

CARRERA	ASIGNATURA	Año	Régimen	Plan	Total Horas
Licenciatura en Sistemas	PROGRAMACION I	2	Anual	071/08	120

**EQUIPO DOCENTE:**

PROFESOR	CATEGORÍA
Ing. Fernanda Carmona	Titular
	Asociado
Dr. Emmanuel Frati	Adjunto
	Jefe de Trabajos Prácticos
	Ayudante de 1 <sup>a</sup>
	Ayudante de 2 <sup>da</sup>

**1. CONTENIDOS MÍNIMOS:**

Programación estructurada. Diseño Top-Down. Tipos de datos y operadores. Estructura de un programa. Sentencias de control de flujo. Funciones. Cadenas de caracteres. Números Aleatorios. Arreglos: vectores y matrices. Biblioteca de funciones. Compiladores. Cadenas de caracteres. Funciones de manejo de cadenas. Estructuras. Archivos. Acceso secuencial y directo. Funciones. Alcances en la Memoria. Punteros. Áreas de memoria. Asignación dinámica. Estructuras estáticas y dinámicas. Gestión de Archivos. Ordenamiento de registros. Búsquedas. Estructura de Datos. Pilas y colas. Listas. Almacenamiento en disco. Recursividad. Árboles.

**2. CONTENIDOS DE LA ASIGNATURA:**

**Unidad N° 1: Introducción al Lenguaje C**

Contenidos:

Introducción al lenguaje C. Datos simples: entero, carácter, lógico, real, doble precisión y sus variaciones. Conversiones léxicas : comentarios, identificadores, palabras reservados, constantes de Diversos tipos. Estructura de un programa en C: main(), sentencias simples y compuestas. Definición de variables, su reserva de memoria, inicialización, asignación. Constantes. Operadores: aritméticos, control (relación, igualdad, lógicos). Binarios y unarios. Prioridad. Operaciones con tipos mezclados. Promoción de tipos. Operadores de asignación compuestos. Operadores unarios: prefijo, postfijo.

Bibliografía específica de la unidad:

- PROGRAMACION EN C - Gottfried Byron **Cap. 1, 2, 3**
- C/C++ CURSO DE PROGRAMACION - Francisco Ceballos **Cap. 1, 2**

**Unidad N° 2: Estructuras de Control y de Datos**

Contenidos:

Entrada / Salida simple con funciones estándar. Funciones estándar para tratar caracteres. La estructura **if**, la sentencia **switch**. Los ciclos: **while**, **do-while**, **for**. Operador coma, sentencias **break** y **continue**. Arrays de tipos simples y cadenas de caracteres. Paso de arrays unidimensionales a funciones. Arrays de más de una dimensión, bidimensionales y multidimensionales.

Bibliografía específica de la unidad:

- PROGRAMACION EN C - Gottfried Byron **Cap. 4, 6, 9**
- C/C++ CURSO DE PROGRAMACION - Francisco Ceballos **Cap. 4, 5, 6**

**Unidad N° 3: Funciones en C**

**Contenidos:**

Sintaxis: Formato general. Argumentos formales y actuales. Paso de argumentos a una función. Alcance de los argumentos. Definición y duración de variables dentro de un bloque. Clases de almacenamiento: estático, automático, externo. Recursividad. Estructuras como parámetros y retornos de funciones.

**Bibliografía específica de la unidad:**

- PROGRAMACION EN C - Gottfried Byron **Cap. 7, 8**
- C/C++ CURSO DE PROGRAMACION - Francisco Ceballos **Cap. 3, 8**

**Unidad N°4: Punteros y Estructuras Lineales**

**Contenidos:**

El tipo puntero. Concepto de dirección y valor. Punteros como argumentos de funciones. Aritmética de punteros. Arrays de punteros. Estructuras y Uniones. Tipos de Datos definidos por el usuario. Paso de estructuras a función. Punteros de estructuras. Estructuras lineales en memoria dinámica: pilas, colas y listas. Sus particularidades. Simple y doble vinculación. Archivos sin formato.

**Bibliografía específica de la unidad:**

- PROGRAMACION EN C - Gottfried Byron **Cap. 9, 10**
- C/C++ CURSO DE PROGRAMACION - Francisco Ceballos **Cap. 6, 7**

**Unidad N°5: Archivos**

Conceptos básicos. Archivos secuenciales. Acceso secuencial y directo.

