



# Algoritmos y Estructuras de Datos II

## Práctico para Repaso: Estructuras de datos simples. Algoritmos lineales. Funciones y Procedimientos. Arreglos. Archivos

### OBJETIVOS:

Que el alumno:

- Que el alumno profundice los conceptos ya aprendidos (algoritmo, programación, partes de un programa, instrucción, tipos de datos, estructuras de datos simples, funciones, procedimientos y arreglos).
- Que ahonde en la obtención de soluciones a problemas propuestos utilizando estructuras de datos: simples y arreglos, y enmarcar la solución del problema en forma estructurada utilizando funciones y procedimientos.

### METODOLOGÍA

- Lectura de los conceptos recordatorios de los temas de estructuras simples, procedimientos y funciones.
- El alumno deberá resolver individualmente los ejercicios propuestos.
- Se podrá realizar trabajos en grupos para consolidar conceptos, comprensión de lo solicitado y alternativas de solución.
- El alumno deberá codificar las soluciones en el lenguaje de programación propuesto en las clases prácticas de laboratorio.
- Interactuar en el aula virtual de la asignatura.

### DURACIÓN

Según planificación de la asignatura se deberán utilizar para la resolución de los ejercicios de la serie de repaso (número 0), no más de dos (2) clases prácticas.

### CONSIGNA:

Resolver la siguiente ejercitación, teniendo en cuenta que los ejercicios propuestos que siguen a continuación, se deberán resolver utilizando procedimientos o funciones según corresponda.

## EJERCICIOS PROPUESTOS

1. Escribir un programa que permita ingresar por teclado los siguientes datos correspondientes a los clientes de un banco: *Número de cuenta, nombre del cliente y saldo*. Se pide determinar la cantidad de cuentas con saldo positivo y la cantidad de cuentas con saldo negativo e informar al finalizar.
2. Crear un programa para calcular la remuneración semanal de los empleados de una fábrica. El valor de la hora laboral es de \$125 hasta 35 hs trabajadas, por encima de este valor se considera hora extra y el valor es de \$140.

Los datos que se ingresan son: apellido y nombre, y cantidad de horas trabajadas, y el programa debe permitir realizar el cálculo tantas veces como el usuario lo desee.

Se debe escribir una función **salarioSemanal** que reciba la cantidad de horas trabajadas y retorne el salario del trabajador.

3. Escribir un programa que permita ingresar una frase por teclado. El programa debe disponer de una función **fraseInvertida** que tome el valor de la frase ingresada y la muestre por pantalla en orden inverso.
4. Desarrollar un programa para calcular la suma de dígitos de un número entero de dos dígitos o más. El programa debe permitir el ingreso del número y debe contener una función **sumaDeDigitos** que devuelva la suma de los dígitos. Por ejemplo: si el número ingresado es 258 = 2 + 5 + 8 = 15
5. Escribir una función **max** que tome dos números enteros, y devuelva el máximo de los dos. Probarla en la función principal **main**.
6. Escribir la función **cargarVector** que permita cargar elementos de tipo *float* en un vector a partir del ingreso de datos desde el teclado. Luego, escribir la función **sumaDeVectores** que tome dos vectores A y B como parámetros de entrada, y permita visualizar la suma de cada elemento de A y B. Hacer uso de estas funciones invocándolas desde la función **main** (bloque principal).
7. Se desea almacenar en una matriz **ventas** de 4 filas por 3 columnas, el valor correspondiente al importe vendido en cada una de las sucursales (filas) y en cada mes del primer trimestre (columnas) del año 2019 de una cadena de supermercados. Escribir una función para obtener el máximo importe de venta y a qué sucursal y mes corresponde.
8. Realizar un programa en C que permita grabar un archivo binario de datos de alumnos, en donde cada registro contiene los siguientes campos: *Número de Documento, Nombre y apellido, Código de Carrera* (1- Lic. En Sistemas de Información, 2-Bioquímica, 3-Agrimensura), *código de género* (f-femenino, m-masculino), *edad*.
9. Realizar un programa en C que permita leer el archivo generado en el punto anterior y emita un listado de los alumnos de la carrera de Lic. en Sistemas de Información que tenga el siguiente formato:

DOCUMENTO	NOMBRE Y APELLIDO	EDAD	GÉNERO
11111111	MARIA GOMEZ	18	FEMENINO
22222222	JOSÉ PALACIOS	19	MASCULINO
.....	.....	.....	.....

Además, obtener e informar al final el porcentaje de alumnos de la carrera de Lic. en Sistemas de Información sobre el total de alumnos.

## EJERCICIOS COMPLEMENTARIOS

1. Crear un programa en el que se introduce un número entero y se crea una pirámide de asteriscos. Donde las filas de la pirámide serán igual al número ingresado, y donde la primera fila tendrá un solo asterisco (centrado), en la 2da fila, el doble de la primera más uno, y así sucesivamente.

Por ejemplo, si se introduce el 6, el resultado sería:

```

      *
    ***
  *****
 *****
*****
*****

```

2. Escribir una función **encriptar** que tome una frase que recibe como parámetro, la encripte sumando 3 al código ASCII de cada carácter, y permita visualizar el resultado de la encriptación. Probar esta función en el bloque principal.
3. Escribir un programa que rellene un array con los números pares comprendidos entre 1 y 100, y luego permita visualizar los valores del array en orden ascendente.
4. Realizar un programa en C que permita grabar un archivo de datos (binario) con los siguientes campos: *Número de Cuenta, Documento, Código de movimiento (1-Depósito 2-Extracción), Importe*.
5. Realizar un programa en C que permita leer el archivo generado en el punto anterior y emita un listado de todas las cuentas. Obtener e informar al final los totales de importe de depósito y extracción.

Detalle del informe:

NRO. DE CUENTA	DOCUMENTO	MOVIMIENTO	IMPORTE
11111	11111111	DEPÓSITO	\$2.500
11111	11111111	EXTRACCIÓN	\$1.500
2222	2222222	EXTRACCIÓN	\$4.000
3333	3333333	DEPÓSITO	\$10.000

TOTAL DEPÓSITOS: \$12.500

TOTAL EXTRACCIONES: \$4.500