



Universidad Guadalajara  
Centro Universitario del Sur

**Programa de Estudio**

**1. IDENTIFICACIÓN DE LA UNIDAD DE APRENDIZAJE**

División

**DIVISIÓN DE CIENCIAS EXACTAS, NATURALES Y TECNOLÓGICAS**

Departamento

**DEPARTAMENTO DE CIENCIAS COMPUTACIONALES E INNOVACIÓN TECNOLÓGICA**

Academia

**ACADEMIA DE PROGRAMACIÓN**

Programa(s) educativo(s)

**PROGRAMA DE ESTUDIO POR COMPETENCIAS**

Denominación de la unidad de aprendizaje:

**PROGRAMACIÓN**

Clave de la materia:	Horas de teoría:	Horas de práctica:	Carga horaria global:	Valor en créditos:
19887	48	32	80	8

Tipo de curso:		Nivel en que se ubica:	Prerrequisitos:
C = curso		Técnico Medio	Ninguno
CL = curso laboratorio		Técnico Superior	
L = laboratorio		Universitario	
P = práctica		<b>Licenciatura</b>	
T = taller		Especialidad	
<b>CT = curso - taller</b>	<b>x</b>	Maestría	
N = clínica		Doctorado	
M = módulo			
S = seminario			

Área de formación:

Básico Común

Perfil docente:

Se requiere un profesional con estudios mínimos de licenciatura en las ciencias computacionales, de carreras que tengan carga en el área de programación.

Elaborado por:

Actualizado por:

Dr. Abraham Jair López Villalvazo	Dr. Abraham Jair López Villalvazo
-----------------------------------	-----------------------------------

Fecha de elaboración:

Fecha de última actualización:

Fecha de última evaluación:

Fecha de aprobación por Colegio Departamental:

Julio de 2018	Julio de 2020		
---------------	---------------	--	--

## 2. PRESENTACIÓN DE LA UNIDAD DE APRENDIZAJE

Esta Unidad de Aprendizaje tiene como objetivo iniciar en la comprensión de los algoritmos, su definición, metodologías de desarrollo y en su posterior codificación en algún lenguaje de programación.

La materia de Programación contiene teoría y práctica de la programación estructurada brindando las bases necesarias para resolver y diseñar soluciones a problemas computables mediante la representación de diagrama o pseudocódigo. Al concluir esta asignatura servirá de suministro para que el alumno pueda continuar con futuras materias de programación del plan curricular y obtener conocimientos que se aplican en las empresas desarrolladoras de software, aunque en la actualidad lo que se utiliza en éstas es la programación orientada a objetos; la programación estructurada proporciona técnicas de abstracción y conceptualización requeridas para tal fin.

## 3. OBJETIVO GENERAL/COMPETENCIA

Diseña soluciones de problemas computables de baja y mediana complejidad mediante lenguajes algorítmicos para su posterior codificación en un lenguaje de programación utilizando diseño estructurado.

## 4. CAMPO DE APLICACIÓN PROFESIONAL DE LOS CONOCIMIENTOS

--

## 5. SABERES:

<b>Prácticos</b>	<ul style="list-style-type: none"> <li>• Elementos generales de un sistema de proceso.</li> <li>• Características y concepto de algoritmo</li> <li>• Representación de un algoritmo</li> <li>• Elementos y estructuras de control básicas</li> <li>• Estructuras de control desde la perspectiva de programación estructurada.</li> <li>• Analizar un problema dado desde la perspectiva de programación estructurada.</li> <li>• Diseño de soluciones a problemas desde la perspectiva de programación estructurada.</li> <li>• Terminología de la programación estructurada.</li> </ul>
<b>Teóricos</b>	<ul style="list-style-type: none"> <li>• Analizar un problema dado desde la perspectiva de la programación estructurada.</li> <li>• Abstractar y conceptualizar elementos de un problema a resolver para diseñar soluciones a problemas computables desde la perspectiva de programación estructurada.</li> <li>• Identificar y solucionar errores lógicos en la elaboración de diseño de soluciones.</li> <li>• Identificación y solución de errores lógicos en el replanteamiento del diseño de soluciones.</li> <li>• Conceptualizar formas de programación estructurada.</li> <li>• Abstracción para la solución de problemas para su futura codificación en un lenguaje de programación.</li> </ul>

<b>Formativos</b>	<ul style="list-style-type: none"> <li>• Ubicar y valorar el análisis en el logro de un objetivo.</li> <li>• Resaltar y valorar la importancia del orden lógico de sus acciones en el logro de un objetivo.</li> <li>• Resaltar y fomentar la importancia de la planeación para el logro de un objetivo, mediante el diseño de soluciones de problemas computables.</li> <li>• Realizar análisis de información sobre los diversos aspectos que integran la programación estructurada.</li> <li>• actitudes de colaboración, compañerismo y responsabilidad de trabajo en grupo.</li> <li>• Importancia de considerar antecedentes en el replanteamiento de un objetivo al realizar replanteamiento de un diseño.</li> <li>• Se formará con valores de responsabilidad, profesionalismo, búsqueda de calidad y excelencia en los servicios y el valor de superación continua para mantenerse actualizado.</li> </ul>
-------------------	--

