



<b>CARRERA</b>	<b>Tecnicatura Superior en Análisis de Sistemas y Desarrollo de Software</b>			
<b>PLAN DE ESTUDIOS</b>	<b>Resolución 013/23</b>			
<b>DOCENTE</b>	<b>Daniel Reyes</b>			
<b>UNIDAD CURRICULAR</b>	<b>Período</b>	<b>Año</b>	<b>Régimen</b>	<b>Hs. Cátedra</b>
Fundamentos de Programación (FP)	2024	1ro.	Cuatrimestral (1er. cuat.)	5 (cinco)

## 1. Fundamentación

La cátedra de Fundamentos de Programación representa un pilar fundamental en la formación de los estudiantes de la Tecnicatura Superior en Análisis de Sistemas y Desarrollo de Software. Esta asignatura constituye el primer paso crucial en el camino de los estudiantes hacia el dominio de la programación y la ciencia computacional.

El objetivo principal de esta Unidad Curricular es proporcionar a los estudiantes una base sólida en los principios fundamentales de la computación y los algoritmos, así como las habilidades necesarias para comprender y aplicar conceptos informáticos en diversos contextos personales y profesionales. A lo largo del trayecto académico, los estudiantes adquieren conocimientos teóricos y prácticos que les permiten abordar problemas desde el análisis hasta la obtención de soluciones mediante el diseño de algoritmos.

Durante el cursado de la asignatura, se introducen los conceptos básicos de la informática, incluyendo el marco histórico y conceptual necesario para comprender el funcionamiento interno de las computadoras modernas y su impacto en la sociedad. Además, se exploran temas como la clasificación de las computadoras y la representación de la información, sentando así las bases para unidades posteriores. Se abordan también los sistemas de numeración y el cambio de base, aspectos fundamentales para entender la representación de datos en la computación.

Finalmente, se profundiza en el mundo de los algoritmos y la resolución de problemas, proporcionando a los estudiantes una visión detallada de cómo diseñar y ejecutar procedimientos computacionales. Se enfatiza el paradigma de programación estructurada, que será fundamental en asignaturas posteriores, así como la importancia de la prueba de escritorio para verificar la corrección y eficacia de los algoritmos diseñados.

Al finalizar el cursado, los estudiantes incorporarán la capacidad de diseñar algoritmos de manera sistemática y eficiente, preparados para enfrentar desafíos informáticos en su futura práctica profesional.

## 2. Objetivos

- Dominar los conceptos fundamentales de la computación, incluyendo la historia y evolución de las computadoras, así como la clasificación actual de los sistemas informáticos.
- Adquirir habilidades en el diseño, construcción y ejecución de algoritmos para resolver problemas de complejidad media usando diagramas de Nassi-Shneiderman o pseudocódigo.
- Comprender los principios y operaciones de los sistemas de numeración, incluyendo la conversión entre bases y el cambio de base.
- Familiarizarse con los distintos tipos de software, como el software de sistema y el software de aplicaciones, así como su impacto en el funcionamiento de los sistemas informáticos.
- Explorar el paradigma de programación estructurada y sus diferentes elementos, incluyendo el flujo de control, las variables y los operadores.
- Aplicar el álgebra de Boole en el diseño de algoritmos, utilizando operadores booleanos y tablas de verdad para resolver problemas.
- Dominar las estructuras de control de flujo de los algoritmos, incluyendo estructuras secuenciales, selectivas y repetitivas.
- Desarrollar habilidades para la prueba de escritorio, utilizando tablas de prueba para verificar la corrección y eficacia de los algoritmos diseñados.
- Adquirir conocimientos y habilidades que sienten las bases para asignaturas posteriores en el área de la informática y la programación.
- Aplicar los conceptos aprendidos en la asignatura para diseñar algoritmos que aborden problemas del mundo real de manera sistemática y eficiente.

## 3. Contenidos

### **Unidad 1: Historia de la Computación y de la Informática**

Los precursores de la Computación. ¿Qué es una computadora? Generaciones de computadoras. Clasificación actual de las computadoras. Organización física de una computadora (hardware): dispositivos de Entrada/Salida (E/S), Memoria central de la computadora, Memoria caché y Dispositivos de almacenamiento secundario.

Software: concepto. Software de sistema y software de aplicaciones. Software de código abierto y software propietario. El lenguaje de la computadora. Representación de la información. La evolución de la Programación.

