

# Guía práctica de estudio 06.

## Estructuras de datos lineales: Cola circular y cola doble.

---



---

***Elaborado por:***

M.C. Edgar E. García Cano  
Ing. Jorge A. Solano Gálvez

***Autorizado por:***

M.C. Alejandro Velázquez Mena

# Guía práctica de estudio 06.

## Estructuras de datos lineales: Cola circular y cola doble.

### Objetivo:

Revisarás las definiciones, características, procedimientos y ejemplos de las estructuras lineales Cola circular y Cola doble, con la finalidad de que comprendas sus estructuras y puedas implementarlas.

### Actividades:

- Revisar definición y características de la estructura de datos cola circular.
- Revisar definición y características de la estructura de datos cola doble.
- Implementar las estructuras de datos cola circular y cola doble.

### Introducción

La cola (queue o cola simple) es una estructura de datos lineal, en la cual el elemento obtenido a través de la operación ELIMINAR está predefinido y es el que se encuentra al inicio de la misma.

Una cola simple implementa la política First-In, First-Out (FIFO), esto es, el primer elemento que se agregó es el primero que se elimina.

La cola simple es una estructura de datos de tamaño fijo y cuyas operaciones se realizan por ambos extremos; permite INSERTAR elementos al final de la estructura y permite ELIMINAR elementos por el inicio de la misma. La operación de INSERTAR también se le llama ENCOLAR y la operación de ELIMINAR también se le llama DESENCOLAR.

En una cola simple, cuando se eliminan elementos se recorre el apuntador HEAD al siguiente elemento de la estructura, dejando espacios de memoria vacíos al inicio de la misma. Existen dos mejoras de la cola simple que utilizan de manera más eficiente la memoria: la cola circular y la cola doble.

## Cola circular

La cola circular es una mejora de la cola simple, debido a que es una estructura de datos lineal en la cual el siguiente elemento del último es, en realidad, el primero. La cola circular utiliza de manera más eficiente la memoria que una cola simple.

Debido a que una cola circular es una mejora de la cola simple, maneja las mismas operaciones para INSERTAR (ENCOLAR) y ELIMINAR (DESENCOLAR).

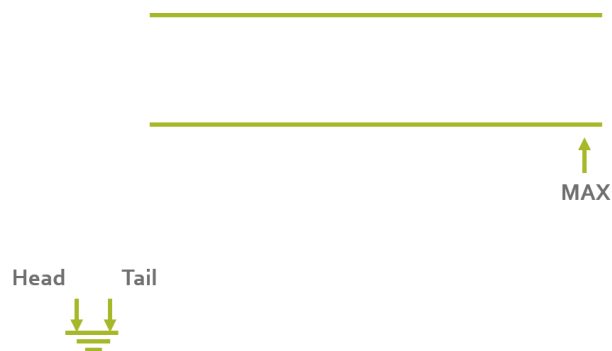
Para diseñar un algoritmo que defina el comportamiento de la cola circular es necesario considerar 3 casos para las operaciones de ENCOLAR y DESENCOLAR:

- Estructura vacía (caso extremo).
- Estructura llena (caso extremo).
- Estructura con elemento(s) (caso base).

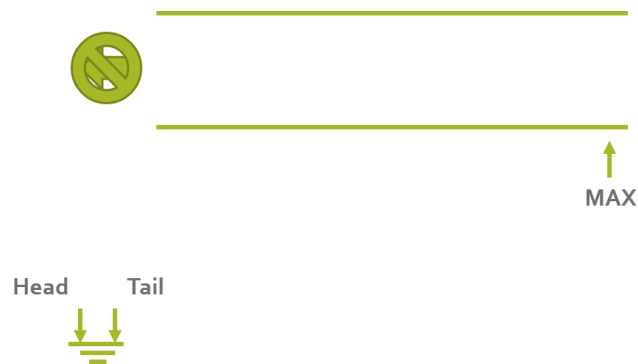
En algoritmo de una cola circular para los casos extremos (cuando la estructura está vacía y cuando la estructura está llena) es el mismo con respecto a la cola simple, el único algoritmo que hay que volver a diseñar se presenta en el caso base, cuando la estructura tiene elementos.

### Cola circular vacía

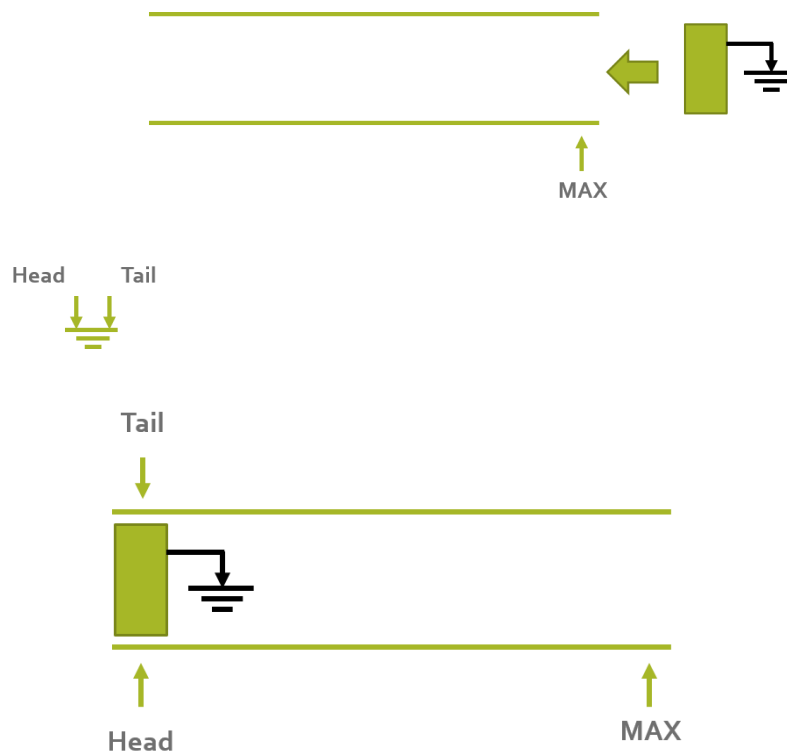
La cola circular posee dos referencias, una al inicio (HEAD) y otra al final (TAIL) de la cola. En una cola circular vacía ambas referencias (HEAD y TAIL) apuntan a *nulo*.



