

CIENCIA DE LA COMPUTACIÓN

ALGORITMOS Y ESTRUCTURAS DE DATOS

4 CRÉDITOS



ÍNDICE

ASIGNATURA	3
DATOS GENERALES	3
Ciclo: 4	3
Créditos: cuatro (4) créditos	3
Horas de teoría: dos (2) semanales	3
Horas de práctica: cuatro (4) semanales	3
Duración del período: dieciséis (16) semanas	3
Condición:	3
Modalidad: Virtual	3
Requisitos:	3
PROFESORES	3
Profesor coordinador del curso	3
3.2. Profesor(es) instructor(es) del curso	3
INTRODUCCIÓN AL CURSO	3
OBJETIVOS	4
COMPETENCIAS	4
RESULTADOS DE APRENDIZAJE	5
TEMAS	5
PLAN DE TRABAJO	6
Metodología	6
Sesiones de teoría	6
Sesiones de práctica (laboratorio o taller)	6
SISTEMA DE EVALUACIÓN	7
REFERENCIAS BIBLIOGRÁFICAS	7

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

SILABO 2021-1

1. ASIGNATURA

CS2100 - Algoritmos y Estructuras de Datos

2. DATOS GENERALES

2.1 Ciclo: 4°

2.2 Créditos: cuatro (4) créditos

2.3 Horas de teoría: dos (2) semanales

2.4 Horas de práctica: cuatro (4) semanales

2.5 Duración del período: dieciséis (16) semanas

2.6 Condición:

- CS2102 Análisis y Diseño de Algoritmos

2.7 Modalidad: Virtual

2.8 Requisitos:

- CS1103 Programación Orientada a Objetos II

3. PROFESORES

3.1 Profesor coordinador del curso

Pedro Shiguijara (pshiguijara@utec.edu.pe)

Horario de atención: previa coordinación con el profesor TP

3.2. Profesor(es) instructor(es) del curso

Pedro Shiguijara (pshiguijara@utec.edu.pe)

Horario de atención: previa coordinación con el profesor TP

4. INTRODUCCIÓN AL CURSO

El fundamento teórico de todas las ramas de la informática descansa sobre los algoritmos y estructuras de datos. Este curso cubre el diseño, análisis e implementación de estructuras de datos y algoritmos para resolver problemas de ingeniería. Los temas incluyen estructuras de datos base (i.e. listas, pilas, colas y arreglos), estructuras avanzadas (i.e. hashing, árboles y grafos), y algoritmos de manipulación y aplicación de dichas estructuras. Formando así una base que servirá para los siguientes cursos en la carrera.

Fecha de actualización: 09/04/2021

Revisado y aprobado por el Centro de Excelencia en Enseñanza y Aprendizaje y la Dirección de Ciencia de la Computación

5. OBJETIVOS

Sesión 1: Explicar los conceptos básicos de los algoritmos y estructuras de datos y dar ejemplos de su aplicación en diversos problemas.

Sesión 2: Introducir el concepto de listas a partir de un problema, y mostrar ejemplos de sus operaciones.

Sesión 3: Describir los distintos tipos de listas y sus particularidades en las implementaciones. Explicar recorrido utilizando iteradores y el uso de type traits.

Sesión 4: Describir el tipo de dato abstracto árbol, y sus subtipos como árboles binarios, heaps y disjoint sets.

Sesión 5: Mostrar múltiples problemas donde se pueda aplicar los diversos métodos de los árboles.

Sesión 6: Explicar la teoría básica de grafos y describir las diversas formas de representarlos. Mostrar las múltiples aplicaciones de búsqueda en grafos.

Sesión 7: Describir el uso e implementaciones de matrices esparza y su relación con grafos.

Sesión 8: Identificar los diversos usos del hashing y explicar su relación con las tablas hash. Mostrar ejemplos de su aplicación en el lenguaje C++ (i.e. unordered maps).

Sesión 9: Mostrar los problemas del árbol binario de búsqueda para introducir los árboles autobalanceados (e.g. B, AVL). Analizar las diferencias entre los árboles autobalanceados, y sus diferentes usos (e.g. base de datos, diccionarios).

6. COMPETENCIAS Y CRITERIOS DE DESEMPEÑO

Los criterios de desempeño que se van a trabajar en este curso son:

1.3. Aplica conocimientos de computación apropiados para la solución de problemas definidos y sus requerimientos en la disciplina del programa (*nivel 3*).

2.1: Identifica problemas complejos de computación y otras disciplinas relevantes en el dominio (*nivel 2*).

