

**Universidad Católica San Pablo**  
**Facultad de Ingeniería y Computación**  
**Escuela Profesional de**  
**Ciencia de la Computación**  
**SILABO**



**CS103O. Algoritmos y Estructuras de Datos**  
**(Obligatorio)**

2010-1

**1. DATOS GENERALES**

1.1 CARRERA PROFESIONAL	:	Ciencia de la Computación
1.2 ASIGNATURA	:	CS103O. Algoritmos y Estructuras de Datos
1.3 SEMESTRE ACADÉMICO	:	4 <sup>to</sup> Semestre.
1.4 PREREQUISITO(S)	:	CS102O. Objetos y Abstracción de Datos. (3 <sup>er</sup> Sem)
1.5 CARÁCTER	:	Obligatorio
1.6 HORAS	:	2 HT; 2 HP; 2 HL;
1.7 CRÉDITOS	:	4

**2. DOCENTE**

**3. FUNDAMENTACIÓN DEL CURSO**

El fundamento teórico de todas las ramas de la informática descansa sobre los algoritmos y estructuras de datos, este curso brindará a los participantes una introducción a estos temas, formando así una base que servirá para los siguientes cursos en la carrera.

**4. SUMILLA**

1. PF/Estructuras de Datos.2. PF/Recursividad.3. AL/Algoritmos Fundamentales.4. Grafos 5. Matrices Esparzas 6. Arboles Equilibrados

**5. OBJETIVO GENERAL**

- Hacer que el alumno entienda la importancia de los algoritmos para la solución de problemas.
- Introducir al alumno hacia el campo de la aplicación de las estructuras de datos.

## 6. CONTRIBUCIÓN A LA FORMACIÓN PROFESIONAL Y FORMACIÓN GENERAL

Esta disciplina contribuye al logro de los siguientes resultados de la carrera:

- a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. [Nivel Bloom: 4]
- b) Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución. [Nivel Bloom: 4]
- c) Diseñar, implementar y evaluar un sistema, proceso, componente o programa computacional para alcanzar las necesidades deseadas. [Nivel Bloom: 3]
- d) Trabajar efectivamente en equipos para cumplir con un objetivo común. [Nivel Bloom: 3]
- i) Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. [Nivel Bloom: 4]
- j) Aplicar la base matemática, principios de algoritmos y la teoría de la Ciencia de la Computación en el modelamiento y diseño de sistemas computacionales de tal manera que demuestre comprensión de los puntos de equilibrio involucrados en la opción escogida. [Nivel Bloom: 4]
- k) Aplicar los principios de desarrollo y diseño en la construcción de sistemas de software de complejidad variable. [Nivel Bloom: 3]

## 7. CONTENIDOS

### UNIDAD 1: PF/Estructuras de Datos.(8 horas)

Nivel Bloom: 5

#### OBJETIVO GENERAL

#### CONTENIDO

- Describir la representación de datos numéricos y de caracteres
- Entender como la precisión y el redondeo puede afectar los cálculos numéricos.
- Discutir la representación y uso de tipos de datos primitivos y estructuras de datos incorporadas en el lenguaje.
- Describir aplicaciones comunes para cada estructura de datos en la lista de temas.
- Implementar estructuras de datos definidas por el usuario en un lenguaje de alto nivel.
- Comparar implementaciones alternativas de estructuras de datos considerando su desempeño.
- Escribir programas que usan cada una de las siguientes estructuras de datos: arreglos, registros, cadenas, listas enlazadas, pilas, colas y tablas de *hash*.
- Comparar y contrastar los costos y beneficios de las implementaciones dinámicas y estáticas de las estructuras de datos.
- Escoger la estructura de datos apropiada para modelar un problema dado.

- Representación de datos numéricos
- Rango, precisión y errores de redondeo.
- Arreglos.
- Registros.
- Cadenas y procesamiento de cadenas.
- Representación de caracteres.
- Administración del almacenamiento en tiempo de ejecución.
- Punteros y referencias.
- Estructuras enlazadas.
- Estrategias de implementación para pilas, colas y tablas *hash*.
- Estrategias de implementación para grafos y árboles.
- Estrategias para escoger la estructura de datos correcta.

