



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* M.I. Edgar Tista García

*Asignatura:* Estructura de Datos y Algoritmos I

*Grupo:* 1

*No de Práctica(s):* 6

*Integrante(s):* Díaz Hernández Marcos Bryan

*No. de Equipo de  
cómputo empleado:*

*No. de Lista o Brigada:* 9

*Semestre:* 2020-2

*Fecha de entrega:* 23 de mayo del 2020

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

## Objetivo de la práctica

Revisar las definiciones, características, procedimientos y ejemplos de las estructuras lineales Cola circular y Cola doble, con la finalidad de comprender sus estructuras y poder implementarlas.

## Introducción

Este reporte esta elaborado para poder comprender los elementos de las colas, mediante diagramas, explicaciones, todo con el fin de que sea más comprensible el tema.

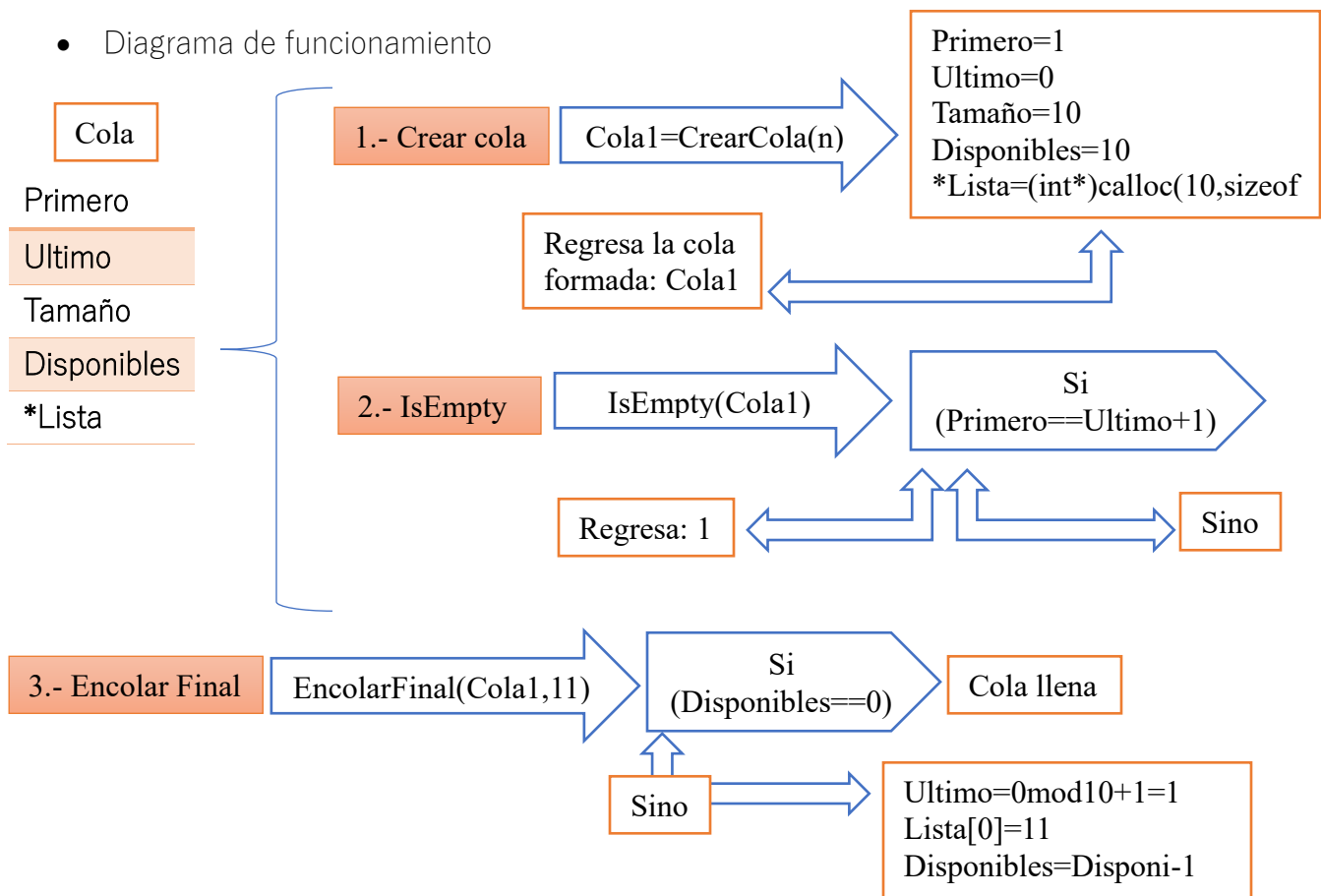
## Ejercicios de la guía de laboratorio

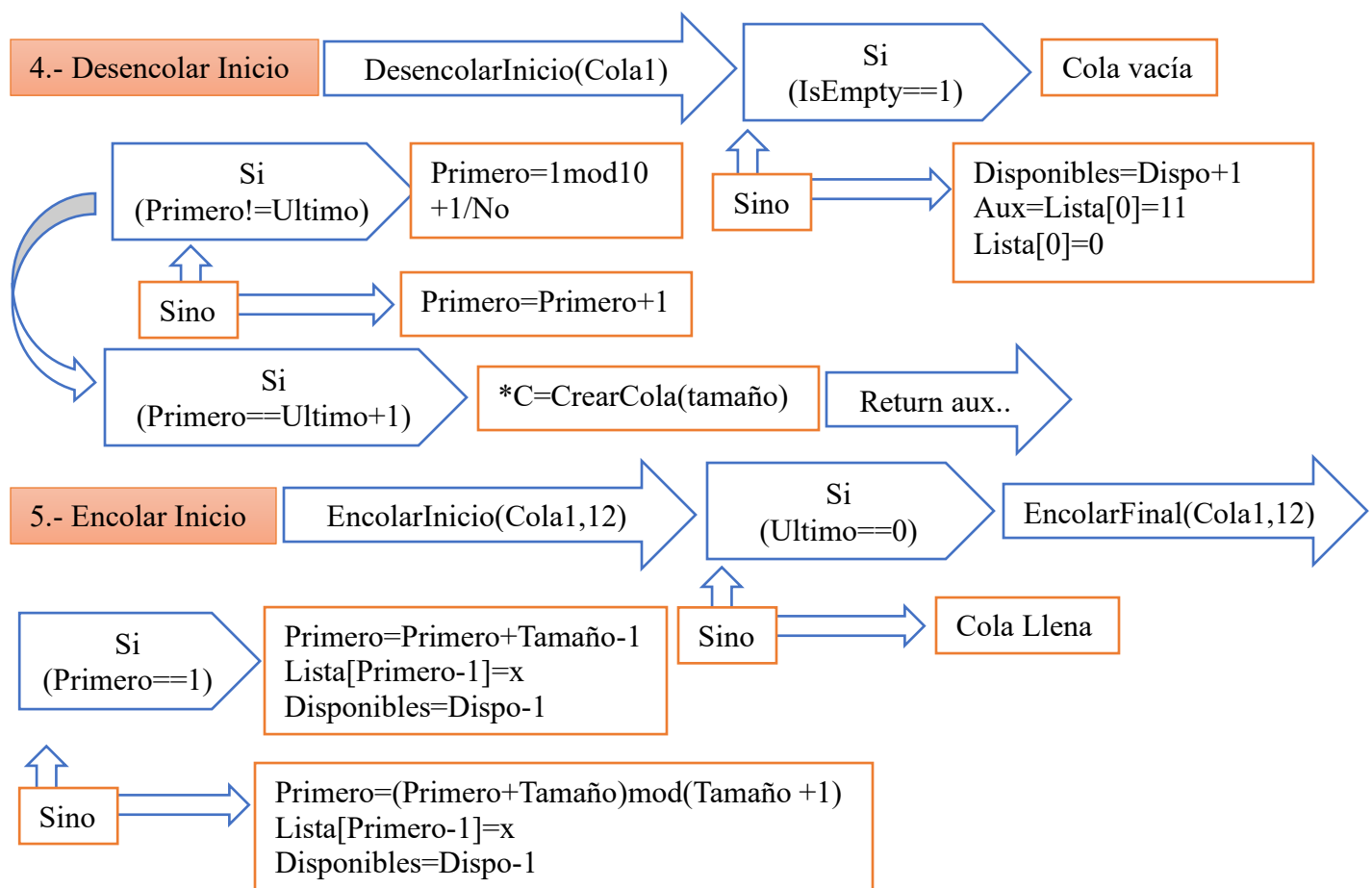
En esta ocasión no hubo ejercicios que elaborar en la guía del laboratorio.

