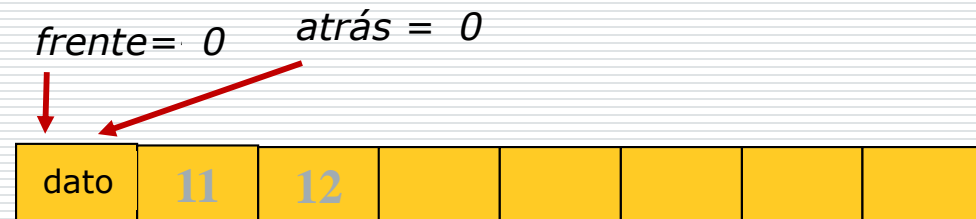
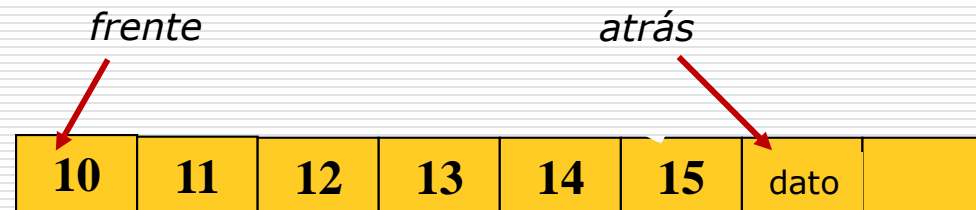
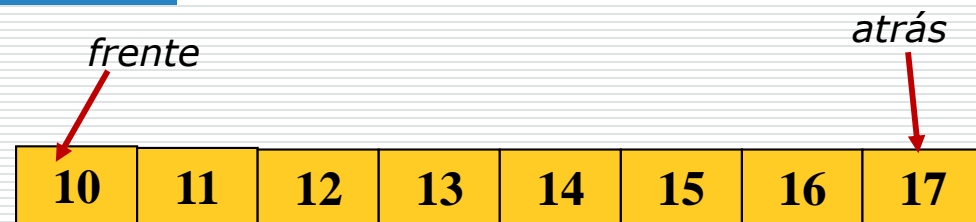
2.3 si frente == -1

entonces frente = 0

2.4 fin

condicional\_paso2.3

del



3. fin del condicional del paso\_2

4. Fin

# Ejercicios

---

- Para el TDA Cola circular, diseña los algoritmos
    - 1) totalcircular()
    - 2) info()
    - 3) imprimecola()
-

# Algoritmo totalcircular()

1.Inicio

2.Si (colallena())

entonces retorna maximo

sino

cont=0

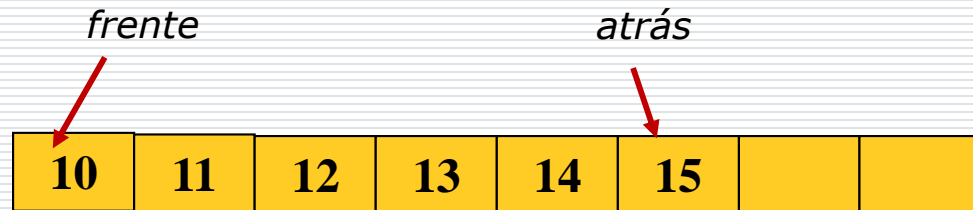
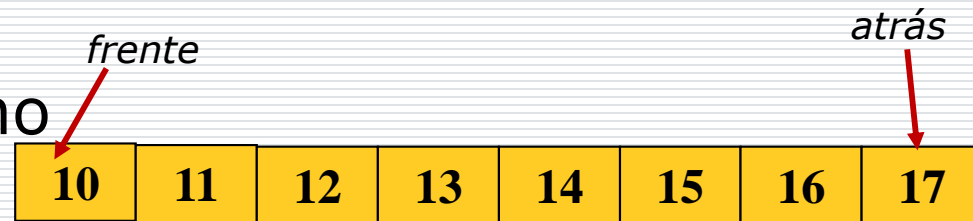
para i desde frente hasta atrás

