

**Universidad Tecnológica Nacional  
Facultad Regional  
San Francisco**



**Tecnicatura Superior en Programación**

# **Programación I**

**PLANIFICACIÓN CICLO LECTIVO  
2013**

## **ÍNDICE**

<b>ÍNDICE .....</b>	<b>2</b>
<b>PROFESIONAL DOCENTE A CARGO .....</b>	<b>3</b>
<b>UBICACIÓN.....</b>	<b>4</b>
<b>OBJETIVOS .....</b>	<b>5</b>
<b>ORGANIZACIÓN DE CONTENIDOS .....</b>	<b>6</b>
<b>PROGRAMA ANALÍTICO.....</b>	<b>8</b>
<b>CRITERIOS DE EVALUACIÓN.....</b>	<b>9</b>
EVALUACIÓN:.....	9
AUTOEVALUACIÓN:.....	9
<b>PLAN DE TRABAJO .....</b>	<b>10</b>
<b>METODOLOGÍA .....</b>	<b>12</b>
<b>BIBLIOGRAFÍA .....</b>	<b>¡ERROR! MARCADOR NO DEFINIDO.</b>
<b>ARTICULACIÓN .....</b>	<b>14</b>
ARTICULACIÓN CON EL ÁREA: .....	14
TEMAS RELACIONADOS CON MATERIAS DEL ÁREA: .....	15
ARTICULACIÓN CON EL NIVEL: .....	16
TEMAS RELACIONADOS CON MATERIAS DEL NIVEL: .....	17
ARTICULACIÓN CON LAS CORRELATIVAS: .....	18
TEMAS RELACIONADOS CON LAS CORRELATIVAS: .....	19
<b>ORIENTACIÓN.....</b>	<b>20</b>
DEL ÁREA: .....	20
DE LA ASIGNATURA: .....	20

## **PROFESIONAL DOCENTE A CARGO**

Docente	Categoría	Título Profesional
<b>Sergio Andrés Revilla</b>	Adjunto interino	Ingeniero en Sistemas de Información

## **UBICACIÓN**

Dentro del contexto curricular prescripto se ubica en:

**Carrera:** Tecnicatura Superior en Programación

**Plan:** 2003

**Orientación:** Prioritariamente práctica con fundamentos teóricos

**Área:** Disciplinas Tecnológicas

**Nivel:** 1

**Carga Horaria Semanal:** 6

**Régimen:** Cuatrimestral

### **Distribución horaria**

Formación							Total de horas
Teórica			Práctica				
Teoría	Práctica	Laboratorio	Formación experimental	Resolución de problemas de Ingeniería	Proyecto y diseño	Práctica profesional supervisada	
24	60	-	-	-	-	-	84

## **OBJETIVOS**

- Brindar una introducción a las herramientas metodológicas necesarias para el desarrollo de software.
- Lograr que el estudiante adquiera aptitud en la resolución de problemas a través del desarrollo de algoritmos.
- Que el estudiante logre autonomía y pueda explorar en forma independiente las posibilidades que ofrecen los distintos lenguajes.
- Que el estudiante pueda evaluar el resultado de sus soluciones utilizando su capacidad reflexiva y crítica.

## **ORGANIZACIÓN DE CONTENIDOS**

### **Eje Temático Nº 1: Introducción al Software**

- Contenidos Conceptuales: Software. Programa. Código Fuente. Sistema Operativo. Lenguajes de alto nivel y lenguajes de bajo nivel. Lenguaje de máquina. Ambiente de Programación. Compilador. Intérprete. Diferencia entre Compilador e Intérprete.
- Contenidos Procedimentales: Comprensión y uso correcto de los distintos conceptos.
- Contenidos Actitudinales: Demostración de atención y valoración sobre los distintos conceptos.

### **Eje Temático Nº 2: Algoritmos**

- Contenidos Conceptuales: Algoritmos. Robustez de un algoritmo. Correctitud de un algoritmo. Completitud de un algoritmo. Eficiencia y Eficacia de un algoritmo. Resolución de problemas. Diagramas de flujo. Pseudocódigo.
- Contenidos Procedimentales: Diseño de algoritmos y comprobación su validez. Comprensión de las características más importantes de los algoritmos. Utilización de diagramas de flujo como forma gráfica de representar una situación real.
- Contenidos Actitudinales: Interés y comprensión en las representaciones gráficas como forma de resolver problemas reales.

