



**Certified Tech  
Developer**

The Ultimate Degree

## Examen final de Programación Imperativa

¡Llegó el momento de poner a prueba todo lo que estuvimos viendo a lo largo de estas semanas!

### Metodología de evaluación

Se evaluarán los siguientes conceptos sobre el código entregado:

- **FORMA**
  - Que el código esté prolijo e implemente buenas prácticas
  - Que las variables, métodos y funciones tengan nombres descriptivos
  - Que utilices nombres en español o en inglés pero no ambos
  - Que utilices camelCase donde corresponda
- **LÓGICA**
  - Que la lógica corresponda con lo que solicitan las consignas
  - Que utilices los métodos más adecuados para cada caso
- **FUNCIONAMIENTO**
  - Que el código funcione correctamente, sin arrojar errores
  - Que el código produzca el resultado esperado a partir de los datos suministrados

## Duración, formato y entrega

El examen tendrá una duración de 100 minutos. La hora de finalización será informada por el docente. ⚠️ **Las entregas realizadas después del tiempo estipulado no serán tenidas en cuenta.**

Al terminar el examen, deben entregarlo en formato **.txt** haciendo uso del formulario que les enviará el docente por chat de zoom.

⚠️ **Recuerden verificar la recepción del formulario con el/la docente antes de retirarse del zoom** ⚠️

## Algunos consejos 🧐💡

- Es normal tener nervios en esta etapa. Recordá que practicamos mucho estas semanas y tratá de mantener la mayor calma que puedas.
- Lee todas las consignas antes de empezar el examen para tener una idea general de lo que se pide.
- Si alguna consigna no queda clara, recordá que podés pedir aclaración por privado al docente.
- Si una consigna te bloquea demasiado, pasá a la siguiente, lo más probable es que luego se te ocurra una solución.



## Ejercicio 1

Dado el siguiente array de países

```
let paises = [  
  {  
    nombre: "Argentina",  
    continente: "Sudamérica",  
    poblacion: 40000000  
  },  
  {  
    nombre: "Colombia",  
    continente: "Sudamérica",  
    poblacion: 50372000  
  },  
  {  
    nombre: "Brasil",  
    continente: "Sudamérica",  
    poblacion: 300000000  
  },  
  {  
    nombre: "Etiopía",  
    continente: "África",  
    poblacion: 15000000  
  },  
  {  
    nombre: "Chile",  
    continente: "Sudamérica",  
    poblacion: 10000000  
  }  
];
```

- Crear una función que reciba como parámetro el arreglo de objetos, un continente, y un número de población.

La misma debe retornar un **nuevo arreglo** solo con los países que sean del continente pasado por parámetro, y además, que su población sea mayor o igual a la del parámetro.

Si no encuentra coincidencias deberá retornar un array vacío



## Ejercicio 2

Dado el siguiente array de numeros :

```
let arrayDeNumeros = [1,2,3, 8, 9, 104, 5, 6, 7, 15];
```

- Crear una función llamada bubbleSort que reciba como parámetros el array y un *string* que represente el sentido de ordenamiento, la función deberá realizar un ordenamiento interno del array según el orden indicado
  - "ASC" lo ordenará ascendente
  - "DESC" lo ordenará descendente

Para analizar el correcto funcionamiento imprimir por consola el array antes y después de ejecutar la función.



## Ejercicio 3

Dada la matriz

```
let matriz = [  
  [5, 2, 2], // 0  
  [2, 5, 2], // 1  
  [4, 4, 5]  // 2  
];
```

- Crear una función que reciba la matriz por parámetro, sume todos los valores de la fila 1 y **retorne** la suma.
- Crear una función que reciba la matriz generada por parámetro y **retorne** un nuevo array con todos los valores **pares** de la matriz.