En una cola circular vacía no es posible desencolar debido a que la estructura no posee elementos.

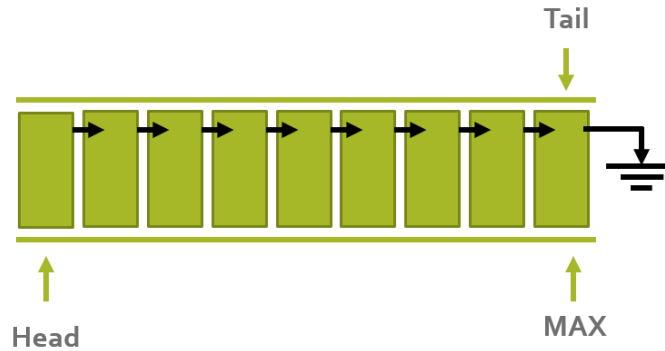


En una cola circular vacía sí se pueden encolar elementos, en este caso las referencias HEAD y TAIL apuntan al mismo elemento, que es el único en la estructura.

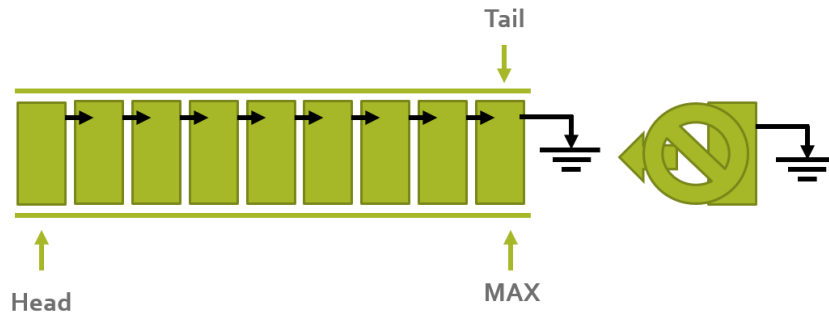


### Cola circular llena

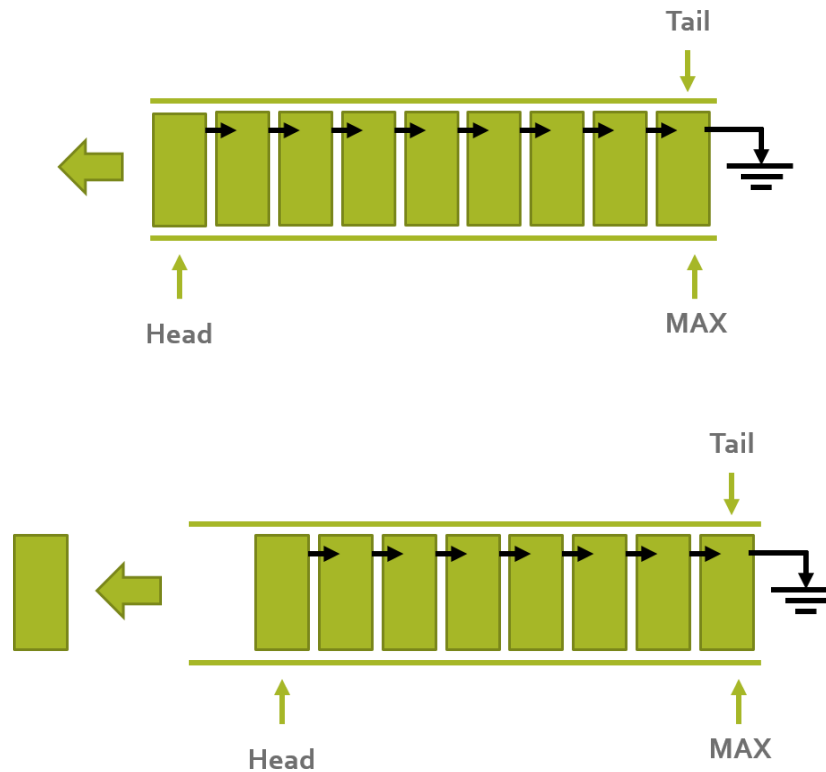
Cuando la referencia a TAIL de una cola llega a su máxima capacidad de almacenamiento (MAX) se dice que la cola está llena.



En una cola circular llena no es posible encolar más elementos.