### **Eje Temático Nº 3: Programas**

- Contenidos Conceptuales: Constantes. Variables. Definición de Variables. Reservando memoria. Tipos de datos. Inicializar una variable. Asignación de un valor. Palabras reservadas o claves. Expresiones y Sentencias. Operadores. Aplicaciones en Pseudocódigo..
- Contenidos Procedimentales: Conocimiento de los distintos tipos de datos. Resolución de diferentes problemas de la vida real utilizando pseudocódigo.
- Contenidos Actitudinales: Interés y valoración de la realización de programas en Pseudocódigo.

#### **Eje Temático Nº 4: Estructuras de Programación**

- Contenidos Conceptuales: Estructuras de control. Partes de una estructura de control. Sentencias o instrucciones. Estructuras de selección if. Estructuras de selección if / else. Estructuras condicionales Anidadas. Estructura While. Estructura For. Estructura Do While. Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.
- Contenidos Procedimentales: Elaboración de algoritmos utilizando las estructuras elegidas.
- Contenidos Actitudinales: Valoración de la importancia de la correcta selección de una estructura de programación.

#### **Eje Temático Nº 5: Vectores, Matrices**

- Contenidos Conceptuales: Concepto. Dimensión. Notación de un vector. Inicialización de vectores y matrices. Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.
- Contenidos Procedimentales: Resolución de ejercicios de aplicación con vectores y matrices. Aplicación de los conceptos a problemas reales.
- Contenidos Actitudinales: Interés y comprensión sobre el correcto uso de vectores y matrices.

## **PROGRAMA ANALÍTICO**

### **Eje Temático Nº 1: Introducción al Software**

#### **Unidad Nº 1: Introducción al Software**

Software.  
Programa.  
Código Fuente.  
Sistema Operativo.  
Lenguajes de alto nivel y lenguajes de bajo nivel. Lenguaje de máquina.  
Ambiente de Programación.  
Compilador. Intérprete.  
Diferencia entre Compilador e Intérprete.

### **Eje Temático Nº 2: Algoritmos**

#### **Unidad Nº 2: Algoritmos**

Algoritmos.  
Robustez de un algoritmo.  
Correctitud de un algoritmo.  
Complejidad de un algoritmo.  
Eficiencia y Eficacia de un algoritmo.  
Resolución de problemas.  
Diagramas de flujo.  
Pseudocódigo.

### **Eje Temático Nº 3: Programas**

#### **Unidad Nº 3: Programas**

Constantes.  
Variables.  
Definición de Variables.  
Reservando memoria.  
Tipos de datos.  
Inicializar una variable.  
Asignación de un valor.  
Palabras reservadas o claves.  
Expresiones y Sentencias.  
Operadores.  
Aplicaciones en Pseudocódigo.

### **Eje Temático Nº 4: Estructuras de Programación**

#### **Unidad Nº 4: Estructuras de control**

Estructuras de control.  
Partes de una estructura de control.  
Sentencias o instrucciones.



### **Unidad Nº 5: Estructuras de Programación condicionales**

Estructuras de selección if.  
Estructuras de selección if / else.  
Estructuras condicionales Anidadas.  
Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.

### **Unidad Nº 6: Estructuras de programación iterativas**

Estructura While.  
Estructura For.  
Estructura Do While.  
Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.

### **Eje Temático Nº 5: Vectores, Matrices**

#### **Unidad Nº 7: Vectores y Matrices**

Concepto. Dimensión.  
Notación de un vector.  
Inicialización de vectores y matrices.  
Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.

## **CRITERIOS DE EVALUACIÓN**

### **Evaluación:**

#### **Regularidad:**

- 70% de asistencia.
- Entrega de carpeta de trabajos prácticos completa.
- Aprobación de dos parciales escritos, de contenidos prácticos, con una nota superior a 4 en cada uno. Cada parcial tiene su recuperatorio.

#### **Examen final:**

- Será escrito y se aprobará con un porcentaje mínimo de 60%. Se evaluarán contenidos teóricos y prácticos.

### **Autoevaluación:**

Será realizada utilizando el instrumento elaborado desde Secretaría Académica y aprobado por Consejo Académico.