cont++

retorna cont

3.fin\_condicional\_paso2

4. Fin



frente=-1 atrás=-1



# totalcircular( )

{ }

1. Inicio (Si colallena())

2. Si (atrás == max-1) y (frente == 0) o ((atrás+1) == frente)

entonces

regresa elem.length

si no

2.1 si frente = -1 y atrás = -1

regresa 0

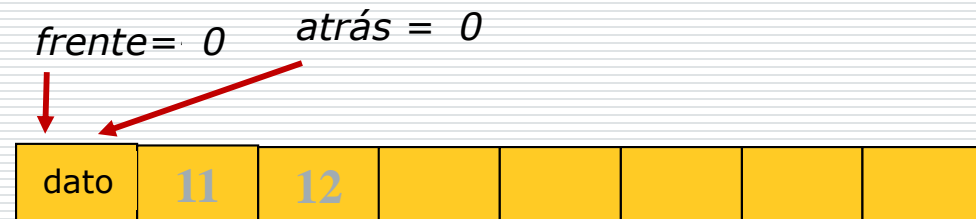
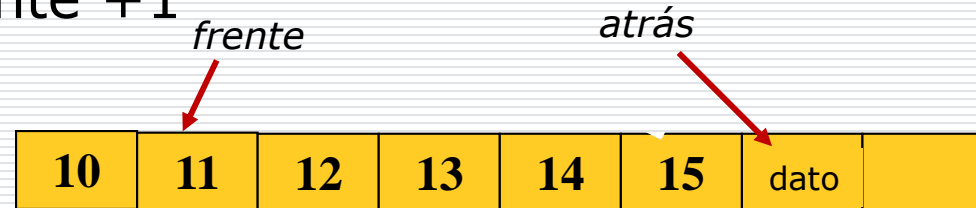
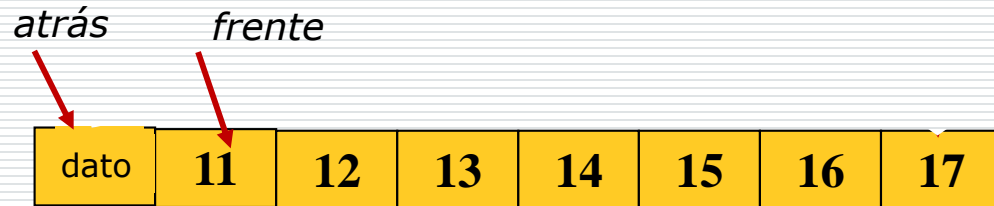
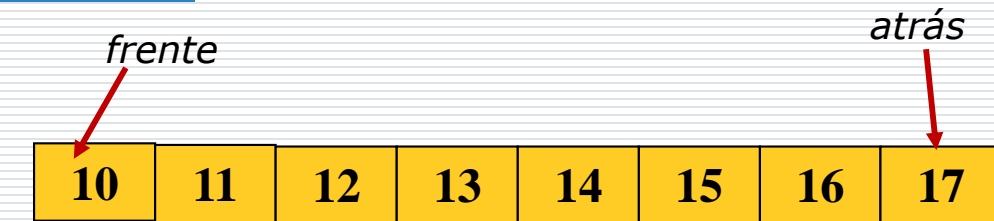
sino