Archivo de texto y estructurados.

Entrada, salida utilizando funciones estándar.

**Bibliografía específica de la unidad:**

- PROGRAMACION EN C - Gottfried Byron **Cap. 12**
- C/C++ CURSO DE PROGRAMACION - Francisco Ceballos **Cap. 9**

**3. Programa de Trabajos Prácticos:**

**Práctico N° 1: Conceptos Básicos**

**Objetivo:**

Presentar un panorama general de la estructura de un programa escrito en C, con el fin de adquirir desde el principio familiaridad con algunos aspectos importantes. Profundizar sobre las reglas y conceptos de los elementos básicos que forman el programa desde el punto de vista léxico.

**Actividades a desarrollar:**

Reconocer identificadores, constantes y constantes simbólicas válidas en C.  
Escribir declaraciones apropiadas para cada grupo de variables y arrays dados.  
Identificar sentencias de expresión, compuestas y de control entre varias sentencias dadas.  
Escribir definiciones apropiadas para cada una de las constantes simbólicas dadas.

**Práctico N° 2: Operadores y Expresiones**

**Objetivo:**

Analizar e identificar los distintos tipos de operadores y funciones disponibles en el lenguaje. Trabajar con expresiones, los operadores y funciones de biblioteca afines.

**Actividades a desarrollar:**

Determinar el valor numérico de cada una de las expresiones aritméticas dadas.  
Identificar el tipo de dato del resultado de las expresiones dadas.  
Identificar los distintos tipos de operadores que intervienen en cada una de las expresiones dadas.  
Identificar el propósito de cada función de biblioteca presentada.  
Identificar entre un grupo de funciones de biblioteca cuáles están disponibles en la versión de C estudiada y que archivos de cabecera requiere.

**Práctico N° 3: Entrada y Salida de Datos**

**Objetivo:**

Identificar las diferentes funciones de entrada y salida disponibles en el lenguaje. Utilizar funciones de entrada y salida en la resolución de problemas.

**Actividades a desarrollar:**

Escribir las sentencias de entrada y salida apropiadas para introducir y visualizar los caracteres individuales dados.

Escribir las sentencias de entrada y salida apropiadas para introducir y visualizar las cadenas de caracteres y valores numéricos dados.

Interpretar el significado de la cadena de control de cada función de entrada y salida presentadas.

Escribir las sentencias necesarias que permitan introducir valores para un grupo de variables dadas.

Resolver problemas a través de la utilización de funciones de entrada y salida de datos, comprobando los resultados en el Laboratorio de Computación.

**Práctico N° 4 : Sentencias de Control**

**Objetivo:**

Analizar e Identificar las diferentes sentencias de control disponibles en el lenguaje. Utilizar sentencias de control en la resolución de problemas.

**Actividades a desarrollar:**

Identificar todas las sentencias presentes en un fragmento de programa dado.

Describir la salida generada por cada uno de los programas presentados.

Resolver problemas con procedimientos repetitivos para la utilización de las sentencias *while*, *do while* y *for*, comprobando los resultados en el Laboratorio de Computación.

Resolver problemas que presenten bifurcaciones para la utilización de sentencias y operadores condicionales, comprobando los resultados en el Laboratorio de Computación.

**Práctico N° 5 : Arrays**

**Objetivo:**

Identificar algunos de los mecanismos que C ofrece para la creación de tipos de datos complejos. Utilizar diferentes tipos de arrays en la resolución de problemas.

**Actividades a desarrollar:**

Describir el array definido en cada una de sentencias dadas e indicar los valores asignados a los elementos individuales del array.

Escribir la definición de arrays apropiada para cada uno de los problemas dados.

Escribir las definiciones y declaraciones para transferir las variables y los arrays a las funciones indicadas en cada una de las situaciones dadas.

Describir la salida generada por cada uno de los programas dados.

Resolver problemas con arrays dados, comprobando los resultados en el Laboratorio de Computación..

**Práctico N° 6 : Funciones y Tipos de Almacenamiento**

**Objetivo:**

Trabajar con los aspectos relacionados a la definición y uso de las funciones. Analizar y aplicar las reglas de su alcance, y los tipos de almacenamiento asociados a las variables que en ellas se definen.

**Actividades a desarrollar:**

Explicar el significado de cada una de las declaraciones de funciones dadas.

Explicar el significado de cada una de las definiciones de funciones dadas.