## PLAN DE TRABAJO

<b>Eje temático Nº 1: Introducción al Software</b>					
Semana	Contenidos	Metodología	Evaluación	Nivel de Profundidad	Bibliografía
1	<b>Unidad Nº 1: Introducción al Software</b> Software. Programa. Código Fuente. Sistema Operativo. Lenguajes de alto nivel y lenguajes de bajo nivel. Lenguaje de máquina. Ambiente de Programación. Compilador. Intérprete. Diferencia entre Compilador e Intérprete.	Exposición teórica y resolución de ejercicios prácticos	Informal en clases. Parte práctica: 1º Parcial	Informativo y conceptual en un principio y luego orientado a la resolución de situaciones problemáticas reales.	Deitel y Deitel <b>C++ Cómo Programar – Sexta Edición</b>

<b>Eje temático Nº 2: Algoritmos</b>					
Semana	Contenidos	Metodología	Evaluación	Nivel de Profundidad	Bibliografía
2	<b>Unidad Nº 2: Algoritmos</b> Algoritmos. Robustez de un algoritmo. Correctitud de un algoritmo. Complejidad de un algoritmo. Eficiencia y Eficacia de un algoritmo. Resolución de problemas. Diagramas de flujo. Pseudocódigo.	Exposición teórica y resolución de ejercicios prácticos	Informal en clases. Parte práctica: 1º Parcial	Informativo y conceptual en un principio y luego orientado a la resolución de situaciones problemáticas reales.	Deitel y Deitel <b>C++ Cómo Programar – Sexta Edición</b>

<b>Eje temático Nº 3: Programas</b>					
Semana	Contenidos	Metodología	Evaluación	Nivel de Profundidad	Bibliografía
3-4	<b>Unidad Nº 3: Programas</b> Constantes. Variables. Definición de Variables. Reservando memoria. Tipos de datos. Inicializar una variable. Asignación de un valor. Palabras reservadas o claves. Expresiones y Sentencias. Operadores. Aplicaciones en Pseudocódigo.	Exposición teórica y resolución de ejercicios prácticos	Informal en clases. Parte práctica: 1º Parcial	Informativo y conceptual en un principio y luego orientado a la resolución de situaciones problemáticas reales.	Deitel y Deitel <b>C++ Cómo Programar – Sexta Edición</b>

<b>Eje temático Nº 4: Estructuras de Programación</b>					
Semana	Contenidos	Metodología	Evaluación	Nivel de Profundidad	Bibliografía
5-6-7-8-9-10	<p><b>Unidad Nº 5: Estructuras de control</b> Estructuras de control. Partes de una estructura de control. Sentencias o instrucciones.</p> <p><b>Unidad Nº 6: Estructuras de programación condicionales</b> Estructuras de selección if. Estructuras de selección if / else. Estructuras condicionales Anidadas. Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.</p> <p><b>Unidad Nº 7: Estructuras de programación iterativa o repetitiva.</b> Estructura While. Estructura For. Estructura Do While. Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.</p>	Exposición teórica y resolución de ejercicios prácticos	Informal en clases. Parte práctica: 1º Parcial	Informativo y conceptual en un principio y luego orientado a la resolución de situaciones problemáticas reales.	Deitel y Deitel <b>C++ Cómo Programar – Sexta Edición</b>

<b>Eje temático Nº 5: Vectores, Matrices y cadenas de caracteres</b>					
Semana	Contenidos	Metodología	Evaluación	Nivel de Profundidad	Bibliografía
12-13-13-14	<p><b>Unidad Nº 8: Vectores, Matrices y cadenas de caracteres</b> Concepto. Dimensión. Notación de un vector. Inicialización de vectores y matrices. Ejercicios de Aplicación en Diagramas de Flujo y Pseudocódigo.</p>	Exposición teórica y resolución de ejercicios prácticos	Informal en clases. Parte práctica: 2º Parcial	Informativo y conceptual en un principio y luego orientado a la resolución de situaciones problemáticas reales.	Deitel y Deitel <b>C++ Cómo Programar – Sexta Edición</b>

## **METODOLOGÍA**

Clases orientadas a la resolución de ejercicios prácticos mediante el desarrollo de algoritmos y pseudocódigo.

Se facilitará a los alumnos un Apunte de la cátedra con el contenido teórico-práctico completo de la asignatura separado en 3 módulos:

Módulo 1 – Unidades 1,2,3

Módulo 2 – Unidades 4,5,6

Módulo 3 – Unidad 7

Se entregarán trabajos prácticos de cada una de las unidades que se irán resolviendo en el aula para la aplicación de los contenidos enseñados.