regresa total = atrás - frente + 1

fin del condicional\_paso\_2.1

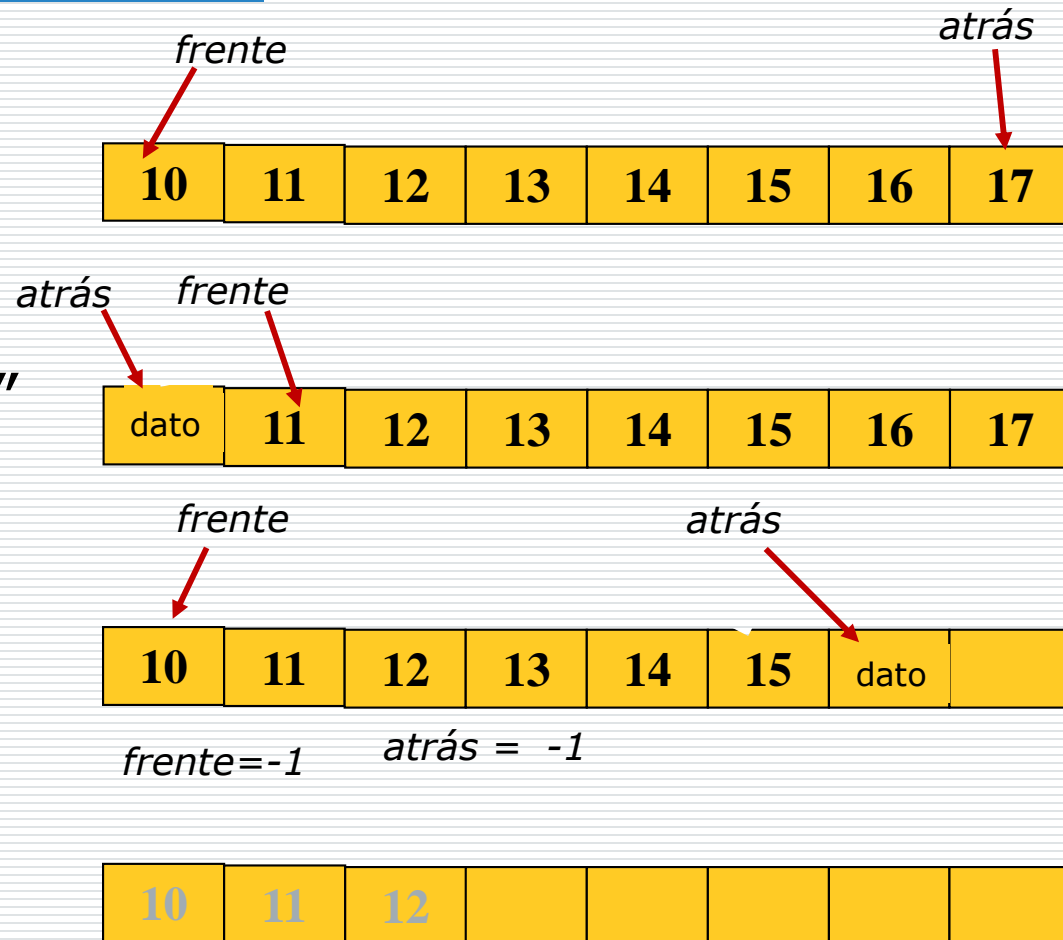
3. fin del condicional del paso\_2

4. Fin



# Algoritmo info()

1. Inicio
2.  $\text{dato} = -1$
3. Si  $\text{colacircvacía}()$
4.    “No hay elementos”
5. sino
6.     $\text{dato} = \text{elem}[\text{frente}]$
7. Regresa dato
8. Fin\_algoritmo





# Eliminacircular( )

{elimina el primer elemento de la cola circular y lo almacena en dato.. frente y final son referencias que indican respectivamente el inicio y fin de la cola, max es el tamaño de la cola circular}

frente=-1    atrás = -1

1. Inicio



2. Si (colavacia()) // se valida si esta vacía  
entonces

frente= -1    atrás = -1

escribe "Subdesbordamiento"

si no



dato= elems[frente]

2.1 si frente==atrás // si hay solo un elemento

entonces frente= -1 atrás= -1

si no

atrás



2.1.1 si frente== max-1

entonces

frente

frente= 0

si no

frente= frente + 1

frente

2.1.2 fin del condicional 2.1.1

2.2 fin del condicional 2.1

atrás



3 fin del condicional del paso 2.

4. Fin

# Ejercicio en clase

## Algoritmo imprimecola()

---



# Aplicaciones de colas

---

- ☐ procesos de las fábricas,
  - ☐ cola del banco,
  - ☐ transporte público,
  - ☐ cola del supermercado
  - ☐ diseños de Call center.
  - ☐ Servidores de internet
  - ☐ en bandeja de entrada de correo.
-