## Ejercicios del Laboratorio

- Ejercicio 1

El primer ejercicio indicaba el codificar y compilar las instrucciones indicadas y posteriormente realizar un análisis del funcionamiento del programa, por otro lado, se indicaba el uso de una biblioteca que tenia las funciones para poder trabajar con una cola doble circular.





El programa crea una cola circular doble donde los índices comienzan en uno y cero, la principal característica de la cola circular es la posibilidad de poder agregar elementos al comienzo de la fila, y al ser doble se puede igual añadir y eliminar en ambos extremos de una cola. En base a lo anterior el programa inserta elemento al final de la cola, y en ese instante la cola está vacía y el ultimo es el único que se mueve posteriormente, se eliminan estos valores ya que al desencolar al final se regresan los índices a sus posiciones iniciales. Por ultimo los valores que añaden al final y principio de la cola, lo principal es que el añadir al principio o desencolar afecta al primero, y hacerlo al final afecta al final.

b) Las operaciones que contienen modulo:

- $\text{Primero} = (\text{Primero} + \text{Tamaño}) \bmod (\text{Tamaño} + 1)$
- $\text{Primero} = 1 \bmod 10 + 1$
- $\text{Ultimo} = 0 \bmod 10 + 1 = 1$

Se utiliza el modulo para poder obtener la posición del índice, por medio de la referencia del tamaño de la lista, mediante la teoría de restos que el operador modulo usa, y con esta se puede determinar la posición siguiente de cada índice, en este caso todas las operaciones donde se usa indican un avance en el índice.

- Relación con teoría.

En cuanto a la parte teórica del ejercicio, vienen los conceptos vistos en clase presencial y en clase virtual, además de que gracias al repaso ha quedado mucho mas claro el funcionamiento de las distintas funciones que utiliza una cola doble circular.

The screenshot shows a C++ IDE with the following components:

- Source Code (Primer Ejercicio.c):**

```

1 void main(){
2     int longitud;
3     printf("Ingresa el tamaño\n");
4     scanf("%d",&longitud);
5     Cola cola=crearCola(longitud);
6     encolarFinal(&cola,11);
7     encolarFinal(&cola,12);
8     mostrarValores(cola);
9     mostrarIndices(cola);
10    printf("Primera ejecucion \n");
11    desencolarInicio(&cola);
12    desencolarInicio(&cola);
13    mostrarValores(cola);
14    mostrarIndices(cola);
15    printf("Segunda ejecucion \n");
16    encolarFinal(&cola,13);
17    encolarFinal(&cola,14);
18    encolarInicio(&cola,9);
19    encolarInicio(&cola,10);
20    mostrarValores(cola);
21    mostrarIndices(cola);
22    printf("Tercera ejecucion\n");
23    liberar(&cola);
24    system("PAUSE");
25 }

```
- Compilation Results:**

```

Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Brain\Documents\EDA\P6\Primer Ejercicio.exe
- Output Size: 131.697265625 KiB
- Compilation Time: 0.42s

```
- Execution Output:**

```

Ingresa el tamaño
10
Posicion 1 valor 11
Posicion 2 valor 12
Posicion 3 valor 0
Posicion 4 valor 0
Posicion 5 valor 0
Posicion 6 valor 0
Posicion 7 valor 0
Posicion 8 valor 0
Posicion 9 valor 0
Posicion 10 valor 0
Primero = 1
Ultimo = 2
Disponibles = 8
Primera ejecucion
Posicion 1 valor 0
Posicion 2 valor 0
Posicion 3 valor 0
Posicion 4 valor 0
Posicion 5 valor 0
Posicion 6 valor 0
Posicion 7 valor 0
Posicion 8 valor 0
Posicion 9 valor 0
Posicion 10 valor 0
Primero = 1
Ultimo = 0
Disponibles = 10
Segunda ejecucion
Posicion 1 valor 13
Posicion 2 valor 14
Posicion 3 valor 0
Posicion 4 valor 0
Posicion 5 valor 0
Posicion 6 valor 0
Posicion 7 valor 0
Posicion 8 valor 0
Posicion 9 valor 10
Posicion 10 valor 9
Primero = 9
Ultimo = 2

