

PROGRAMACIÓN I

CLAVE:		SECTOR :	BÁSICO
SEMESTRE:	2	ÁREA:	INFORMÁTICA
CRÉDITOS:	10	SERIACIÓN:	ASIGNATURA PRECEDENTE INDICATIVA: Cálculo Diferencial e Integral I, Álgebra Superior I y Geometría Analítica I. ASIGNATURA SUBSECUENTE INDICATIVA: Programación II
HORAS POR CLASE		TEÓRICA:	1
CLASES POR SEMANA		TEÓRICA:	4
HORAS POR SEMESTRE		TEÓRICA:	64
		PRÁCTICAS:	2
		PRÁCTICAS:	1
		PRÁCTICAS:	32

Objetivos generales: Al final del curso el alumno:

- Conocerá las ideas básicas de la programación y la solución algorítmica de problemas, tales como el diseño descendente, refinamientos sucesivos, abstracción de procedimientos, estructuras de control, tipos de datos y convenciones de entrada y salida básicos.
- Explicará las características sintácticas y de ejecución de un lenguaje de programación moderno, y las aplicará en la construcción y ejecución de programas completos que resuelven problemas algorítmicos sencillos.
- Identificará las representaciones básicas de datos numéricos y no numéricos directamente en la máquina, así como su aplicación concreta.
- Conocerá los tipos de datos elementales y estructurados, así como los principios necesarios para la creación de tipos de datos definidos por el usuario y sus aplicaciones.
- Comprenderá y aplicará los fundamentos de los algoritmos recursivos en el proceso de solución de problemas.
- Comparará algunos algoritmos para búsquedas y ordenamientos lineales, y conocerá los problemas de complejidad y la contraposición entre espacio requerido y tiempo de ejecución.
- Reconocerá el contexto social en el cual la computación interacciona y sirve a la sociedad.

Tema 1. Conceptos fundamentales en métodos para la solución de problemas

Explicará las ideas básicas de programación y solución algorítmica de problemas. **19 horas teóricas**
9 horas teórico-prácticas

- 1.1. Abstracción de procedimientos; parámetros.
- 1.2. Estructuras de control: selección, iteración, recursividad.
- 1.3. Tipos de datos (i.e. números, cadenas, booleanos) y sus usos en la solución de problemas.
- 1.4. El proceso de diseño de programas; desde la especificación hasta la instrumentación; refinamientos sucesivos; representación gráfica.

Tema 2. Introducción a un lenguaje de programación

12 horas teóricas
7 horas teórico-prácticas

Comprenderá las características y la utilización de un lenguaje de programación.

- 2.1. Declaración de tipos básicos.
- 2.2. Operadores aritméticos y de asignación.
- 2.3. Enunciados condicionales.
- 2.4. Iteraciones y recursividad.
- 2.5. Procedimientos, funciones y parámetros.
- 2.6. Arreglos y registros.
- 2.7. Estructura general de un programa.

Tema 3. Representación de datos a nivel de la máquina

Reconocerá las representaciones básicas de datos en máquina.

6 horas teóricas
3 horas teórico-prácticas

- 3.1. Representación de datos numéricos, esto es, binario, octal, hexadecimal, punto fijo, complemento a 1 y 2, con signo, punto flotante, decimal, BCD, XS3.
- 3.2. Datos no numéricos como alfanuméricos, ASCII, ISO.

Tema 4. Representación de tipos de datos

Explicará las características de los distintos tipos de datos y la manera de crearlos.

6 horas teóricas
3 horas teórico-prácticas

- 4.1 Selección y representación de tipos de datos elementales: enteros, reales, booleanos, carácter.
- 4.2 Especificación y representación de tipos de datos estructurados: arreglos, registros, conjuntos.

Tema 5. Algoritmos recursivos

Reconocerá los fundamentos y usos de los algoritmos recursivos y su aplicación en la solución de problemas.

8 horas teóricas
4 horas teórico-prácticas

- 5.1 Introducción a algoritmos recursivos.
- 5.2 Conexión con la inducción matemática.
- 5.3 Comparación entre algoritmos recursivos e iterativos.

Tema 6. Búsquedas y ordenamientos lineales

Comparará algunos algoritmos e identificará algunos problemas importantes.

8 horas teóricas
4 horas teórico-prácticas

- 6.1 Algoritmo de ordenamientos de selección e inserción en arreglos a través de apuntadores, con asignación dinámica de memoria; complejidad en el tiempo y el espacio; mejores y peores casos.
- 6.2 Búsqueda lineal, búsqueda binaria y arboles binarios de búsqueda; complejidad en el tiempo y el espacio; mejores y peores casos.

Tema 7. Contexto histórico y social de la computación

Explicará el contexto social en el que se desenvuelve la computación, y los compromisos de esta con la sociedad.

5 horas teóricas
2 horas teórico-prácticas

- 7.1 Contexto social e histórico de la computación.
- 7.2 Definición de su área de estudio y actividades profesionales.
- 7.3 Similitudes y diferencias con otras disciplinas científicas y profesionales.
- 7.4 Uso, mal uso y límites de la tecnología computacional.
- 7.5 Responsabilidades sociales (seguridad y privacidad).
- 7.6 Tipos de riesgos: errores latentes, seguridad y privacidad, mal uso, etcétera.

Bibliografía básica:

- Tucker, A.B. *Fundamental of Computing, I: Logic, Problem Solving, Programs and Computers*. 2nd edition. USA. McGraw-Hill. 1994.

- Warfords, S. J. *Computer Science*. USA. D.C. Heat and Company. 1991.

Bibliografía complementaria:

- Ledgard H. with Tauer J. *Professional Software Engineering Concepts*. Vol.1. USA. Addison Wesley Publishing Company. 1987.
- Ledgard H. with Tauer J. *Professional Software, Programming Practice*. Vol. II. USA. Addison Wesley Publishing Company. 1987.
- Naur, P. *A Human Activity*. USA. ACM Press. Addison Wesley Publishing Company. 1992.
- Salmon, W.I. *Structures and Abstractions: An Introduction to Computer Science with Pascal*. USA. Richard D. Irwing, Inc. 1991.
- Warnier, J.D. *Logical Construction of Systems*. USA. Van Nostrand Reinhold Company. 1981.
- Ioren H. *Introduction to Computer Architecture and Organization*. 2nd edition. USA. John Wiley and Sons. Inc. 1989.

Sugerencias didácticas:

Se recomiendan:

Tareas semanales en las cuales el alumno aplique el material visto en clase.

Trabajos de investigación bibliográfica para que el alumno amplíe sus conocimientos y conozca diferentes enfoques del material del curso.

Prácticas de cómputo para la experimentación con los algoritmos vistos en clase.

Análisis y resolución de casos prácticos.

Forma de evaluación:

Se recomiendan de 3 a 4 exámenes parciales y un examen final, así como la realización de tareas sobre los temas vistos en clase para reforzar los conocimientos teóricos adquiridos.

Perfil profesiográfico:

Egresado preferentemente de las licenciaturas en Ciencias de la Computación, Ingeniería en Sistemas, Matemáticas, Actuaría o alguna afín con experiencia docente y profesional en el área.