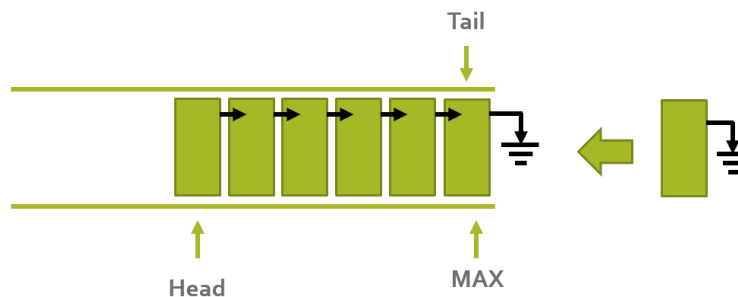


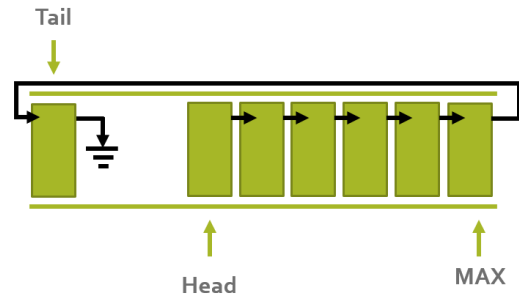
En una cola circular llena sí se pueden desencolar elementos, en tal caso se obtiene el elemento al que hace referencia HEAD y esta referencia se recorre al siguiente elemento.



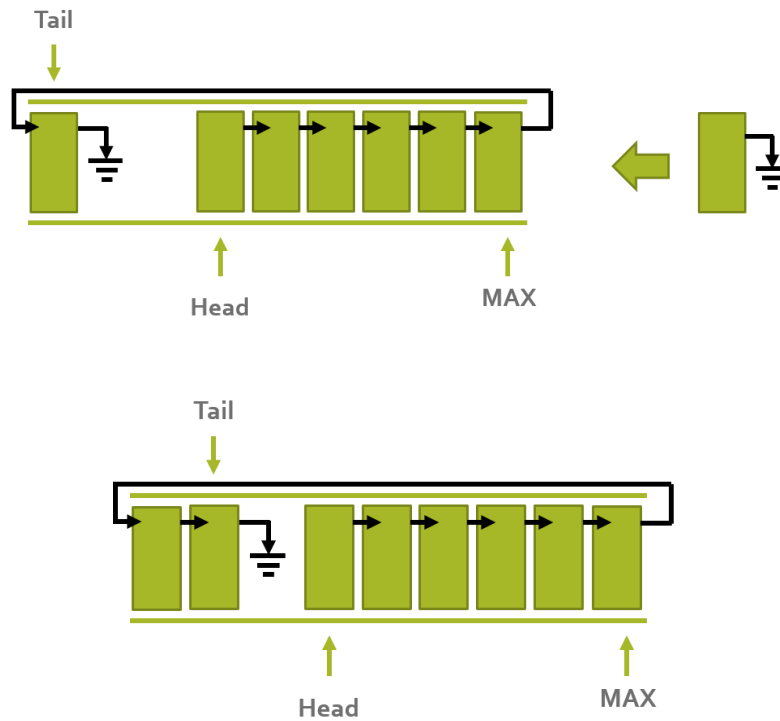
### Cola circular con elementos

En una cola circular con elementos, cuando se intenta insertar un nuevo elemento hay que tener en cuenta el número de los elementos dentro de la estructura y no la referencia TAIL y MAX. Por lo tanto, se debe verificar si el número de elementos que tiene la estructura es menor al número máximo de elementos definidos, si es así, existe espacio para alojar el nuevo elemento y el nuevo nodo se puede insertar.