## **6. CONTENIDO TEMÁTICO (TEÓRICO-PRÁCTICO)**

### **UNIDAD 1. CONCEPTOS BASICOS**

- 1.1 Conceptos básicos de programación, algoritmos y lenguajes (Algoritmo, Programación, Programa, Programador, Traductores, Intérpretes, Compiladores, Palabras reservadas, Código Fuente, Código objeto, Código ejecutable, Sistema informático o aplicación, Usuario, Lenguajes de alto y bajo nivel)
- 1.2 Fases de la solución de problemas (Creación de un programa informático)

### **UNIDAD 2. METODOLOGIA Y DISEÑO DE ALGORITMOS**

- 2.1 Descripción narrada
- 2.2 Diagramas de flujo
- 2.3 Diagramas NS
- 2.4 Pseudocódigo

### **UNIDAD 3. ESTRUCTURA DE CONTROL SECUENCIAL**

- 3.1 Tipos de datos Simples
- 3.2 Tipos de datos Compuestos
- 3.3 Variables y Constantes
- 3.4 Reglas para construir identificadores
- 3.5 Asignación de datos
  - 3.5.1 Asignación aritmética
  - 3.5.2 Asignación lógica
  - 3.5.3 Asignación de cadena de caracteres
  - 3.5.4 Práctica de Asignación
- 3.6 Expresiones Aritméticas
  - 3.6.1 Operador y operando
  - 3.6.2 Operadores aritméticos
  - 3.6.3 Reglas de prioridad de operadores aritméticos
  - 3.6.4 Funciones aritméticas predefinidas
- 3.7 Expresiones lógicas (booleanas)
- 3.8 Práctica de expresiones
- 3.9 Funciones predefinidas de cadena
- 3.10 Práctica de funciones predefinidas

### **UNIDAD 4. ESTRUCTURAS DE SELECCION**

- 4.1. Estructura de Selección Simple (If-Then)
- 4.2. Estructura de Selección Doble (If-Then-Else)
- 4.3 Estructura de Selección Anidada
- 4.4 Estructura de Selección múltiple (Case)

## UNIDAD 5. ESTRUCTURAS DE CONTROL REPETITIVA

- 5.1. Estructura de control repetitiva no condicionada (For)
- 5.2. Estructura de control repetitiva condicionada (While)
- 5.3. Estructura de control repetitiva condicionada (Repeat, While)
- 5.4. Arreglos

## 7. ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE

Para el desarrollo de esta unidad académica se estará trabajando bajo el esquema de aprendizaje basado en evidencias: Los alumnos elaborarán ejercicios prácticos sobre la materia, los que al final conformarán un proyecto mayor.

## 8. EVALUACIÓN DEL APRENDIZAJE

8.1. Evidencias de aprendizaje	8.2. Criterios de desempeño
Prácticas y tareas	Cumplir con las instrucciones y los requerimientos técnicos solicitados.
Exámenes	Desarrollar con solvencia ejercicios planteados así como dar respuesta correcta a los cuestionamientos teóricos sobre los temas vistos en clase.
Trabajo final	Consiste en la resolución de un problema que integra todos los conocimientos adquiridos en la materia. Deberá contener todos los puntos solicitados. Cumplir con las instrucciones y los requerimientos técnicos solicitados. Deberá dar solución al problema planteado.
Actividades extracurriculares	<ul style="list-style-type: none"><li>• En base a la realización de tres actividades que pueden ser deportivas, culturales o recreativas cada una de ellas con una reflexión breve de media cuartilla de cómo le abona a su formación como futuros Ingenieros en Geofísica.</li><li>• En lugar de las tres actividades puede ser un taller ofertado en CUSur o externo a él, entregando al final una reflexión en congruencia con su formación académica.</li><li>• Para que los 5 puntos de formación integral sean válidos, el alumno debe tener al menos 60 puntos en los demás criterios del apartado de calificación final.</li></ul>

## 9. CALIFICACIÓN

<b>Prácticas y tareas</b>	40 %
<b>Tareas</b>	10 %
<b>Exámenes</b>	20 %
<b>Proyecto Final</b>	30 %
<b>Total:</b>	<b>100%</b>

