



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE INGENIERÍA



PROGRAMA DE ESTUDIO

FUNDAMENTOS DE PROGRAMACIÓN

1

10

Asignatura

Clave

Semestre

Créditos

INGENIERÍA ELÉCTRICA

INGENIERÍA
EN COMPUTACIÓN

INGENIERÍA
EN COMPUTACIÓN

División

Departamento

Licenciatura

Asignatura:

Obligatoria ☒

Optativa ☐

Horas/semana:

Teóricas

Prácticas

Total

Horas/semestre:

Teóricas

Prácticas

Total

Modalidad: Curso teórico-práctico

Seriación obligatoria antecedente: Ninguna

Seriación obligatoria consecuente: Estructura de Datos y Algoritmos I

Objetivo(s) del curso:

El alumno resolverá problemas aplicando los fundamentos de programación para diseñar programas en el lenguaje estructurado C, apoyándose en metodologías para la solución de problemas.

Temario

NÚM.	NOMBRE	HORAS
1.	Panorama general	2.0
2.	Resolución de problemas	20.0
3.	Fundamentos para la construcción de código a partir del algoritmo	24.0
4.	Paradigmas de programación	10.0
5.	Cómputo aplicado a diferentes áreas de la ingeniería y otras disciplinas	8.0
		64.0
	Actividades prácticas	32.0
	Total	96.0

1 Panorama general

Objetivo: El alumno definirá la importancia de la programación como herramienta en el quehacer del ingeniero.

Contenido:

- 1.1 Evolución de la programación.
- 1.2 Beneficios de la programación (a la sociedad, a la industria, a la medicina, entre otros).
- 1.3 Algoritmos en la solución de problemas y sus retos.
- 1.4 Explicar el propósito y el papel de los fundamentos de la programación en la ingeniería.

2 Resolución de problemas

Objetivo: El alumno resolverá problemas mediante la especificación algorítmica.

Contenido:

- 2.1 Definición, planteamiento y modelado del problema.
 - 2.1.1 Formular el problema.
 - 2.1.2 Analizar el problema.
 - 2.1.3 Diseñar una estrategia de búsqueda de la solución.
- 2.2 Algoritmos para la resolución del problema.
 - 2.2.1 Definición y representación de algoritmos.
 - 2.2.2 Conversión del planteamiento del problema al algoritmo.
- 2.3 Definición del modelo computacional.
 - 2.3.1 Máquina de Von Neuman.
 - 2.3.2 Máquina de Turing.
- 2.4 Refinamiento del algoritmo paso a paso.
 - 2.4.1 Planteamiento de la solución del problema.
 - 2.4.2 Descomposición de la solución del problema en submódulos.
 - 2.4.3 Aplicación de las estructuras básicas de control: secuencial, condicional e iterativo.

3 Fundamentos para la construcción de código a partir del algoritmo

Objetivo: El alumno construirá programas utilizando el lenguaje de programación C a través de un análisis y modelado algorítmico previo.

Contenido:

- 3.1 Sintaxis básica y semántica.
- 3.2 Variables, tipos, expresiones y asignación.
- 3.3 Estructuras de control condicional e iterativo.
- 3.4 Funciones y paso de parámetros.
- 3.5 Descomposición estructurada.
- 3.6 Manejo de E/S.
- 3.7 Estrategias de depuración.
 - 3.7.1 Tipo de errores.
 - 3.7.2 Técnicas de depuración.

4 Paradigmas de programación

Objetivo: El alumno distinguirá los diversos paradigmas de programación; y seleccionará el uso de ellas de acuerdo

con las características y tipo de problemas por resolver.

Contenido:

- 4.1 Programación estructurada.
- 4.2 Programación orientada a objetos.
- 4.3 Programación lógica.
- 4.4 Programación paralela.
- 4.5 Principales usos de los paradigmas para la solución de problemas.
- 4.6 Nuevas tendencias.

5 Cómputo aplicado a diferentes áreas de la ingeniería y otras disciplinas

Objetivo: El alumno identificará la aplicación del cómputo para la solución de problemas en las diferentes áreas disciplinares.