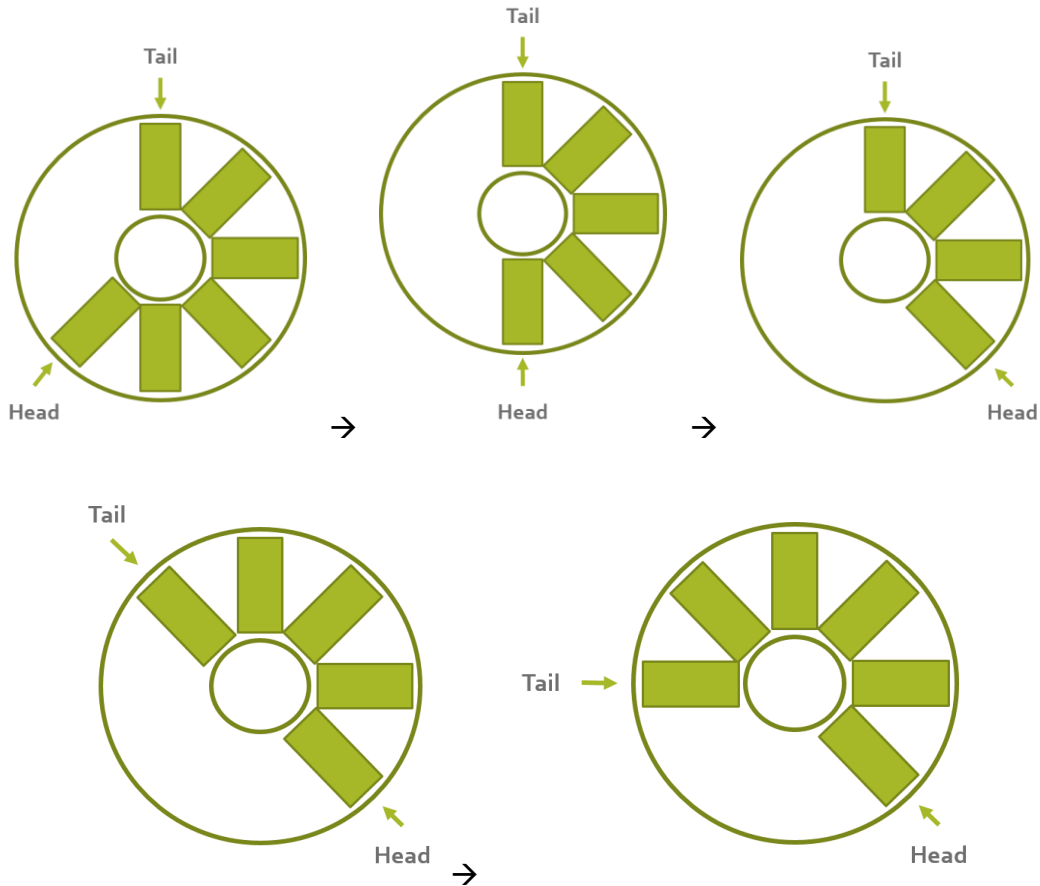




Cada vez que se desee almacenar un nuevo elemento en la estructura se debe revisar el número de elementos insertados y comparar con el número máximo de elementos que se pueden almacenar.



La posibilidad de insertar (ENCOLAR) elementos mientras se tenga espacio disponible hace más eficiente el uso de la memoria, ya que los espacios liberados cada vez que se DESENCOLA un nodo se pueden volver a utilizar, a diferencia de la cola simple.





## Aplicación

La elección de un elemento dentro de un conjunto de datos es muy común en diversas aplicaciones, sobre todo en juegos de consola. La selección de un conjunto de elementos finitos donde a partir del último elemento se puede regresar al primero utiliza, de manera implícita, una cola circular: selección de un personaje, selección de un arma, cambios de uniformes, etc.



**Figura 1.** Cambio de armas en Resident evil.

En general, cuando dentro de una aplicación se puede recorrer un conjunto de elementos finito e invariable en el tiempo y el sucesor del último elemento es el primero se tiene una cola circular.

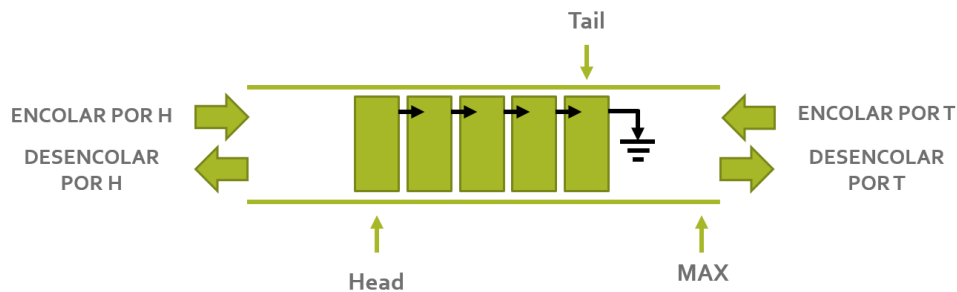


**Figura 2.** Elegir de uniformes en FIFA.

## Cola doble

Una cola doble (o bicola) es una estructura de datos tipo cola simple en la cual las operaciones ENCOLAR y DESENCOLAR se pueden realizar por ambos extremos de la estructura, es decir, en una cola doble se pueden realizar las operaciones:

- ENCOLAR POR HEAD
- DESENCOLAR POR HEAD
- ENCOLAR POR TAIL
- DESENCOLAR POR TAIL



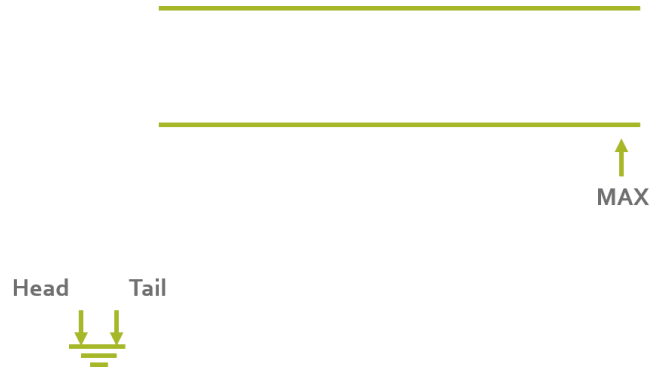
La cola doble es una mejora de una cola simple debido a que es posible realizar operaciones de inserción por ambos extremos de la estructura, permitiendo con esto utilizar el máximo espacio disponible de la estructura.

Para poder diseñar un programa que defina el comportamiento de una COLA DOBLE se deben considerar 3 casos para las 4 operaciones (INSERTAR y ELIMINAR tanto por T como por H):

