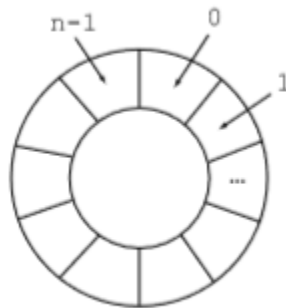


## Colas circulares

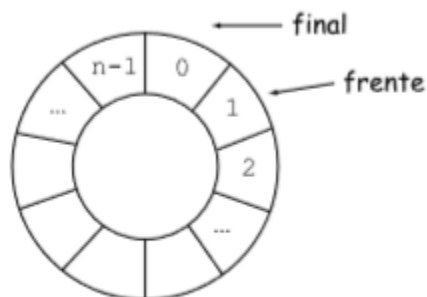
En las que el ultimo elemento y primero están unidos ya que están en un formato circular.

La alternativa, sugerida en la operación de quitar un elemento, de desplazar los restantes elementos del array de modo que la cabeza de la cola vuelva al principio del array, es costosa en términos de tiempo de computadora, especialmente si los datos almacenados en el array son estructuras de datos grandes. La forma más eficiente de almacenar una cola en un array es modelarlo de tal forma que se una el extremo final con el extremo cabeza. Tal array se denomina array circular y permite que la totalidad de sus posiciones se utilicen para almacenar elementos de la cola sin necesidad de desplazar elementos. La Figura 10.4 muestra un array circular de  $n$  elementos.



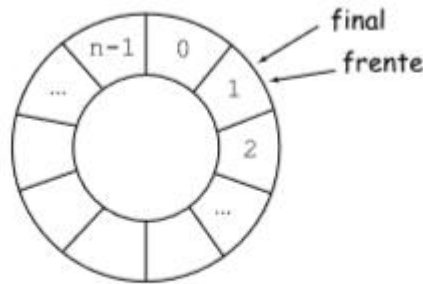
**Figura 10.4** Un array circular

El array se almacena de modo natural en la memoria como un bloque lineal de  $n$  elementos. Se necesitan dos marcadores (apuntadores) frente y fin para indicar, respectivamente, la posición del elemento cabeza y del último elemento puesto en la cola.



**Figura 10.5** Una cola vacía

El frente siempre contiene la posición del primer elemento de la cola y avanza en el sentido de las agujas del reloj; fin contiene la posición donde se puso el último elemento y también avanza en el sentido del reloj (circularmente a la derecha). La implementación del movimiento circular se realiza según la teoría de los restos, de tal forma que se generen índices de 0 a MAXTAMQ-1:



**Figura 10.6** Una cola que contiene un elemento

Los algoritmos que formalizan la gestión de colas en un array circular han de incluir las operaciones básicas del TAD Cola, en concreto, las siguientes tareas básicas:

- Creación de una cola vacía, de tal forma que fin apunte a una posición inmediatamente anterior a frente:

frente = 0; fin = MAXTAMQ-1.

- Comprobar si una cola está vacía:

frente == siguiente(fin)

- Comprobar si una cola está llena. Para diferenciar la condición de cola llena de cola vacía se sacrifica una posición del array, de tal forma que la capacidad de la cola va a ser MAXTAMQ-1.

La condición de cola llena es:

frente == siguiente(siguiente(fin))

- Poner un elemento a la cola: si la cola no está llena, avanzar fin a la siguiente posición,

fin = (fin + 1) % MAXTAMQ, y asignar el elemento.

- Retirar un elemento de la cola: si la cola no está vacía, quitarlo de la posición frente y avanzar frente a la siguiente posición: (frente + 1) % MAXTAMQ.

- Obtener el elemento primero de la cola, si la cola no está vacía, sin suprimirlo de la cola.