Escribir la primera línea de la definición de las funciones y declaraciones de argumento formales para cada una de las situaciones dadas.

Describir la salida generada por cada uno de los programas dados, identificando los tipos de almacenamientos presentes.

Añadir las declaraciones de funciones necesarias a cada fragmento de programa presentado.

Resolver problemas dados con llamadas a funciones, comprobando los resultados en el Laboratorio de Computación.

#### Práctico N° 7 : Punteros

##### Objetivo:

Manipular punteros mediante diversas operaciones como asignación, comparación y operaciones aritméticas. Identificar la fuerte relación entre los punteros y las estructuras de datos.

##### Actividades a desarrollar:

Escribir la definición de punteros apropiada para cada una de las situaciones dadas.

Distinguir los valores representados y asignados por las variables presentadas utilizando los operadores dirección e in dirección.

Distinguir el tipo de argumento y el tipo de información que se le pasa y devuelven las distintas definiciones de funciones presentadas.

Escribir las declaraciones necesarias en cada una de las situaciones dadas.

Explicar el propósito de cada una de las declaraciones de arrays en términos de punteros dadas.

Resolver problemas con punteros dados, comprobando los resultados en el Laboratorio de Computación.

#### Práctico N° 8: Estructuras y Uniones

##### Objetivo:

Analizar e identificar las bases para la construcción y manejo de estructuras de datos dinámicas. Trabajar con uniones y estructuras, reconociendo las diferencias, ventajas y desventajas de cada una de ellas.

##### Actividades a desarrollar:

Escribir las definiciones de estructuras y uniones apropiadas para cada una de las situaciones dadas.

Definir nuevos tipos de datos utilizando estructuras y uniones.

Declarar variables punteros a estructura y uniones dadas.

Definir y utilizar estructuras autorreferenciadoras.

Describir la salida generada por cada uno de los programas dados.

Resolver problemas con estructuras y uniones dados, comprobando los resultados en el Laboratorio de Computación.

#### Práctico N° 9: Archivos

##### Objetivo:

Reconocer a los archivos como colecciones de datos almacenadas en dispositivos de memoria secundaria. Utilizar archivos en la resolución de problemas.

##### Actividades a desarrollar:

Generar nuevos archivos secuenciales y abrir archivos existentes según las especificaciones dadas.

Escribir valores introducidos desde el teclado en archivos de acuerdo a las especificaciones dadas.

Actualizar los valores existentes en los archivos especificados.

Resolver problemas con archivos secuenciales dados, comprobando los resultados en el Laboratorio de Computación.

## 12. BIBLIOGRAFÍA:

BIBLIOGRAFÍA BÁSICA			
TÍTULO	AUTOR(ES)	EDITORIAL	LUGAR Y AÑO DE EDICIÓN
PROGRAMACION EN C	Gottfried Byron	MCGRRAW-HILL	2005
ENCICLOPEDIA DEL LENGUAJE C++	Francisco Javier Ceballos Sierra	ALFAOMEGA	2007
ENCICLOPEDIA DEL LENGUAJE C++	Francisco Javier Ceballos Sierra	RA-MA	2003
GUÍA DE TRABAJOS PRÁCTICO	Equipo de la Asignatura	-----	-----
BIBLIOGRAFÍA COMPLEMENTARIA			

TÍTULO	AUTOR(ES)	EDITORIAL	LUGAR Y AÑO DE EDICIÓN
INTRODUCCIÓN A LA PROGRAMACIÓN ESTRUCTURADA EN C.	Francisco A. Martínez Gil, Gregorio Martín Quetglás	UNIVERSITAT DE VALÈNCIA	2003
PROGRAMACION C, MANUAL COMPLETO DE PROGRAMACION	Marcelo Hernán Ruiz	MP EDICIONES CORP	2003
C MANUAL DE REFERENCIA	Herbert Schildt	Mc Graw Hill	2001
INTRODUCCIÓN A LA PROGRAMACIÓN EN C	Marco A. Peña Basurto, José M. Cela Espín	EDICIONS UPC	2000
EL LENGUAJE DE PROGRAMACIÓN c	Brian W. Kernighan - Dennis M. Ritchie	PEARSON EDUCACIÓN	1992

CHILECITO, Provincia de La Rioja, 17 de febrero 2017.

\_\_\_\_\_  
 Profesor Titular