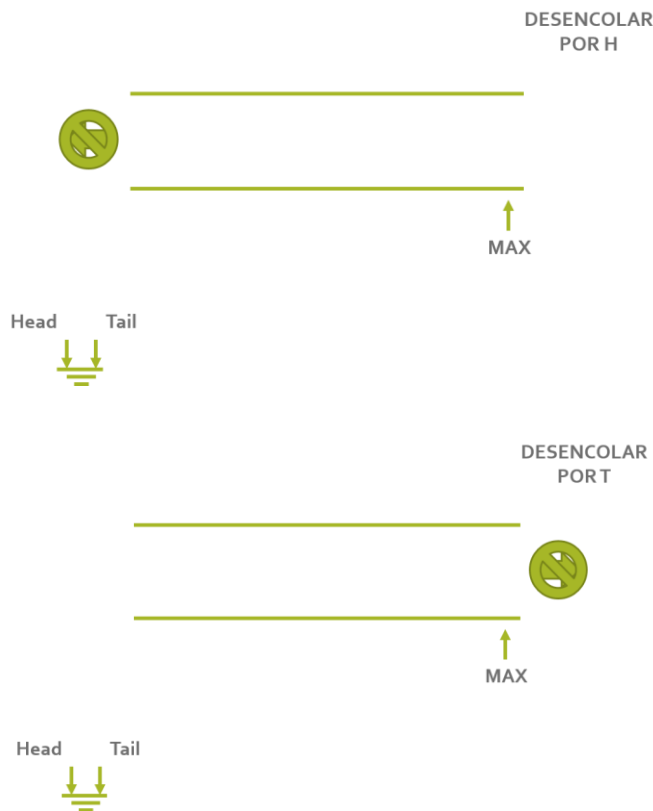
- Estructura vacía (caso extremo).
- Estructura llena (caso extremo).
- Estructura con elemento(s) (caso base).

### Cola doble vacía

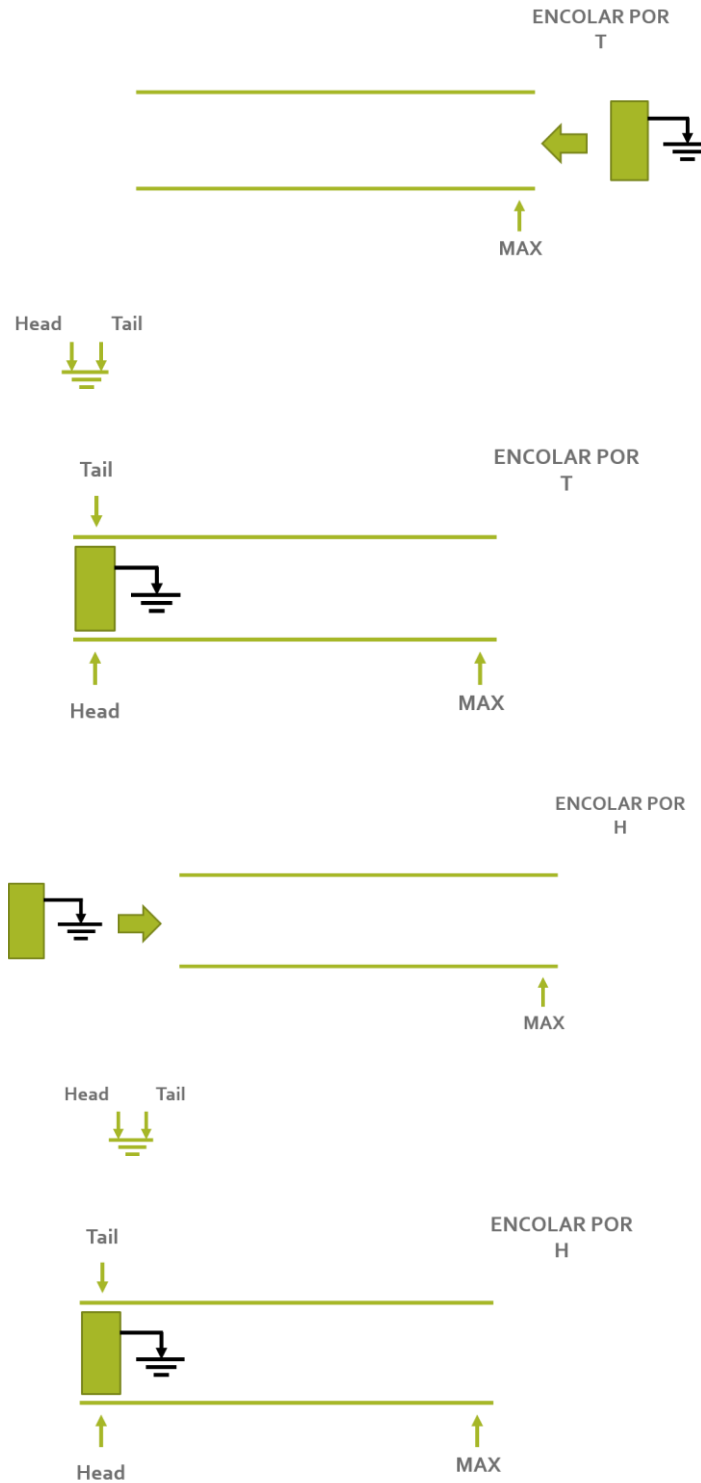
La cola doble posee dos referencias, una al inicio (HEAD) y otra al final (TAIL) de la cola. En una cola doble vacía ambas referencias (HEAD y TAIL) apuntan a *nulo*.



En una cola doble vacía no es posible desencolar debido a que la estructura no posee elementos.

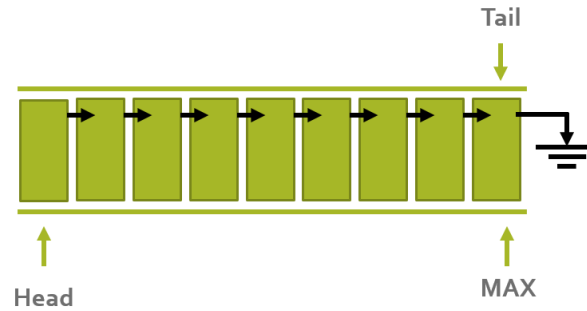


En una cola doble vacía sí se pueden encolar elementos tanto por HEAD como por TAIL, y, en este caso, las referencias HEAD y TAIL apuntan al mismo elemento, que es el único en la estructura.