## **BIBLIOGRAFÍA**

- JOYANES AGUILAR, Luis ; ZAHONERO MARTÍNEZ, Ignacio.  
*Programación en C: metodología, algoritmos y estructura de datos.*  
2a. ed.  
McGraw-Hill Interamericana, [2007].  
ISBN: 9788448198442.  
(AI 2013: 3 ejemplar/es en Colección UTN)
- DEITEL, Harvey M. ; DEITEL, Paul J.  
*C++ cómo programar.*  
6a. ed.  
Pearson Educación, 2009.  
ISBN: 9789702612735.  
(AI 2013: 2 ejemplar/es en Colección UTN)
- JOYANES AGUILAR, Luis.  
*Fundamentos de programación: algoritmos y estructura de datos.*  
1a. ed.  
McGraw-Hill Interamericana, 1988.  
ISBN: 8476152019.  
(AI 2013: 1 ejemplar/es en Colección UTN)
- ARRIOJA LANDA COSIO, Nicolás.  
*C#: guía total del programador: [nivel principiante-intermedio].*  
1a. ed.  
Fox Andina ; Gradi, 2010.  
ISBN: 9789872601355.  
(AI 2013: 2 ejemplar/es en Colección UTN)

## **ARTICULACIÓN**

### **Articulación con el Área:**

Asignatura	Carga Horaria	Porcentaje
<b>Programación I</b>	<b>6 hs cátedra</b>	<b>20%</b>
Sistemas de Procesamiento de Base de Datos	6 hs cátedra	20%
Laboratorio de Computación I	6 hs cátedra	50%
Programación II	6 hs cátedra	30%
Arquitectura y Sistemas Operativos	6 hs cátedra	20%
Laboratorio de Computación II	6 hs cátedra	20%
Programación III	6 hs cátedra	20%
Elementos de Investigación Operativa	6 hs cátedra	20%
Laboratorio de Computación III	6 hs cátedra	20%
Metodología de Sistemas I	12 hs cátedra	40%
Diseño y Administración de Base de Datos	6 hs cátedra	20%
Laboratorio de Computación IV	6 hs cátedra	20%

**Temas relacionados con materias del área:**

<b>Programación II</b>	<b>Tema relacionado</b>
Arreglos, cadenas	Arreglos, cadenas
Funciones	Funciones

<b>Arquitectura y Sistemas Operativos</b>	<b>Tema relacionado</b>
Software	Introducción al Software
Sistema Operativo	Sistema Operativo

**Articulación con el Nivel:**

Asignatura	Carga Horaria	Porcentaje
<b>Programación I</b>	<b>6 hs cátedra</b>	<b>20%</b>
Sistemas de Procesamiento de Base de Datos	6 hs cátedra	30%
Matemática	9 hs cátedra	20%
Inglés I	3 hs cátedra	20%
Laboratorio de computación I	6 hs cátedra	50%



**Temas relacionados con materias del nivel:**

<b>Matemática</b>	<b>Tema relacionado</b>
Algoritmos en Pseudocódigo	Conjuntos / Matrices
Variables Estructuradas Arrays y Funciones	Matrices / Grafos y árboles

**Articulación con las correlativas:**

Asignatura	Para cursar		Para rendir
	Cursada	Aprobada	Aprobada
<b>Programación I</b>	No tiene correlativas hacia abajo por ser una materia del 1º cuatrimestre de la carrera.		

**Temas relacionados con las correlativas:**

<b>Programación I</b>	No tiene correlativas hacia abajo por ser una materia del 1º cuatrimestre de la carrera
-----------------------	---

## **ORIENTACIÓN**

### **Del Área:**

La carrera Tecnicatura Superior en Programación y sus asignaturas en general tienen una orientación práctica-operativa. El diseño curricular demuestra una clara tendencia a la aplicación directa de los conocimientos de cada cátedra en la programación de sistemas.

### **De la Asignatura:**

Dentro del marco anterior, la orientación de la cátedra Programación I es un tanto teórica y gran parte práctica. Se pretende que el alumno aplique los conceptos adquiridos a casos y situaciones reales. Es por tal motivo que, para cada tema desarrollado, se realizan prácticas que incluyen problemas de aplicación en el área de programación u organización de empresas en general.