**Nota:** Se agregaran 5 puntos a la calificación por concepto de actividades de Formación Integral.

## 10. ACREDITACIÓN

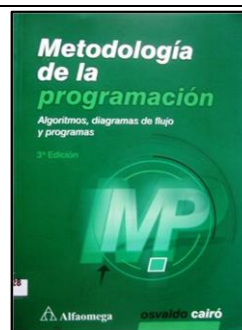
<b>Periodo ordinario.</b> De conformidad con el artículo 20 del Reglamento General de Evaluación y Promoción	<b>Periodo extraordinario.</b> De conformidad con el artículo 27 del Reglamento General de Evaluación y
--	---

<p>de Alumnos de la Universidad de Guadalajara, para que el alumno tenga derecho al registro del resultado final de la evaluación en el periodo ordinario, establecido en el calendario escolar aprobado por el Consejo General Universitario, se requiere:</p> <ol style="list-style-type: none"> <li>I. Estar inscrito en el plan de estudios y curso correspondiente, y</li> <li>II. Tener un mínimo de asistencia del 80% a clases y actividades registradas durante el curso.</li> </ol>	<p>Promoción de Alumnos de la Universidad de Guadalajara, para que el alumno tenga derecho al registro de la calificación en el periodo extraordinario, se requiere:</p> <ol style="list-style-type: none"> <li>I. Estar inscrito en el plan de estudios y curso correspondiente.</li> <li>II. Haber pagado el arancel y presentar el comprobante correspondiente.</li> <li>III. Tener un mínimo de asistencia del 65% a clases y actividades registradas durante el curso.</li> </ol> <p>Se exceptúan de este caso las materias de orden práctico que requerirán la repetición del curso (Art. 23 RGEYPA).</p>
---	---

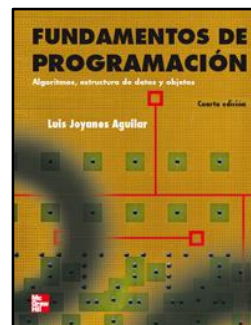
## 11. BIBLIOGRAFÍA

### BIBLIOGRAFÍA BÁSICA

Cairó Battistutti, Osvaldo. (2005)  
**Metodología de la programación. Algoritmos, diagramas de flujo y programas.**  
 México, Alfaomega  
 ISBN: 970-15-1100-X  
 Clasificación: **005.1 CAI 2005**  
*5 ejemplares en biblioteca*



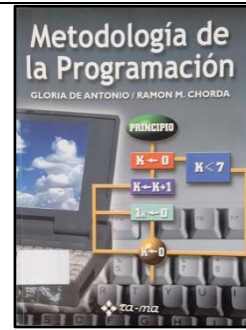
Joyanes Aguilar, Luis. (2008)  
**Fundamentos de programación. Algoritmos, estructura de datos y objetos.**  
 Madrid, McGraw-Hill  
 ISBN: 978-84-481-6111-8  
 Clasificación: **005.1 JOY 2008**  
*13 ejemplares en biblioteca*



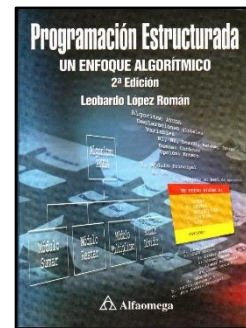
Sznajdleder, Pablo Augusto. (2012)  
**Algoritmos a fondo con implementaciones en C y Java.**  
 Buenos Aires, Alfaomega  
 ISBN: 978-987-1609-37-6  
 Clasificación: **005.1 SZN 2012**  
*10 ejemplares en biblioteca*



De Antonio, Gloria; Chorda, Ramón M. (1997)  
**Metodología de la programación con ejemplos prácticos.**  
Madrid, Ra-Ma  
ISBN: 84-86381-69-X  
Clasificación: **005.1 ANT 1997**  
*4 ejemplares en biblioteca*



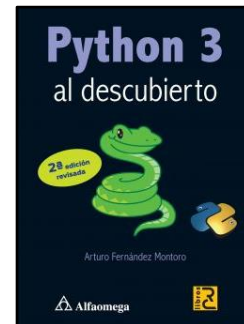
López Román, Leobardo. (2003)  
**Programación estructurada. Un enfoque algorítmico**  
México, Alfaomega  
ISBN: 970-15-0856-4  
Clasificación: **005.13 LOP 2003**  
*5 ejemplares en biblioteca*



Heileman, Gregory L. (1998)  
**Estructuras de datos algoritmos y programación orientada a objetos.**  
Madrid, McGraw-Hill  
ISBN: 84-481-1173-7  
Clasificación: **005.73 HEI 1998**  
*5 ejemplares en biblioteca*



Fernández Montoro, Arturo. (2012)  
**Python 3 al descubierto**  
México D.F., Alfaomega  
ISBN: 978-607-707-470-0  
Clasificación: **005.133 FER 2012**  
*5 ejemplares en biblioteca*



Pérez Castaño, Arnaldo. (2016)  
**Python fácil**  
México, Alfaomega  
ISBN: 978-607-622-661-2  
Clasificación: **005.133 PER 2016**  
*5 ejemplares en biblioteca*