```

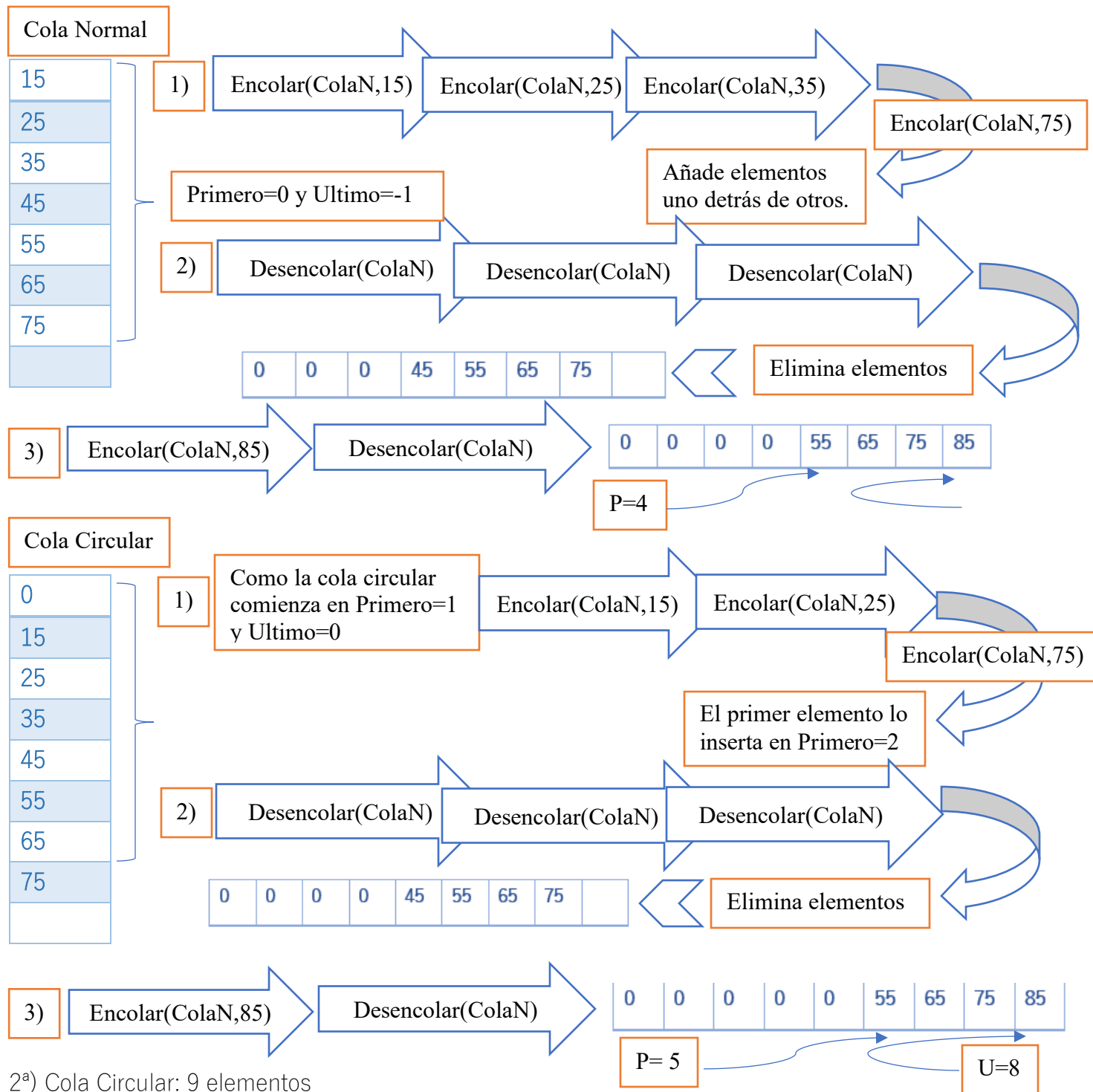
## • Ejercicio 2

El segundo ejercicio consistía en realizar las operaciones de una cola normal: en una cola normal, en una cola circular de 9 elementos y en una cola de 5 elementos. Lo principal fue el crear las instrucciones de una cola normal en la biblioteca, y ejecutar para cada caso el programa, además de responder las distintas preguntas para el caso de las colas circulares, que tiene una interacción distinta con las operaciones normales de una cola.