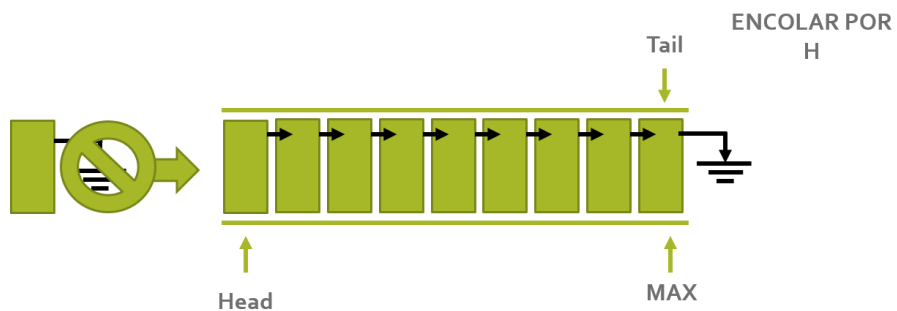
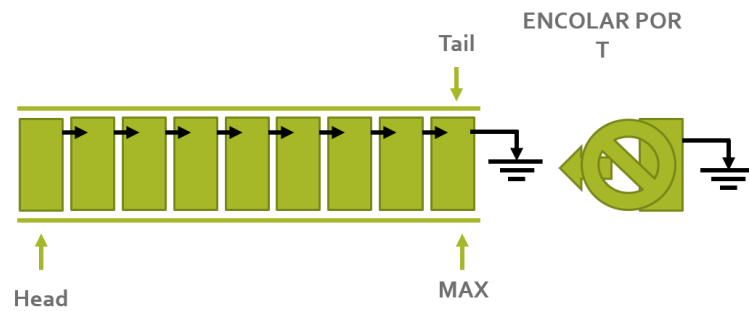


**Cola doble llena**

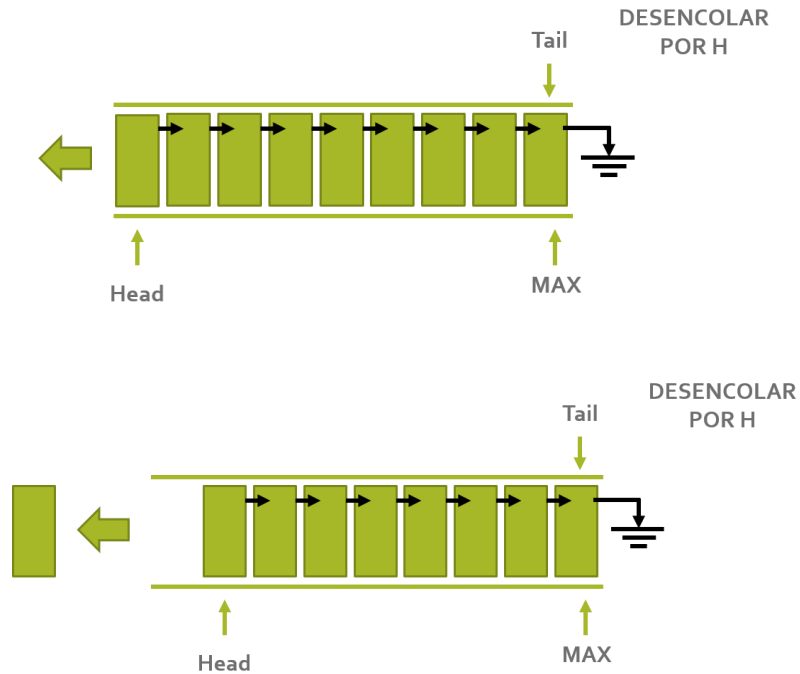
Cuando el número de elementos de la estructura es igual a la capacidad máxima de almacenamiento (MAX) se dice que la cola está llena.



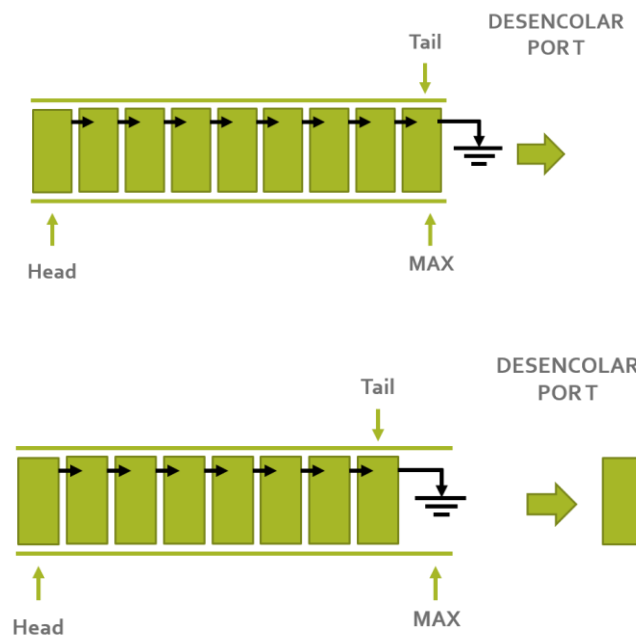
En una cola doble llena no es posible encolar más elementos, ni por HEAD ni por TAIL.



