

Universidad Católica San Pablo (UCSP)
Escuela Profesional de
Ciencia de la Computación
SILABO



CS210. Algoritmos y Estructuras de Datos
(Obligatorio)

2020-I

1. Información general

1.1 Escuela	:	Ciencia de la Computación
1.2 Curso	:	CS210. Algoritmos y Estructuras de Datos
1.3 Semestre	:	4 ^{to} Semestre.
1.4 Prerrequisitos	:	<ul style="list-style-type: none">• CS113. Ciencia de la Computación II. (3^{er} Sem)• CS100. Introducción de Ciencia de la Computación. (2^{do} Sem)
1.5 Condición	:	Obligatorio
1.6 Modalidad de aprendizaje	:	Virtual
1.7 horas	:	2 HT; 2 HP; 2 HL;
1.8 Créditos	:	4

2. Profesores

Titular

- Alex Jesús Cuadros Vargas <acuadros@ucsp.edu.pe>
 - ¡¡PosDocIn!! Ciencia de la Computación, ICMC-USP, Brasil, 2009.
 - Doctor en Ciencia de la Computación, ICMC-USP, Brasil, 2007.
 - Master en Ciencia de la Computación, ICMC-USP, Brasil, 2001.

Laboratorio

- Gustavo Delgado Ugarte <ggdelgado@ucsp.edu.pe>
 - Master en Ingeniería del Software, Escuela Universitaria de Ingeniería Industrial, Informática y Sistemas - UTA, Chile, 2009.

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. Resumen

1. Grafos 2. Matrices Esparzas 3. Arboles Equilibrados

5. Objetivos Generales

- 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 los resultados (*Outcomes*)

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

- 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 CS en el modelamiento y diseño de sistemas. (**Usar**)
- k) Aplicar los principios de desarrollo y diseño en la construcción de sistemas de software de complejidad variable. (**Usar**)

7. Contenido

UNIDAD 1: Grafos (12)

Competencias: a,b,c

Contenido

Objetivos Generales

- Concepto de Grafos.
- Grafos Dirigidos y Grafos no Dirigidos.
- Utilización de los Grafos.
- Medida de la Eficiencia. En tiempo y espacio.
- Matrices de Adyacencia.
- Matrices de Adyacencia etiquetada.
- Listas de Adyacencia.
- Implementación de Grafos usando Matrices de Adyacencia.
- Implementación de Grafos usando Listas de Adyacencia.
- Inserción, Búsqueda y Eliminación de nodos y aristas.
- Algoritmos de búsqueda en grafos.

- Adquirir destreza para realizar una implementación correcta. [Usar]
- Desarrollar los conocimientos para decidir cuando es mejor usar una técnica de implementación que otra. [Usar]

Lecturas: Cormen et al. (2009), Fager et al. (2014), Knuth (1997), Knuth (1998)

UNIDAD 2: Matrices Esparzas (8)	
Competencias: a,b,c	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> • Conceptos Iniciales. • Matrices poco densas • Medida de la Eficiencia en Tiempo y en Espacio • Creación de la matriz esparza estática vs Dinámicas. • Métodos de inserción, búsqueda y eliminación 	<ul style="list-style-type: none"> • Comprender el uso y implementación de matrices esparzas.[Evaluar]
Lecturas: Cormen et al. (2009), Fager et al. (2014), Knuth (1997), Knuth (1998)	

UNIDAD 3: Árboles Equilibrados (16)	
Competencias: a,b,c	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> • Árboles AVL. • Medida de la Eficiencia. • Rotaciones Simples y Compuestas • Inserción, Eliminación y Búsqueda. • Árboles B , B+ B* y Patricia. 	<ul style="list-style-type: none"> • Comprender las funciones básicas de estas estructuras complejas con el fin de adquirir la capacidad para su implementación. [Evaluar]
Lecturas: Cormen et al. (2009), Fager et al. (2014), Knuth (1997), Knuth (1998)	

8. Metodología
<p>El profesor del curso presentará clases teóricas de los temas señalados en el programa propiciando la intervención de los alumnos.</p> <p>El profesor del curso presentará demostraciones para fundamentar clases teóricas.</p> <p>El profesor y los alumnos realizarán prácticas</p> <p>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.</p>

9. Evaluar
<p>Evaluación Continua 1 : 20 %</p> <p>Examen parcial : 30 %</p> <p>Evaluación Continua 2 : 20 %</p> <p>Examen final : 30 %</p>

References

- Cormen, Thomas H. et al. (2009). *Introduction to Algorithms*. Third Edition. ISBN: 978-0-262-53305-8. MIT Press.
- Fager, José et al. (2014). *Estructura de datos*. First Edition. Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIN).
- Knuth, Donald E. (1997). *The Art of Computer Programming, Vol. 1: Fundamental Algorithms*. 3rd. Addison-Wesley Professional.

Knuth, Donald E. (1998). *The art of computer programming, volume 3:Sorting and searching*. 2nd. Addison-Wesley Professional.