**Lecturas:** [Cormen et al., 2009], [Fager et al., 2014]

UNIDAD 2: PF/Recursividad.(4 horas)	
Nivel Bloom: 5	
OBJETIVO GENERAL	CONTENIDO
<ul style="list-style-type: none"> <li>▪ Describir el concepto de recursividad y dar ejemplos de su uso.</li> <li>▪ Identificar el caso base y el caso general de un problema definido recursivamente.</li> <li>▪ Comparar soluciones iterativas y recursivas para problemas elementales tal como factorial.</li> <li>▪ Describir la técnica dividir y conquistar.</li> <li>▪ Implementar, probar y depurar funciones y procedimientos recursivos simples.</li> <li>▪ Describir como la recursividad puede ser implementada usando una pila.</li> <li>▪ Discutir problemas para los cuales el <i>backtracking</i> es una solución apropiada.</li> <li>▪ Determinar cuando una solución recursiva es apropiada para un problema.</li> </ul>	<ul style="list-style-type: none"> <li>▪ El concepto de recursividad.</li> <li>▪ Funciones matemáticas recursivas.</li> <li>▪ Funciones recursivas simples.</li> <li>▪ Estrategias de dividir y conquistar.</li> <li>▪ <i>Backtracking</i> recursivo.</li> </ul>
<b>Lecturas:</b> [Cormen et al., 2009], [Fager et al., 2014]	

UNIDAD 3: AL/Algoritmos Fundamentales.(12 horas)	
Nivel Bloom: 4	
OBJETIVO GENERAL	CONTENIDO
<ul style="list-style-type: none"> <li>▪ Implementar los algoritmos cuadráticos más comunes y los algoritmos de ordenamiento <math>O(N\log N)</math>.</li> <li>▪ Diseñar e implementar una función de (<i>hash</i>) apropiada para una aplicación.</li> <li>▪ Diseñar e implementar un algoritmo de resolución de colisiones para tablas de <i>hash</i>.</li> <li>▪ Discutir la eficiencia computacional de los principales algoritmos de ordenamiento, búsqueda y (<i>hashing</i>).</li> <li>▪ Discutir otros factores, además de la eficiencia computacional, que influyen en la elección de los algoritmos, tales como tiempo de programación, mantenimiento y el uso de patrones específicos de aplicación en los datos de entrada.</li> <li>▪ Resolver problemas usando los algoritmos de grafos fundamentales, incluyendo búsqueda por amplitud y profundidad; caminos más cortos con uno y múltiples orígenes, cerradura transitiva, ordenamiento topológico y al menos un algoritmo de árbol de expansión mínima.</li> <li>▪ Demostrar las siguientes capacidades: evaluar algoritmos, seleccionar una opción de un rango posible, proveer una justificación para tal elección e implementar el algoritmo..</li> </ul>	<ul style="list-style-type: none"> <li>▪ Algoritmos numéricos simples.</li> <li>▪ Búsqueda secuencial y binaria.</li> <li>▪ Algoritmos cuadráticos de ordenamiento (selección, inserción).</li> <li>▪ Algoritmos de tipo <math>O(N\log N)</math> (Quicksort, heapsort, mergesort).</li> <li>▪ Tablas de (<i>hash</i>) incluyendo estrategias de solución para las colisiones.</li> <li>▪ Árboles de búsqueda binaria.</li> <li>▪ Representación de grafos (Listas y Matrices de adyacencia).</li> <li>▪ Recorridos por amplitud y profundidad.</li> <li>▪ El algoritmo del camino más corto (algoritmos de Dijkstra y Floyd).</li> <li>▪ Cerradura transitiva (algoritmo de Floyd).</li> <li>▪ Árbol de expansión mínima (algoritmos de Kruskal y Prim).</li> <li>▪ Ordenamiento Topológico.</li> </ul>
<b>Lecturas:</b> [Cormen et al., 2009], [Fager et al., 2014]	