- ¿Cómo lo resolví?

En la práctica anterior se habían utilizado listas, lo único que hice fue el utilizar las instrucciones y verificar que estuviesen bien, las agregue a la biblioteca, y posteriormente separe las creaciones de las colas, para poder ver la diferencia entre los valores de una cola normal y la de la cola circular. De esa forma pude comprobar las diferencias entre las operaciones de cada tipo de cola.

- Diagrama de funcionamiento.



2ª) Cola Circular: 9 elementos

- ¿Cuál es el contenido de la estructura?

0	0	0	0	0	55	65	75	85
---	---	---	---	---	----	----	----	----

- ¿Cuántos elementos vacíos o disponibles hay en la estructura?: Quedan 5 elementos vacíos.
- ¿Cuál es el valor del índice "Primero"?: Su valor es igual a 5
- ¿Cuál es el valor del índice "Ultimo"?: Su valor es igual a 8

La diferencia entre la cola normal y la cola circular es la posición de los índices porque en la normal comienzan:  $P=0$ ,  $U=-1$ , y en la circular:  $P=1$ ,  $U=0$ . Por esto las operaciones de la cola normal se ven distintas en la cola circular, y comienzan saltándose una posición por le  $P=1$ . Además el contador de elementos disponibles está asociada a un cambio en los espacios disponibles y como las operaciones normales normalmente no contienen operaciones de comprobación dentro de las funciones, este contador permanece constante.

## 2<sup>B</sup>) Cola Circular: 5 elementos

- ¿Cuál es el contenido de la estructura?

0	0	0	0	0
---	---	---	---	---

- ¿Cuántos elementos vacíos o disponibles hay en la estructura?: Son cinco elementos vacíos.
- ¿Cuál es el valor del índice “Primero”? Es igual a 5
- ¿Cuál es el valor del índice “Ultimo”? Es igual a 8

Como se mencionaba anteriormente se comprueban las diferencias entre el uso de los índices, en base a la cola, además que las operaciones de inserción no tienen un controlador o límite de inserción, porque el contador del índice ultimo en este caso se continuaba sumando al índice y por eso llego a 8.

- Relación con la teoría

Lo principal es la diferencia entre las colas circulares y su funcionamiento, de ahí que lo visto en clase virtual se vuelva importante para reconocer los elementos característicos de cada tipo de cola.

```

1 #include "colaDoble.h"
2
3 void main()
4 {
5     int tamaño;
6     printf("Inserte el tamaño de la lista\n");
7     scanf("%d", &tamaño);
8     Cola Cola1=crearcola(tamaño);
9     encolar(&Cola1,15);
10    encolar(&Cola1,25);
11    encolar(&Cola1,35);
12    encolar(&Cola1,45);
13    encolar(&Cola1,55);
14    encolar(&Cola1,65);
15    encolar(&Cola1,75);
16    mostrarValores(Cola1);
17    printf("\n");
18    desencolar (&Cola1);
19    desencolar (&Cola1);
20    desencolar (&Cola1);
21    mostrarValores(Cola1);
22    printf("\n");
23    encolar(&Cola1,85);
24    desencolar (&Cola1);
25    mostrarValores(Cola1);
26    liberar(&Cola1);
27    system("pause");
28 }

```

```

Inserte el tamaño de la lista
10
Posición 1      valor 15
Posición 2      valor 25
Posición 3      valor 35
Posición 4      valor 45
Posición 5      valor 55
Posición 6      valor 65
Posición 7      valor 75
Posición 8      valor 0
Posición 9      valor 0
Posición 10     valor 0

Posición 1      valor 0
Posición 2      valor 0
Posición 3      valor 0
Posición 4      valor 45
Posición 5      valor 55
Posición 6      valor 65
Posición 7      valor 75
Posición 8      valor 0
Posición 9      valor 0
Posición 10     valor 0

Posición 1      valor 0
Posición 2      valor 0
Posición 3      valor 0
Posición 4      valor 0
Posición 5      valor 55
Posición 6      valor 65
Posición 7      valor 75
Posición 8      valor 85
Posición 9      valor 0
Posición 10     valor 0
Presione una tecla para continuar . . .

