

## TRABAJO PRÁCTICO 4

---

**Tema:** Funciones definidas por el usuario.

**Fecha de entrega:** 28 de septiembre de 2025  
mediante plataforma del CUV

### Objetivos:

Que el estudiante logre...

- Reconocer la importancia de las funciones definidas por el usuario y la diferencia entre prototipo de función, definición de función y convocatoria de función mediante su aplicación en ejercicios de programación.
- Comprender la vinculación entre parámetros formales y parámetros actuales en el momento de ejecución de una función.
- Aplicar las funciones definidas por el usuario en la resolución de problemas.

### Condiciones de presentación.

- Este trabajo práctico debe realizarse en forma individual.
- El código de los programas solicitados debe estar correctamente rotulado para su identificación, incluyendo comentarios de seguimiento y deberá ser enviado mediante la plataforma CUV.FCEYT hasta el día indicado como límite de presentación.
- La resolución completa de este trabajo práctico, incluyendo diagramas de flujo, deberá pasar a integrar la carpeta de práctica y autoevaluación del alumno.

**Criterios de evaluación y aprobación. Este trabajo práctico recibirá una calificación de aprobado o desaprobado.** Para aprobar la presentación debe cumplir como mínimo con los siguientes ítems:

- El trabajo práctico debe estar desarrollado completamente.
- La codificación en lenguaje C debe realizarse siguiendo las recomendaciones de la cátedra.
- La presentación del código de los enunciados solicitados deberá realizarse en tiempo y forma mediante la plataforma CUV.FCEYT: <http://cuv.unse.edu.ar>.

### Recursos Bibliográficos

- Introducción al Lenguaje C - Santos Espino.
- Introducción a la Programación con C – Marzal.
- Diapositivas, lecciones, disponibles en plataforma del CUV.

### Tareas a desarrollar para cada enunciado

- ❖ Desarrolle la diagramación de flujo y/o la codificación en Lenguaje C de los siguientes enunciados.
- ❖ Todas las entradas de datos deben ser verificadas.
- ❖ El funcionamiento del programa en cada caso debe implicar un experiencia agradable para el usuario (con mensajes adecuados, con facilidades para el ingreso de datos, etc.)
- ❖ La resolución de todos los enunciados deberá estar desarrollada en la carpeta de práctica de la asignatura.
- ❖ La entrega de los programas en lenguaje C de los enunciados indicados, deberán ser enviados mediante la plataforma CUV según las fechas y horarios de presentación de cada uno.

---

### Ejercicio 1. Estudiantes y promedio general.

Crear un programa que permita registrar varios estudiantes con los siguientes datos: nombre, edad y promedio.  
Se pide:

1. Cargar un arreglo de estudiantes.
2. Mostrar la información de todos los estudiantes usando funciones que reciban punteros.
3. Calcular y mostrar el promedio general de todos los estudiantes.

Ejemplo 1. Evaluar promedio con 3 estudiantes

Entrada: 3

Juan 20 8.5

Ana 22 9.0

Luis 19 7.5

Salida: Promedio general: 8.33

Ejemplo 2. Evaluar promedio con 2 estudiantes

Entrada: 2

Maria 21 6.0

Pedro 23 8.0

Salida: Promedio general: 7.00

### Ejercicio 2. Productos y precio total

Definir un programa que permita registrar productos (código, descripción y precio).

Se pide:

1. Cargar un arreglo de productos.
2. Mostrar todos los productos usando punteros.
3. Calcular y mostrar el precio total de todos los productos.

TRABAJO PRÁCTICO 4

---

Ejemplo 1. Evaluar total de 2 productos  
Entrada: 2  
101 Pan 50.0  
102 Leche 120.5  
Salida: Total: 170.50

Ejemplo 2. Evaluar total de 3 productos  
Entrada: 3  
201 Jugo 80.0  
202 Galletas 100.0  
203 Azucar 120.0  
Salida: Total: 300.00

**Ejercicio 3. Contactos y búsqueda por nombre.**

Crear un programa que gestione una agenda de contactos (nombre, teléfono y correo).

Se pide:

1. Cargar varios contactos.
2. Implementar una función que busque un contacto por nombre usando punteros.
3. Mostrar en pantalla el contacto encontrado o indicar que no existe.

Ejemplo 1. Buscar contacto existente  
Entrada: 2  
Juan 1234 juan@mail.com  
Ana 5678 ana@mail.com  
Ana  
Salida: Contacto: Ana 5678 ana@mail.com

Ejemplo 2. Buscar contacto inexistente  
Entrada: 1  
Luis 9999 luis@mail.com  
Pedro  
Salida: No existe

**Ejercicio 4. Libros y el más antiguo.**

Crear un programa para registrar libros (título, autor, año).

Se pide:

1. Cargar varios libros.
2. Implementar una función que reciba el arreglo y devuelva el libro más antiguo.
3. Mostrar en pantalla el libro más antiguo.

Ejemplo 1. Buscar libro mas antiguo  
Entrada: 2  
LibroA AutorA 2005  
LibroB AutorB 1999  
Salida: Mas antiguo: LibroB AutorB 1999

Ejemplo 2. Otro ejemplo  
Entrada: 3  
CienAnios Garcia 1967  
ElQuijote Cervantes 1605  
Rayuela Cortazar 1963  
Salida: Mas antiguo: ElQuijote Cervantes 1605

**Ejercicio 5. Vehículos y conteo posterior a 2015.**

Desarrollar un programa que permita registrar vehículos (marca, modelo, año).

Se pide:

1. Cargar una lista de vehículos.
2. Mostrar todos los vehículos usando punteros.
3. Contar y mostrar cuántos son posteriores al año 2015.

Ejemplo 1. Contar vehiculos posteriores  
Entrada: 3  
Toyota Corolla 2010  
Ford Focus 2018  
Honda Civic 2020  
Salida: Posteriores a 2015: 2

Ejemplo 2. Sin vehiculos posteriores  
Entrada: 2  
Fiat Uno 2000  
Chevrolet Corsa 2005  
Salida: Posteriores a 2015: 0

**Ejercicio 6. Jugadores y goleador.**

Construir un programa que registre jugadores de fútbol (nombre, equipo, goles).

Se pide:

1. Cargar un arreglo de jugadores.
2. Implementar una función que encuentre el jugador con más goles.
3. Mostrar en pantalla el goleador del torneo.

Ejemplo 1. Goleador con 3 jugadores  
Entrada: 3  
Messi PSG 30  
Ronaldo AlNassr 25  
Mbappe PSG 28  
Salida: Goleador: Messi PSG 30

Ejemplo 2. Otro torneo  
Entrada: 2  
Suarez Gremio 18  
Cavani Boca 10  
Salida: Goleador: Suarez Gremio 18