## BIBLIOGRAFÍA COMPLEMENTARIA

García Sánchez, Luis; et al. (2002)

**Construcción lógica de programas. Teoría y problemas resueltos**

México, Alfaomega

ISBN: 970-15-0794-0

Clasificación: **005.1 CON**

*3 ejemplares en biblioteca*



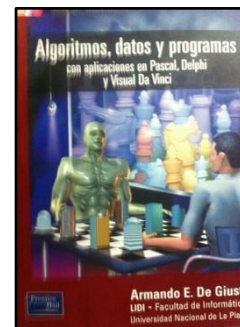
Armando E. De Giusti. (2001)

**Algoritmos, datos y programas con aplicaciones en Pascal, Delphi y Visual.**

Buenos Aires, Prentice Hall / Pearson

Clasificación: **003.1 ALG 2001**

*1 ejemplar en biblioteca*



Thomas H. Cormen. (2001)

**Introduction to algorithms. Second Edition**

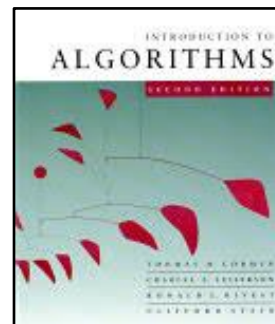
Massachusetts, McGraw Hill

ISBN: 0-262-03293-7

ISBN: 0-07-013151-1

Clasificación: **005.1 INT 2001**

*1 ejemplar en biblioteca*



R. C. T. Lee; et al. (2007)

**Introducción al diseño y análisis de algoritmos. Un enfoque estratégico.**

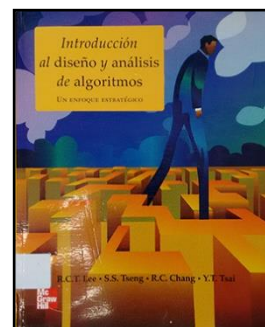
México, McGraw-Hill

ISBN-13: 978-970-10-6124-4

ISBN-10: 970-10-6124-1

Clasificación: **005.1 INT 2007**

*2 ejemplares en biblioteca*





Joyanes Aguilar, Luis. (2000)

**Programación en C++. Algoritmos, estructuras de datos y objetos.**

España, McGraw-Hill

ISBN: 84-481-2487-1

Clasificación: **005.369 JOY 2000**

**5 ejemplares en biblioteca**



Drozdek, Adam. (2007)

**Estructura de datos y algoritmos en Java.**

México, International Thomson Editores

ISBN-13: 978-970-686-611-0

ISBN-10: 970-686-611-6

Clasificación: **005.13 DRO 2007**

**2 ejemplares en biblioteca**



Joyanes Aguilar, Luis. (2002)

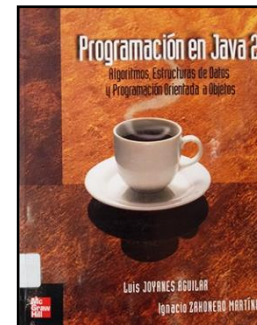
**Programación de Java 2. Algoritmos, estructuras de datos y programación orientada a objetos.**

Madrid, McGraw Hill

ISBN: 84-481-3290-4

Clasificación: **005.13 JOY 2002**

**3 ejemplares en biblioteca**



Aho, Alfred V. (1998)

**Estructuras de datos y algoritmos.**

Naucalpan de Juárez, Edo. de México, Addison Wesley Longman

ISBN: 968-444-345-5

Clasificación: **005.73 AHO 1998**

**3 ejemplares en biblioteca**



Lewis, John. (2006)

**Estructuras de datos con Java. Diseño de estructuras y algoritmos.**

Madrid, Pearson Addison Wesley

ISBN: 84-205-5034-5

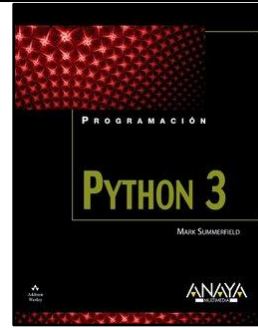
Clasificación: **005.73 LEW 2006**

**5 ejemplares en biblioteca**





Summerfield, Mark. (2009)  
**Python 3**  
Madrid, Anaya Multimedia  
ISBN: 978-84-415-2613-6  
Clasificación: **005.13 SUM 2009**  
*4 ejemplares en biblioteca*



**12. RECURSOS COMPLEMENTARIOS (páginas web, mooc's, plataformas, objetos de aprendizaje)**

**Firma:**

**Mtro. Gerardo Jiménez Haro**

---

**Presidente de Academia**

**Vo.Bo**

**Dr. Jorge Lozoya Arandía**

---

**Jefe de Departamento**