UNIDAD 4: Grafos (12 horas)	
Nivel Bloom: 5	
OBJETIVO GENERAL	CONTENIDO
<ul style="list-style-type: none"> <li>▪ Adquirir destreza para realizar una implementación correcta.</li> <li>▪ Desarrollar los conocimientos para decidir cuando es mejor usar una técnica de implementación que otra.</li> </ul>	<ul style="list-style-type: none"> <li>▪ Concepto de Grafos.</li> <li>▪ Grafos Dirigidos y Grafos no Dirigidos.</li> <li>▪ Utilización de los Grafos.</li> <li>▪ Medida de la Eficiencia. En tiempo y espacio.</li> <li>▪ Matrices de Adyacencia.</li> <li>▪ Matrices de Adyacencia etiquetada.</li> <li>▪ Listas de Adyacencia.</li> <li>▪ Implementación de Grafos usando Matrices de Adyacencia.</li> <li>▪ Implementación de Grafos usando Listas de Adyacencia.</li> <li>▪ Inserción, Búsqueda y Eliminación de nodos y aristas.</li> <li>▪ Algoritmos de búsqueda en grafos.</li> </ul>
<b>Lecturas:</b> [Cormen et al., 2009], [Fager et al., 2014]	

UNIDAD 5: Matrices Esparzas (8 horas)	
Nivel Bloom: 5	
OBJETIVO GENERAL	CONTENIDO
<ul style="list-style-type: none"> <li>▪ Comprender el uso y implementación de matrices esparzas.</li> </ul>	<ul style="list-style-type: none"> <li>▪ Conceptos Iniciales.</li> <li>▪ Matrices poco densas</li> <li>▪ Medida de la Eficiencia en Tiempo y en Espacio</li> <li>▪ Creación de la matriz esparza estática vs Dinámicas.</li> <li>▪ Métodos de inserción, búsqueda y eliminación</li> </ul>
<b>Lecturas:</b> [Cormen et al., 2009], [Fager et al., 2014]	

UNIDAD 6: Árboles Equilibrados (16 horas)	
Nivel Bloom: 5	
OBJETIVO GENERAL	CONTENIDO
<ul style="list-style-type: none"> <li>▪ Comprender las funciones básicas de estas estructuras complejas con el fin de adquirir la capacidad para su implementación.</li> </ul>	<ul style="list-style-type: none"> <li>▪ Árboles AVL.</li> <li>▪ Medida de la Eficiencia.</li> <li>▪ Rotaciones Simples y Compuestas</li> <li>▪ Inserción, Eliminación y Búsqueda.</li> <li>▪ Árboles B , B+ B* y Patricia.</li> </ul>
<b>Lecturas:</b> [Cormen et al., 2009], [Fager et al., 2014]	

## 8. METODOLOGÍA

El profesor del curso presentará clases teóricas de los temas señalados en el programa propiciando la intervención de los alumnos.

El profesor del curso presentará demostraciones para fundamentar clases teóricas.

El profesor y los alumnos realizarán prácticas

Los alumnos deberán asistir a clase habiendo leído lo que el profesor va a presentar. De esta manera se facilitará la comprensión y los estudiantes estarán en mejores condiciones de hacer consultas en clase.

## 9. EVALUACIONES

**Evaluación Permanente 1** : 20 %

**Examen Parcial** : 30 %

**Evaluación Permanente 2** : 20 %

**Examen Final** : 30 %

## Referencias

[Cormen et al., 2009] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms*. MIT Press, third edition edition. ISBN: 978-0-262-53305-8.

[Fager et al., 2014] Fager, J., Yépez, W. L. P., Villacrés, M., Martínez, L. A. P., Ochoa, D., and Cuadros-Vargas, E. (2014). *Estructura de datos*. Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIn), first edition edition.