```

Cola Normal

```

#include "colaDoble.h"
void main()
{
    int tamaño;
    printf("Inserte el tamaño de la lista\n");
    scanf("%d",<td>tamaño);
    Cola Cola1=crearCola(tamaño);
    mostrarIndice<td>s(Cola1);
    encolar(<td>Cola1,15);
    encolar(<td>Cola1,25);
    encolar(<td>Cola1,35);
    encolar(<td>Cola1,45);
    encolar(<td>Cola1,55);
    encolar(<td>Cola1,65);
    encolar(<td>Cola1,75);
    mostrarValore<td>s(Cola1);
    mostrarIndice<td>s(Cola1);
    printf("\n");
    desencolar (<td>Cola1);
    desencolar (<td>Cola1);
    desencolar (<td>Cola1);
    mostrarValore<td>s(Cola1);
    mostrarIndice<td>s(Cola1);
    printf("\n");
    encolar(<td>Cola1,85);
}
  
```

```

Inserte el tamaño de la lista
9
Primero = 1
Ultimo = 0
Disponibles = 9
Posición 1    valor 0
Posición 2    valor 15
Posición 3    valor 25
Posición 4    valor 35
Posición 5    valor 45
Posición 6    valor 55
Posición 7    valor 65
Posición 8    valor 75
Posición 9    valor 0
Primero = 1
Ultimo = 7
Disponibles = 9
Posición 1    valor 0
Posición 2    valor 0
Posición 3    valor 0
Posición 4    valor 0
Posición 5    valor 45
Posición 6    valor 55
Posición 7    valor 65
Posición 8    valor 75
Posición 9    valor 0
Primero = 4
Ultimo = 7
Disponibles = 9
Posición 1    valor 0
Posición 2    valor 0
Posición 3    valor 0
Posición 4    valor 0
Posición 5    valor 55
Posición 6    valor 65
Posición 7    valor 75
Posición 8    valor 85
Posición 9    valor 0
Primero = 5
Ultimo = 8
Disponibles = 9
  
```

Cola Circular de 9 elementos

```

#include "colaDoble.h"
void main()
{
    int tamaño;
    printf("Inserte el tamaño de la lista\n");
    scanf("%d",<td>tamaño);
    Cola Cola1=crearCola(tamaño);
    mostrarIndice<td>s(Cola1);
    encolar(<td>Cola1,15);
    encolar(<td>Cola1,25);
    encolar(<td>Cola1,35);
    encolar(<td>Cola1,45);
    encolar(<td>Cola1,55);
    encolar(<td>Cola1,65);
    encolar(<td>Cola1,75);
    mostrarValore<td>s(Cola1);
    mostrarIndice<td>s(Cola1);
    printf("\n");
    desencolar (<td>Cola1);
    desencolar (<td>Cola1);
    desencolar (<td>Cola1);
    mostrarValore<td>s(Cola1);
    mostrarIndice<td>s(Cola1);
    printf("\n");
    encolar(<td>Cola1,85);
    desencolar (<td>Cola1);
    mostrarValore<td>s(Cola1);
    mostrarIndice<td>s(Cola1);
    liberar(<td>Cola1);
}
  
```

```

Inserte el tamaño de la lista
5
Primero = 1
Ultimo = 0
Disponibles = 5
Posición 1    valor 0
Posición 2    valor 15
Posición 3    valor 25
Posición 4    valor 35
Posición 5    valor 45
Primero = 1
Ultimo = 7
Disponibles = 5
Posición 1    valor 0
Posición 2    valor 0
Posición 3    valor 0
Posición 4    valor 0
Posición 5    valor 45
Primero = 4
Ultimo = 7
Disponibles = 5
Posición 1    valor 0
Posición 2    valor 0
Posición 3    valor 0
Posición 4    valor 0
Posición 5    valor 0
Primero = 5
Ultimo = 8
Disponibles = 5
Process exited after 2.955 seconds with return value 3221226356
  