En una cola doble llena sí se pueden desencolar elementos tanto por HEAD como por TAIL. Cuando se desencola por el inicio de la estructura se obtiene el elemento al que hace referencia HEAD y esta referencia se recorre al siguiente elemento (sucesor).

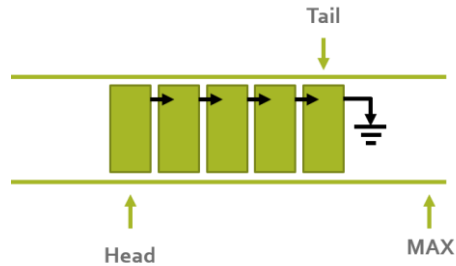


Cuando se desencola por el final de la estructura se obtiene el elemento al que hace referencia TAIL y esta referencia se recorre al elemento anterior (predecesor).

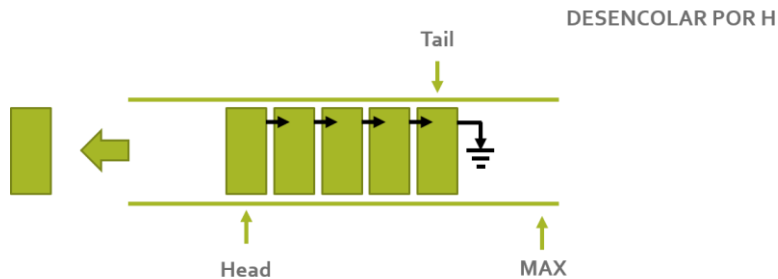
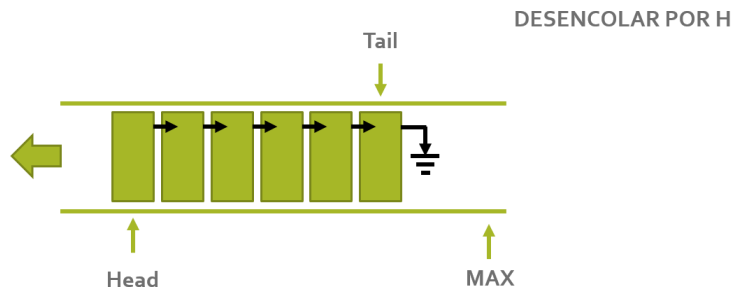


### Cola doble con elementos

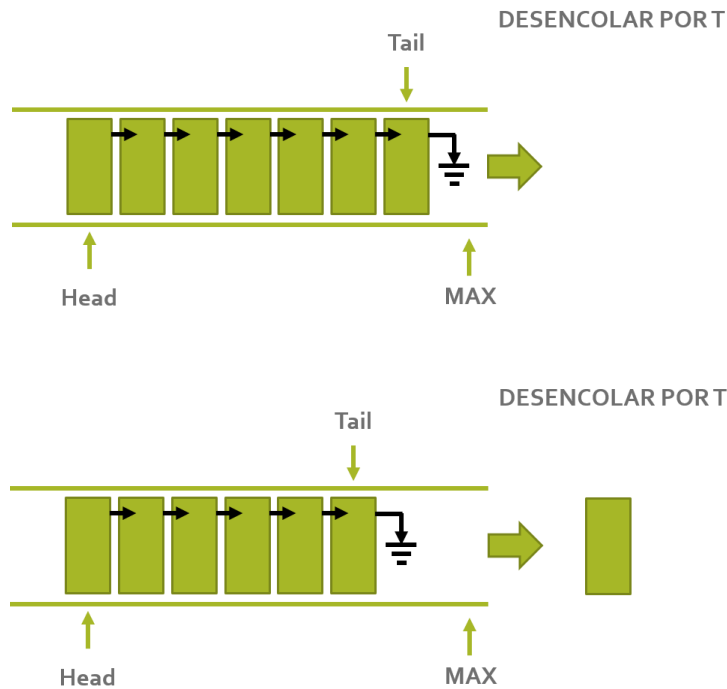
Una cola doble que contiene elementos (sin llegar a su máximo tamaño) representa el caso general de la estructura.



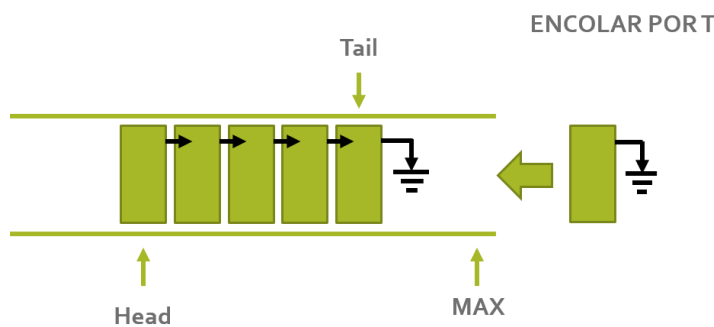
En una cola doble con elementos es posible desencolar nodos, tanto por HEAD como por TAIL. Cuando se desencola por el inicio de la estructura, se debe recorrer la referencia al inicio de la cola (HEAD) al siguiente elemento de la estructura (sucesor).