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Barceló, M.	Una historia de la informática.	Editorial UOC	España, 2008
Martínez Durá, R., Boluda Grau, J. y Pérez Solano, J.	Estructura de Computadoras y Periféricos.	Alfaomega	España, 2001
Joyanes Aguilar, L.	Fundamentos de programación. Algoritmos, estructuras de datos y objetos, 5ta. ed.	Mc Graw Hill	México, 2020
Stallings, W.	Organización y arquitectura de computadoras. Diseño para optimizar prestaciones, 7ma. ed.	Prentice Hall	México, 2006
Riemann, B.	Historia Ilustrada de la Informática.	CreateSpace Publishing	EEUU, 2015
Tanenbaum, A.	Organización de computadoras : Un enfoque estructurado.	Pearson Educación	México, 2012
Wilson, K.	Computer Fundamentals: The Step-by-step Guide to Understanding Computers	eLuminet Press	EEUU, 2021
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

### **Unidad 2: Sistemas de numeración**

Concepto de sistema de numeración. Historia y evolución de los sistemas de numeración. Tipos de sistemas de numeración. Sistemas de numeración no posicionales y posicionales. Base y dígitos de un sistema de numeración. Sistemas con base. Convenciones para indicar la base en la representación de números. Principio fundamental del sistema posicional. Operaciones básicas con sistemas de numeración posicionales con distintas bases: suma, resta, multiplicación y división.

Cambio de base. Métodos de cambio de Base: sumas ponderadas, divisiones reiteradas y productos reiterados. Conversión de números decimales con parte fraccionaria a otras bases. Método de cambio de base por agrupación de bits. Métodos de conversión directo e indirecto.

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Comellas, F.; Fàbrega, J.; Llado, A.; Lladí, A.; Serra, O.	Matemática discreta	Alfaomega	México, 2002
Floyd, T.	Fundamentos de Sistemas Digitales, 9na. Ed.	Pearson Educación	España, 2006
García Muñoz, M.	Matemática discreta para la computación, 2da. ed.	Universidad de Jaén	España, 2015
Lipschutz, S	Matemáticas para computación.	Mc Graw Hill	México, 1983
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

### **Unidad 3: Algoritmos y Resolución de Problemas**

Concepto de algoritmo. Características de los algoritmos. Tipos de Algoritmos. Algoritmos Computacionales y No Computacionales. Paradigma. Paradigma de programación. Tipos de paradigmas de programación. El Teorema del Programa Estructurado. El paradigma de Programación Estructurada.

Resolución de Problemas con algoritmos. Definición del Problema. Análisis del Problema. Diseño del Algoritmo. Prueba del Algoritmo. Álgebra de Boole. Uso del Álgebra de Boole en el diseño de Algoritmos. Operadores del Álgebra de Boole. Tablas de Verdad. Leyes de De Morgan. Formas comunes de representar algoritmos. Diagramas de Nassi-Shneiderman. El flujo de control de un algoritmo. Variables. Expresiones y operadores. Operadores de asignación. Operadores aritméticos. Operadores de relación. Operadores Booleanos (Lógicos).

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Cairó Battistutti, O.	Metodología de la programación. Algoritmos, diagramas de flujo y programas, 3ra. ed.	Alfaomega	México, 2006
Cormen, T.	Algorithms Unlocked.	The MIT Press	EEUU, 2013
Joyanes Aguilar, L.	Fundamentos de programación. Algoritmos, estructuras de datos y objetos, 5ta. ed.	Mc Graw Hill	México, 2020
Louridas, P.	Algorithms	The MIT Press	EEUU, 2020
Sedgewick, R. y Wayne, K.	Algorithms	Addison-Wesley	EEUU, 2011
Wirth, N.	Algoritmos + Estructuras de Datos = Programas	Ediciones del Castillo	España, 1993
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

#### **Unidad 4: Estructuras de Control de Flujo de Algoritmos y Prueba de Escritorio**

Estructuras de Control de Flujo. Estructura secuencial. Estructuras selectivas o condicionales. Estructura selectiva simple o doble Si- Entonces-SiNo. Estructuras selectivas anidadas (en escalera). Estructura selectiva múltiple Según-hacer. Estructuras repetitivas. Estructura repetitiva condicional Mientras-hacer. Estructura repetitiva condicional Hacer-mientras. Estructura repetitiva incondicional Para. Contadores, sumadores y productores. Equivalencia entre las estructuras de control Mientras-hacer y Para.

Prueba de Escritorio. Objetivos de la prueba de escritorio. Pasos para realizar una Prueba de Escritorio. Tabla de Prueba de Escritorio. Cómo realizar una prueba de escritorio. Ejemplos. Beneficios y limitaciones de la Prueba de Escritorio.

