



## Ejercicios de pilas y colas

### OBJETIVOS

El objetivo de estos ejercicios es repasar los conceptos sobre el TAD pilas y sobre el TAD colas.

En la página de UAMX de la asignatura dispones del esqueleto del código para poder trabajar con estos ejercicios. El examen estará formado por dos ejercicios similares a estos.

### Ejercicio NumeroDeElementosPila

Implementar una función denominada NumeroDeElementosPila que reciba como argumento una pila. Esta función debe devolver el número de elementos de la pila.

```
int NumeroDeElementosPila (Pila P);
```

Ejemplo de ejecución: Tras llamar con P -> 4 -> 3 -> 2 -> 3, la función debe devolver 4.

### Ejercicio EliminaMayoresPila

Implementar una función denominada EliminaMayoresPila que reciba como argumento una pila y un número. Esta función debe eliminar en la pila todos los números que sean mayores que el indicado en el argumento.

```
void EliminaMayoresPila (Pila *P, int num);
```

Ejemplo de ejecución: Tras llamar con P -> 4 -> 3 -> 2 -> 3 y el número 2, la pila resultante es P -> 2.

### Ejercicio DaVueltaPila

Crear una función DaVueltaPila que reciba como parámetro una pila y que devuelva la pila invertida.

```
Pila DaVueltaPila (Pila P);
```

Ejemplo de ejecución: Si llamas a DaVueltaPila con una pila que contenga los elementos 3 -> 3 -> 5 debe devolver una pila con los elementos 5 -> 3 -> 3.

### Ejercicio SonIgualesPilas

Crear una función SonIgualesPilas que reciba como parámetros dos pilas. Debe comprobar que son iguales, devolviendo el valor 1 o 0 si son distintas.

```
int SonIgualesPilas(Pila P1, Pila P2);
```

Ejemplo de ejecución: Tras llamar al programa con dos pilas que contengan los números 11, 5 y 9 debe devolver un 1.



## Ejercicio MayorPila

Realizar una función denominada MayorPila que reciba como argumento una pila y devuelva el mayor número incluido en la pila.

```
int MayorPila (Pila P);
```

Ejemplo de ejecución: Tras llamar P -> 4 -> 3 -> 5 debe devolver el número 5.

## Ejercicio MenorPila

Realizar una función denominada MenorPila que reciba como argumento una pila y devuelva el menor número incluido en la pila.

```
int MenorPila(Pila P);
```

Ejemplo de ejecución: Tras llamar P-> 4 -> 3 -> 5 la función debe devolver 3.

## Ejercicio NumeroDeElementosCola

Implementar una función denominada NumeroDeElementosCola que reciba como argumento una cola. Esta función debe devolver el número de elementos de la cola.

```
int NumeroDeElementosCola (Cola *C);
```

Ejemplo de ejecución: Tras llamar C -> 4 -> 3 -> 2 -> 3, la función debe devolver 4.

## Ejercicio EliminaMayoresCola

Implementar una función denominada EliminaMayoresCola que reciba como argumento una cola y un número. Esta función debe eliminar en la cola todos los números que sean mayores que el indicado en el argumento.

```
void EliminaMayoresCola (Cola *C, int num);
```

Ejemplo de ejecución: Tras llamar C -> 4 -> 3 -> 2 -> 3 y el número 2, la cola resultante es C->2.

## Ejercicio SonIgualesColas

Crear una función SonIgualesColas que reciba como parámetros dos colas. Debe comprobar que son iguales, devolviendo el valor 1 o 0 si son distintas.

```
int SonIgualesColas(Cola *C1, Cola *C2);
```

Ejemplo de ejecución: Tras llamar al programa con dos colas que contengan los números 11, 5 y 9 debe devolver un 1.

## Ejercicio DaVueltaCola

Crear una función DaVueltaCola que reciba como parámetro una cola y que devuelva la cola invertida.

```
Cola *DaVueltaCola (Cola *C);
```



Ejemplo de ejecución: Si llamas a DaVueltaCola con una cola que contenga los elementos 3 -> 3 -> 5 debe devolver una cola con los elementos 5 -> 3 -> 3.

## Ejercicio MayorCola

Realizar una función denominada MayorCola que reciba como argumento una cola y devuelva el mayor número incluido en la cola.

```
int MayorCola (Cola *C);
```

Ejemplo de ejecución: Tras llamar C -> 4 -> 3 -> 5 debe devolver el número 5.

## Ejercicio MenorCola

Realizar una función denominada MenorCola que reciba como argumento una cola y devuelva el menor número incluido en la cola.

```
int MenorCola(Cola *C);
```

Ejemplo de ejecución: Tras llamar C-> 4 -> 3 -> 5 la función debe devolver 3.