```

Cola Circular de 5 elementos

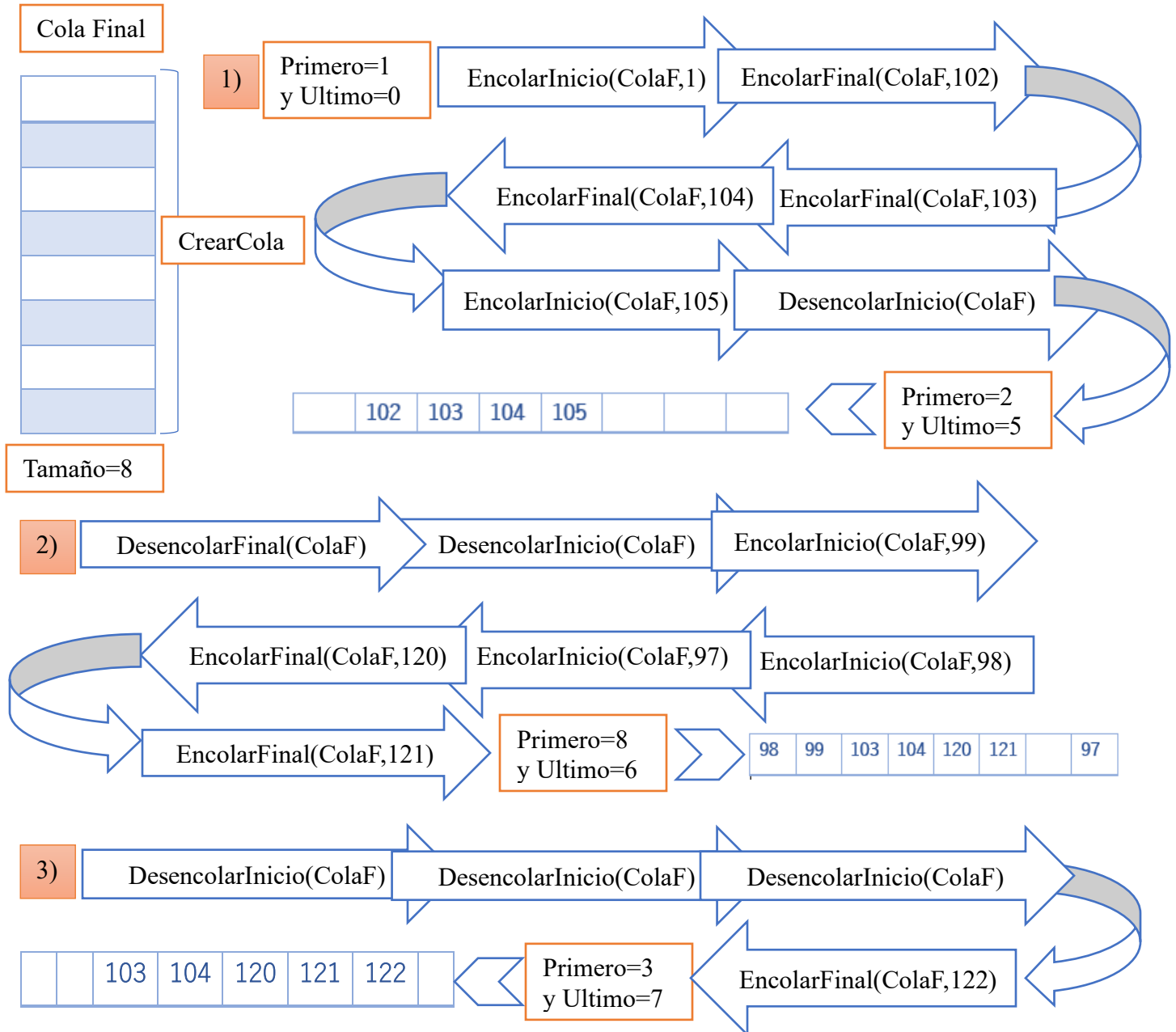
### • Ejercicio 3

El último ejercicio consistía en utilizar las operaciones de la cola circular doble para llegar a los esquemas propuestos, y por ultimo obtener una cola de elementos en determinado orden. La principal condición es utilizar funciones que modifiquen por Inicio y Final la cola.