<b>Bibliografía específica de la Unidad</b>			
<b>Autor</b>	<b>Título</b>	<b>Editorial</b>	<b>Lugar y año de edición</b>
Cairó Battistutti, O.	Metodología de la programación. Algoritmos, diagramas de flujo y programas, 3ra. ed.	Alfaomega	México, 2006
Cormen, T.	Algorithms Unlocked.	The MIT Press	EEUU, 2013
Joyanes Aguilar, L.	Fundamentos de programación. Algoritmos, estructuras de datos y objetos, 5ta. ed.	Mc Graw Hill	México, 2020
Louridas, P.	Algorithms	The MIT Press	EEUU, 2020
Sedgewick, R. y Wayne, K.	Algorithms	Addison-Wesley	EEUU, 2011
Wirth, N.	Algoritmos + Estructuras de Datos = Programas	Ediciones del Castillo	México, 1992
Reyes, D.	Apuntes de Cátedra.	Del autor	Argentina, 2024

#### **4. Estrategias de Enseñanza**

- Trabajos grupales e individuales.
- Elaboración de informes escritos.
- Exposiciones grupales e individuales
- Aprendizaje cooperativo.
- Aula invertida.

#### **5. Evaluación**

##### **5.1. Criterios de Evaluación durante el Cursado**

- Participación comprometida en los trabajos propuestos, ya sean individuales y/o grupales.
- Compromiso y responsabilidad en las tareas a desarrollar.
- Presentación de trabajos en tiempo y forma.
- Manejo adecuado de la bibliografía.
- Coherencia, claridad conceptual, integración y pertinencia tanto en oratoria como en el discurso escrito.
- Dominio, comprensión y transposición de los contenidos del programa.
- Capacidad de resolución de problemáticas presentadas.
- Capacidad de organizar, coordinar y elaborar la tarea encomendada.
- Aplicación de conocimientos y dominio de técnicas, mediante la respuesta a preguntas y la resolución de problema.
- Participación activa y pertinente en el desarrollo de las clases.

##### **5.2. Instrumentos de Evaluación**

Para valorar los aprendizajes de los alumnos y sistematizar dichas valoraciones, se utilizarán los siguientes instrumentos:

- Evaluaciones escritas
- Evaluaciones orales
- Trabajos prácticos
- Coloquios
- Informes

##### **5.3. Criterios de Acreditación**

###### **5.3.1. Para Regularizar la unidad curricular**

Para regularizar la asignatura, el alumno deberá:

- Tener el 80% de asistencia a las clases o de participación en el aula virtual.
- Presentar a término y aprobar el 90% de los trabajos prácticos o actividades propuesto.
- Presentar y aprobar un portfolio de evidencias de las experiencias del alumno.
- Aprobar los exámenes parciales propuestos o sus respectivos exámenes recuperatorios. La nota mínima obtenida no deber ser menor de 5 (cinco), en una escala de 1 (uno) a diez (10).

- Obtener una calificación final mínima de 5 (cinco) puntos, en una escala de 1(uno) a 10(diez), al terminar el cursado de la asignatura.

Las condiciones de regularización descriptas serán expuestas a los alumnos durante la presentación de la Unidad Curricular y del plan de trabajo de la misma.

### 5.3.2. Para Promocionar la Unidad Curricular

Según lo establece la normativa vigente (RAM), la Unidad Curricular NO ES promocionable.

### 5.4. Para Finalizar la Unidad Curricular como alumno Regular

En base a la normativa vigente (RAM), para rendir esta Unidad Curricular en calidad de alumno regular el alumno deberá rendir un primer examen final escrito sobre aspectos teóricos de la misma. Este examen versará sobre las temáticas comprendidas en el programa de la asignatura, desde la perspectiva de la bibliografía obligatoria. Los criterios de evaluación se basan en la acreditación de la lectura de la bibliografía, la argumentación sólida que utilice la terminología y los conceptos del campo de estudio y la apropiación crítica de los mismos.

### 5.5. Para finalizar la Unidad Curricular como alumno Libre

De acuerdo con la normativa actual (RAM), las condiciones establecidas para rendir esta Unidad Curricular en calidad de alumno libre comprenden:

- Presentación y aprobación de la totalidad de los trabajos prácticos o de un trabajo práctico integrador equivalente, en ambos casos, propuestos por la Cátedra. Para ello, el alumno deberá comunicarse con la Cátedra con treinta (30) días de anticipación a la fecha del examen final, a fin de obtener el conjunto de trabajos prácticos que deberá resolver. Los mismos deberán ser entregados al docente responsable siete (7) días antes de la fecha de examen final.
- Superada la instancia anterior, el alumno estará en condiciones de rendir un primer examen final escrito sobre aspectos teóricos (25%) y prácticos (75%) de la asignatura.
- En caso de aprobar el examen escrito, el alumno estará en condiciones de rendir un segundo examen final de características similares al que rinden los alumnos regulares. Este examen será tomado en forma oral o escrita y versará sobre las temáticas comprendidas en el programa de la asignatura, desde la perspectiva de la bibliografía obligatoria. Los criterios de evaluación se basan en la acreditación de la lectura de la bibliografía, la argumentación sólida que utilice la terminología y los conceptos del campo de estudio y la apropiación crítica de los mismos.

