



**Universidad de Ingeniería y Tecnología**  
**Escuela Profesional de**  
**Ciencia de la Computación**  
**Silabo del curso – Periodo Académico 2017-II**

1. **Código del curso y nombre:** CS210. Algoritmos y Estructuras de Datos
2. **Créditos:** 4
3. **Horas de Teoría y Laboratorio:** 2 HT; 4 HP;
4. **Docente(s)**

Atención previa coordinación con el profesor

**5. Bibliografía**

- [Cor+09] Thomas H. Cormen et al. *Introduction to Algorithms*. Third Edition. ISBN: 978-0-262-53305-8. MIT Press, 2009.
- [Fag+14] José Fager et al. *Estructura de datos*. First Edition. Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIN), 2014.

**6. Información del curso**

- (a) **Breve descripción 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.
- (b) **Prerrequisitos:** CS113. Programación Orientada a Objetos II. (3<sup>er</sup> Sem)
- (c) **Tipo de Curso:** Obligatorio

**7. Competencias**

- 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.

**8. Contribución a los resultados (*Outcomes*)**

- a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (**Usar**)
- b) Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución. (**Usar**)
- c) Diseñar, implementar y evaluar un sistema, proceso, componente o programa computacional para alcanzar las necesidades deseadas. (**Usar**)
- 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. (**Usar**)
- k) Aplicar los principios de desarrollo y diseño en la construcción de sistemas de software de complejidad variable. (**Usar**)

**9. Competencias (IEEE)**

- C1.** La comprensión intelectual y la capacidad de aplicar las bases matemáticas y la teoría de la informática (*Computer Science*).⇒ **Outcome a**
- C1.** La comprensión intelectual y la capacidad de aplicar las bases matemáticas y la teoría de la informática (*Computer Science*).⇒ **Outcome j**

**C2.** Capacidad para tener una perspectiva crítica y creativa para identificar y resolver problemas utilizando el pensamiento computacional.⇒ **Outcome b**

**C5.** Capacidad para implementar algoritmos y estructuras de datos en el software.⇒ **Outcome c**

**CS2.** Identificar y analizar los criterios y especificaciones apropiadas a los problemas específicos, y planificar estrategias para su solución.⇒ **Outcome b**

**CS2.** Identificar y analizar los criterios y especificaciones apropiadas a los problemas específicos, y planificar estrategias para su solución.⇒ **Outcome k**

## **10. Lista de temas a estudiar en el curso**

1. Grafos
2. Matrices Esparzas
3. Árboles Equilibrados

## **11. Metodología y Evaluación**

### **Metodología:**

#### **Sesiones Teóricas:**

El desarrollo de las sesiones teóricas está focalizado en el estudiante, a través de su participación activa, resolviendo problemas relacionados al curso con los aportes individuales y discutiendo casos reales de la industria. Los alumnos desarrollarán a lo largo del curso un proyecto de aplicación de las herramientas recibidas en una empresa.

#### **Sesiones de Laboratorio:**

Las sesiones prácticas se desarrollan en laboratorio. Las prácticas de laboratorio se realizan en equipos para fortalecer su comunicación. Al inicio de cada laboratorio se explica el desarrollo de la práctica y al término se destaca las principales conclusiones de la actividad en forma grupal.

#### **Exposiciones individuales o grupales:**

Se fomenta la participación individual y en equipo para exponer sus ideas, motivándolos con puntos adicionales en las diferentes etapas de la evaluación del curso.

#### **Lecturas:**

A lo largo del curso se proporcionan diferentes lecturas, las cuales son evaluadas. El promedio de las notas de las lecturas es considerado como la nota de una práctica calificada. El uso del campus virtual UTEC Online permite a cada estudiante acceder a la información del curso, e interactuar fuera de aula con el profesor y con los otros estudiantes.

#### **Sistema de Evaluación:**

## **12. Contenido**

<b>Unidad 1: Grafos (12)</b>	
<b>Competences esperadas: C1,C2,C5</b>	
Objetivos de Aprendizaje	Tópicos
<ul style="list-style-type: none"> <li>• Adquirir destreza para realizar una implementación correcta. [Usar]</li> <li>• Desarrollar los conocimientos para decidir cuando es mejor usar una técnica de implementación que otra. [Usar]</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> [Cor+09], [Fag+14]	

<b>Unidad 2: Matrices Esparzas (8)</b>	
<b>Competences esperadas: C1,C2,C5</b>	
Objetivos de Aprendizaje	Tópicos
<ul style="list-style-type: none"> <li>• Comprender el uso y implementacion de matrices esparzas.[Evaluar]</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> [Cor+09], [Fag+14]	

<b>Unidad 3: Arboles Equilibrados (16)</b>	
<b>Competences esperadas: C2,C5,C6</b>	
Objetivos de Aprendizaje	Tópicos
<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. [Evaluar]</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> [Cor+09], [Fag+14]	



**Universidad de Ingeniería y Tecnología**  
**Escuela Profesional de**  
**Ciencia de la Computación**  
**Silabo del curso – Periodo Académico 2017-II**

1. **Código del curso y nombre:** CS211. Teoría de la Computación
2. **Créditos:** 4
3. **Horas de Teoría y Laboratorio:** 2 HT; 4 HP;
4. **Docente(s)**

Atención previa coordinación con el profesor

**5. Bibliografía**

- [Bro93] J. Glenn Brookshear. *Teoría de la Computación*. Addison Wesley Iberoamericana, 1993.
- [HU93] John E. Hopcroft and Jeffrey D. Ullman. *Introducción a la Teoría de Autómatas, Lenguajes y Computación*. CECSA, 1993.
- [Kel95] Dean Kelley. *Teoría de Autómatas y Lenguajes Formales*. Prentice Hall, 1995.
- [Kol97] Ross Kolman Busby. *Estructuras de Matemáticas Discretas para la Computación*. Prentice Hall, 1997.

**6. Información del curso**

- (a) **Breve descripción del curso** Este curso hace énfasis en los lenguajes formales, modelos de computación y computabilidad, además de incluir fundamentos de la complejidad computacional y de los problemas NP completos.
- (b) **Prerrequisitos:** CS1D2. Estructuras Discretas II. (2<sup>do</sup> Sem)
- (c) **Tipo de Curso:** Obligatorio

**7. Competencias**

- Que el alumno aprenda los conceptos fundamentales de la teoría de lenguajes formales.

**8. Contribución a los resultados (Outcomes)**

- a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (**Evaluar**)
- b) Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución. (**Evaluar**)
- 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. (**Evaluar**)

**9. Competencias (IEEE)**

- C8.** Entendimiento de lo que las tecnologías actuales pueden y no pueden lograr.⇒ **Outcome a**
- C9.** Comprensión de las limitaciones de la computación, incluyendo la diferencia entre lo que la computación es inherentemente incapaz de hacer frente a lo que puede lograrse a través de un futuro de ciencia y tecnología.⇒ **Outcome b,j**

**10. Lista de temas a estudiar en el curso**

1. Computabilidad y complejidad básica de autómatas
2. Complejidad Computacional Avanzada