- ¿Cómo lo resolví?

Como los ejercicios de las actividades, fui elaborando primero un esquema de como iban a ir las operaciones y posteriormente comencé a escribir el código, verificando que ninguna parte diera un resultado no esperado, además de ir clasificando los estados para tener claro las partes que se tenían que conservar y cuales conservar.

- Diagrama de funcionamiento



Este diagrama resume lo que realice en el ejercicio, la verdad, me ayudo bastante el entender el ejercicio de la actividad virtual, con eso pude hacer la practica sin problemas.



- Relación con teoría.

La teoría en vuelta en este ejercicio es la misma que se vio en las clases virtuales de la ultima semana y me parece que fue bastante agradable poder realizar la practica con fluidez, además de que recordé algunas cosas de otras prácticas anteriores a si que se siente bien el poder y volver a escribir código.

The screenshot shows a C++ IDE with the following components:

- Editor:** Contains C++ code for a double-ended queue. The code defines two states, `EstadoD` and `EstadoT`, which manage a circular queue. It includes functions for adding and removing elements from both ends and for displaying the queue's state.
- Output Window:** Displays the execution results, showing the state of the queue at different points in the program. It includes labels like "Primer Diagrama", "Segundo Diagrama", and "Tercer Diagrama", followed by a list of positions and their values.
- Compilation Results:** Shows the output of the compiler, indicating that the program compiled successfully with no errors or warnings.

```
void EstadoD(Cola *Cola1){
    printf("Segundo Diagrama\n");
    desencolarFinal(Cola1);
    desencolarInicio(Cola1);
    encolarInicio(Cola1,99);
    encolarInicio(Cola1,98);
    encolarInicio(Cola1,97);
    encolarFinal(Cola1,120);
    encolarFinal(Cola1,121);
    mostrarValores(*Cola1);
    mostrarIndices(*Cola1);
    printf("\n");
}

void EstadoT(Cola *Cola1){
    printf("Tercer Diagrama\n");
    desencolarInicio(Cola1);
    desencolarInicio(Cola1);
    desencolarInicio(Cola1);
    encolarFinal(Cola1,122);
    mostrarValores(*Cola1);
    mostrarIndices(*Cola1);
    printf("\n");
}
```

Primer Diagrama

Posicion	valor
Posicion 1	valor 0
Posicion 2	valor 102
Posicion 3	valor 103
Posicion 4	valor 104
Posicion 5	valor 105
Posicion 6	valor 0
Posicion 7	valor 0
Posicion 8	valor 0

Primero = 2  
Ultimo = 5  
Disponibles = 4

Segundo Diagrama

Posicion	valor
Posicion 1	valor 98
Posicion 2	valor 99
Posicion 3	valor 103
Posicion 4	valor 104
Posicion 5	valor 120
Posicion 6	valor 121
Posicion 7	valor 0
Posicion 8	valor 97

Primero = 8  
Ultimo = 6  
Disponibles = 1

Tercer Diagrama

Posicion	valor
Posicion 1	valor 0
Posicion 2	valor 0
Posicion 3	valor 103
Posicion 4	valor 104
Posicion 5	valor 120
Posicion 6	valor 121
Posicion 7	valor 122
Posicion 8	valor 0

Primero = 3  
Ultimo = 7  
Disponibles = 3

Compilation results...

```
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Brain\Documents\EDA\P6\E3.exe
- Output Size: 132.32421875 KiB
- Compilation Time: 0.69s
```

## Conclusiones

En esta practica he reforzado lo aprendido en clase, he añadido conocimiento hacer de las colas circulares dobles, y aprendí al utilizarlas de forma más efectiva, además reconocí elementos clave de cada tipo de cola, lo que me ayudo a poder continuar con la practica sin problemas. E incluso me sentí bien haciendo la practica o volviendo a realizar código, ya que me mantiene en un estado activo. Para finalizar he aprendido las principales características de las colas, y aprendí el cómo utilizarlas.