3.1. Diseña y evalúa sistemas, componentes o procesos que satisfacen las necesidades específicas (*nivel 2*).

5.1. Trabaja eficazmente en equipo (*nivel 2*).

7. RESULTADOS DE APRENDIZAJE

Al finalizar el curso de Algoritmos y estructuras de datos se espera que el estudiante sea capaz de:

RA1 Evaluar técnicas existentes de estructuras de datos para resolver problemas específicos.

RA2. Analizar problemas e identificar las estructuras de datos y algoritmos apropiados para su solución.

RA3. Implementar estructuras de datos como listas, colas, pilas, heaps, hashing, árboles, conjuntos y grafos.

RA4. Desarrollar soluciones computacionales de modo colaborativo usando herramientas adecuadas.

8. TEMAS

1. Listas, Colas y Pilas

- 1.1. Listas
 - 1.1.1. Lista simplemente enlazada
 - 1.1.2. Lista doblemente enlazada
 - 1.1.3. Lista circular
 - 1.1.4. Type traits
 - 1.1.5. Vector (ArrayList)
 - 1.1.6. Iteradores
- 1.2. Colas
- 1.3. Pilas

2. Árboles

- 2.1. Árboles binarios
 - 2.1.1. Sets y maps
 - 2.1.2. Iteradores
- 2.2. Heaps binarios
 - 2.2.1. Min heap
 - 2.2.2. Max heap
- 2.3. Disjoint sets

3. Grafos

- 3.1. Definiciones
- 3.2. Implementación de Grafos usando Matrices de Adyacencia
- 3.3. Implementación de Grafos usando Listas de Adyacencia

- 3.4. Inserción, Búsqueda y Eliminación de nodos y aristas
- 3.5. Árbol de mínima expansión
- 3.6. Algoritmos de búsqueda en grafos

4. Matrices Esparzas

- 4.1. Definiciones
- 4.2. Creación de la matriz esparza estática y Dinámicas
- 4.3. Métodos de inserción, búsqueda y eliminación

5. Tablas Hash

- 5.1. Definiciones
- 5.2. Función Hash
- 5.3. Manejo de colisiones

6. Árboles Equilibrados

- 6.1. Definiciones
- 6.2. Árboles AVL
- 6.3. Árboles B
- 6.4. Tries

9. PLAN DE TRABAJO

9.1 Metodología

Este curso presenta por metodología activa el aprendizaje clásico y el aprendizaje basado en problemas; ambos son fundamentales para introducir al estudiante a los conceptos básicos y afianzar la base necesaria para los siguientes cursos de carrera. Ambos aumentan el interés del estudiante y promueven su compromiso en el aprendizaje.

9.2 Sesiones de teoría

Las sesiones de teoría se llevan a cabo en clases magistrales donde se realizarán actividades que propicien un aprendizaje activo, con dinámicas que permitan a los estudiantes interiorizar los conceptos. 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.

9.3 Sesiones de práctica (laboratorio o taller)

Para verificar que los alumnos hayan alcanzado el logro planteado para cada una de las unidades de aprendizaje, se aplica una metodología basada en retos. Los alumnos realizarán actividades que les permitan aplicar los conocimientos adquiridos durante las sesiones de teoría y se les propondrán retos que permitan evaluar su desempeño.

10. SISTEMA DE EVALUACIÓN

Parte de la evaluación continua serán presentaciones grupales, y prácticas individuales en clase y laboratorio. Las prácticas en clase serán presentadas el mismo día salvo algunas excepciones. Mientras que las prácticas de laboratorio tendrán una fecha de entrega.

	TEORÍA (T)	LABORATORIO (L)
EVALUACIÓN *La ponderación de la evaluación se hará si ambas partes están aprobadas	Examen E1 (25%) Examen E2 (25%)	Evaluación Continua C1 (20%) Evaluación Continua C2 (20%) Proyecto P1 (5%) Proyecto P2 (5%)
	50%	50%
	100%	

Las rúbricas que permitirán medir las actividades más significativas del curso y que, además se relacionan con la evaluación de las competencias del estudiante son: [enlace](#)

11. REFERENCIAS BIBLIOGRÁFICAS

Libro

Thomas H. Cormen et al. Introduction to Algorithms. Third Edition. ISBN: 978-0-262-53305-8. MIT Press, 2009.

Libro

R. Sedgewick, Algorithms in C (parts 1-4), 3rd. edition, Addison-Wesley/Longman, 1998.

Libro

José Fager et al. Estructura de datos. First Edition. Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIN), 2014.

Libro

Sedgewick, R., & Wayne, K. (2011). *Algorithms*. Addison-wesley professional.