## 6. Materiales para el aprendizaje

- Apuntes de Cátedra.
- Guía de trabajos prácticos
- Pizarra y fibrones.
- Notebook.
- Cañón proyector.

## 7. Cronograma tentativo de los Contenidos Programados

El siguiente cronograma representa el avance semanal de los contenidos desarrollados, considerando el calendario escolar en curso, incluyendo los días feriados y/o no laborables.

Semana	Fecha	Tema
1	11/03/2024 - 17/03/2024	Precusores de la Computación. ¿Qué es una computadora? Generaciones de computadoras. Clasificación actual de las computadoras. Organización física de una computadora (hardware).
2	18/03/2024 - 24/03/2024	Software: concepto. Software de sistema y software de aplicaciones. Software de código abierto y software propietario. El lenguaje de la computadora. Representación de la información. La Evolución de la Programación.
3	01/04/2024 - 07/04/2024	Concepto de sistema de numeración. Historia y evolución de los sistemas de numeración. Tipos de sistemas de numeración. Sistemas de numeración no posicionales y posicionales. Base y dígitos de un sistema de numeración.
4	08/04/2024 - 14/04/2024	Sistemas con base. Convenciones para indicar la base en la representación de números. Principio fundamental del sistema posicional. Operaciones básicas con sistemas de numeración posicionales con distintas bases: suma, resta, multiplicación y división. Ejemplos
5	15/04/2024 - 21/04/2024	Cambio de base. Métodos de cambio de Base: sumas ponderadas, divisiones reiteradas y productos reiterados. Ejemplos.



# INSTITUTO DE EDUCACIÓN SUPERIOR N° 6.012

Pellegrini N° 295 - El Carril (Salta) – Rep. Argentina  
FAX - T.E. N° 4-908015 - Email: isfd6012@gmail.com



Semana	Fecha	Tema
6	22/04/2024 - 28/04/2024	Conversión de números decimales con parte fraccionaria a otras bases. Método de cambio de base por agrupación de bits. Métodos de conversión directo e indirecto.
7	29/04/2024 - 05/05/2024	Concepto de algoritmo. Características de los algoritmos. Tipos de Algoritmos. Algoritmos Computacionales y No Computacionales. Paradigma. Paradigma de programación. Tipos de paradigmas de programación.
8	06/05/2024 - 12/05/2024	El Teorema del Programa Estructurado. El paradigma de Programación Estructurada. Resolución de Problemas con algoritmos. Definición del Problema. Análisis del Problema. Diseño del Algoritmo. Prueba del Algoritmo. Álgebra de Boole. Uso del Álgebra de Boole en el diseño de Algoritmos.
9	13/05/2024 - 19/05/2024	Operadores del Álgebra de Boole. Tablas de Verdad. Leyes de De Morgan. Estructuras de Control de Flujo. Estructura secuencial. Estructuras selectivas o condicionales. Estructura selectiva simple o doble Si- Entonces-SiNo. Ejemplos
10	20/05/2024 - 26/05/2024	Estructuras selectivas anidadas. Ejemplos. Estructura selectiva múltiple Según-hacer. Ejemplos. Estructuras repetitivas. Estructura repetitiva condicional Mientras-hacer. Ejemplos.
11	27/05/2024 - 02/06/2024	Estructura repetitiva condicional Hacer-mientras. Ejemplos. Estructura repetitiva incondicional Para. Ejemplos. Contadores, sumadores y productores. Equivalencia entre las estructuras de control Mientras-hacer y Para. Ejemplos.
12	03/06/2024 - 09/06/2024	Prueba de Escritorio. Objetivos de la prueba de escritorio. Pasos para realizar una Prueba de Escritorio. Tabla de Prueba de Escritorio. Cómo realizar una prueba de escritorio. Ejemplos. Beneficios y limitaciones de la Prueba de Escritorio.
13	10/06/2024 - 16/06/2024	<b>Examen parcial</b>
14	17/06/2024 - 23/06/2024	<b>Recuperatorio del examen parcial</b>

Firma del docente