Contenido:

- 5.1 Tendencia de desarrollo de software.
 - 5.1.1 Software propietario.
 - 5.1.2 Software libre.
- 5.2 Aplicaciones.
 - 5.2.1 Ciencias físicas y de la ingeniería.
 - 5.2.2 Ciencias médicas y de la salud.
 - 5.2.3 Leyes, ciencias sociales y del comportamiento.
 - 5.2.4 Artes y humanidades.
 - 5.2.5 Otras disciplinas.

Bibliografía básica

Temas para los que se recomienda:

BROOKSHEAR, J. Gleen

Computer Science: An Overview

11th edition

Boston

Prentice Hall, 2011

Todos

CAIRÓ, Osvaldo

Metodología de la Programación. Algoritmos, Diagramas de

Flujo y Programas 2a. edición

México

Alfaomega, 2003

Tomos I y II

Todos

FELLEISEN, Matthias, FINDLET, Robert Bruce, et al.

How to Design Programs. An Introduction to Programming and

Computing Cambridge

MIT Press, 2001

Todos

HOROWITZ, Ellis

Computer Algorithms

Todos

2nd edition
Summit, NJ
Silicon Press, 2007

KERNIGHAN, Brian W., PIKE, Rob
*The Practice of Programming (Addison-Wesley Professional
Computing Series)* New Jersey
Addison-Wesley, 1994

Todos

KERNIGHAN, Brian, RITCHIE, Dennis
C Programming Language
2nd edition
New Jersey
Prentice Hall, 1988

Todos

MCCONNELL, Steve
Code Complete 2
2nd edition
Redmond, WA
Microsoft Press, 2004

Todos

SZNAJDLEDER, Pablo
Algoritmos a fondo: con implementación en C y JAVA
Buenos Aires
Alfaomega, 2012

Todos

VOLAND, Gerard
Engineering by Design
2nd edition
Upper Saddle River, NJ
Prentice Hall, 2003

Todos

Bibliografía complementaria

Temas para los que se recomienda:

ALLEN, Tucker, ROBERT, Noonan
Programming Languages
2nd edition
New Jersey
McGraw-Hill, 2006

1, 2 y 4

MICHAEL, L. Scott
Programming Language Pragmatics
Third Edition
Cambridge
Morgan Kaufmann, 2009

1, 2 y 3

PETER, Sestoft

*Programming Language Concepts (Undergraduate Topics in
Computer Science* Copenhagen

Springer, 2012

1, 2 y 3

Sugerencias didácticas

Exposición oral	<input checked="" type="checkbox"/>
Exposición audiovisual	<input checked="" type="checkbox"/>
Ejercicios dentro de clase	<input checked="" type="checkbox"/>
Ejercicios fuera del aula	<input checked="" type="checkbox"/>
Seminarios	<input checked="" type="checkbox"/>
Uso de software especializado	<input type="checkbox"/>
Uso de plataformas educativas	<input type="checkbox"/>

Lecturas obligatorias	<input checked="" type="checkbox"/>
Trabajos de investigación	<input checked="" type="checkbox"/>
Prácticas de taller o laboratorio	<input checked="" type="checkbox"/>
Prácticas de campo	<input checked="" type="checkbox"/>
Búsqueda especializada en internet	<input type="checkbox"/>
Uso de redes sociales con fines académicos	<input type="checkbox"/>

Forma de evaluar

Exámenes parciales	<input checked="" type="checkbox"/>
Exámenes finales	<input checked="" type="checkbox"/>
Trabajos y tareas fuera del aula	<input checked="" type="checkbox"/>

Participación en clase	<input checked="" type="checkbox"/>
Asistencia a prácticas	<input checked="" type="checkbox"/>

Perfil profesiográfico de quienes pueden impartir la asignatura

Licenciatura en Ingeniería en Computación, Ciencias de Computación, Matemáticas Aplicadas o una carrera similar. Deseable haber realizado estudios de posgrado, contar con conocimientos y experiencia en el diseño de algoritmos y programas del paradigma estructurado, contar con experiencia docente o haber participado en cursos o seminario de iniciación en la práctica docente.