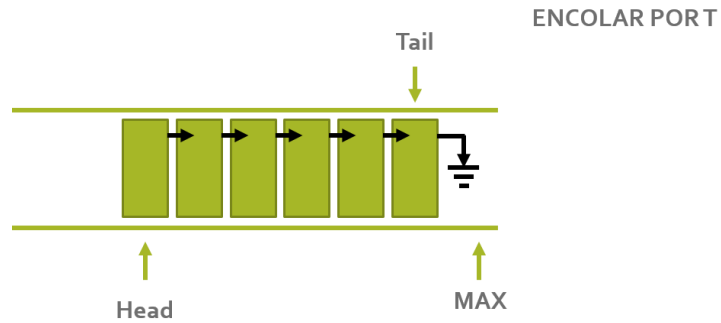
Cuando se desencola por el final de la estructura, se debe recorrer la referencia al final de la cola (TAIL) al elemento anterior de la estructura (predecesor).



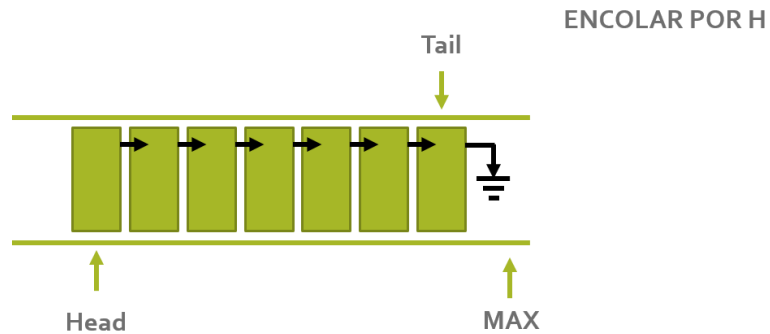
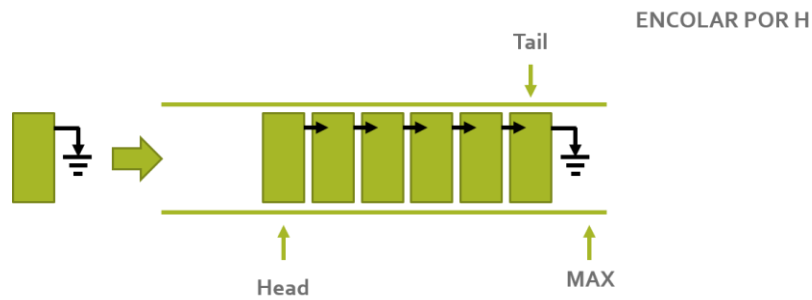
Así mismo, se pueden encolar elementos en una cola doble mientras no se exceda la capacidad máxima de la estructura. Es posible encolar elementos tanto por HEAD como por TAIL. Cuando se encola un elemento por el final, el nodo al que apunta TAIL tiene como sucesor el nuevo nodo y la referencia a TAIL apunta al nuevo elemento.







Cuando se encola un elemento por el inicio, el nodo al que apunta HEAD tiene como predecesor el nuevo nodo y la referencia a HEAD apunta al nuevo elemento.



## Aplicación

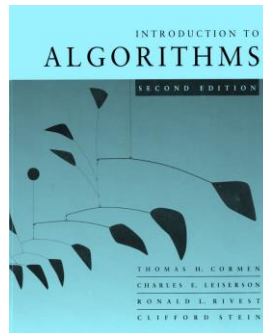
Una cola doble en la vida cotidiana podría verse como un abuso, es decir, formar a personas delante de la fila, a pesar de haber llegado después de los que ya están formados (y sin embargo pasa). Sin embargo, en las ciencias de la computación hay muchas aplicaciones que trabajan así, ya que hay procesos que tienen prioridad y deben ser ejecutados antes que otros procesos menos importantes.

Dentro del sistema operativo no todas las aplicaciones tienen la misma exigencia en cuanto a tiempo y recursos, existen procesos que se tienen que ejecutar de manera inmediata ante algún suceso que se presente en el sistema, mientras que otros solo tengan que procesar información y puedan (y deban) esperar a que el sistema se recupere.

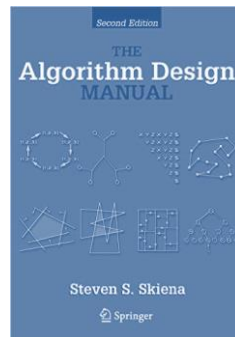
Todo sistema operativo define un valor de urgencia con que debe ejecutarse una aplicación, es decir, define la prioridad que tiene un proceso frente a otros que se estén ejecutando en el sistema.

La prioridad se expresa como un número entero. Por tanto, si un proceso A tiene asignada una prioridad  $PR_x$  y un proceso B tiene asignada una prioridad  $PR_y$ , si  $PR_x > PR_y$ , el proceso A será más prioritario que el proceso B y, por ende, será el que se ejecute primero. Por otro lado, si existen varios procesos en ejecución con prioridad  $PR_y$ , si llega un proceso con mayor prioridad, por ejemplo,  $PR_x$ , éste último se ejecutará primero, es decir, no se encola al final si no al inicio del conjunto.

## Bibliografía



Introduction to Algorithms. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, McGraw-Hill.



The Algorithm Design Manual. Steven S. Skiena, Springer.

CAPCOM (2012). CAPCOM: RESIDENT EVIL 6 | Manual web oficial [Figura 1]. Consulta: Enero de 2016. Disponible en: <http://game.capcom.com/manual/bio6/es/page-74.html>

SergioGameplayer (2014). FIFA World Cup Brazil 2014 - Juego Completo Menús, Modos de Juego Equipos Uniformes y mas! [Figura 2]. Consulta: Enero de 2016. Disponible en: <https://www.youtube.com/watch?v=Pyu0Xp7MVJI> 9