

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



CS312. Estructuras de Datos Avanzadas (Obligatorio)

1. Información general

1.1 Escuela	:	Ciencia de la Computación
1.2 Curso	:	CS312. Estructuras de Datos Avanzadas
1.3 Semestre	:	6 ^{to} Semestre.
1.4 Prerrequisitos	:	CS212. Análisis y Diseño de Algoritmos. (5 ^{to} 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

- Eddie Rogger Peralta Aranibar <erperalta@ucsp.edu.pe>
– Master en Ciencia de la Computación, Universidad Católica San Pablo, Perú, 2019.

3. Fundamentación del curso

Los algoritmos y estructuras de datos son una parte fundamental de la ciencia de la computación que nos permiten organizar la información de una manera más eficiente, por lo que es importante para todo profesional del área tener una sólida formación en este aspecto.

En el curso de estructuras de datos avanzadas nuestro objetivo es que el alumno conozca y analice estructuras complejas, como los Métodos de Acceso Multidimensional, Métodos de Acceso Espacio-Temporal y Métodos de Acceso Métrico, etc.

4. Resumen

1. Técnicas Básicas de Implementación de Estructuras de Datos 2. Métodos de Acceso Multidimensionales 3. Métodos de Acceso Métrico 4. Métodos de Acceso Aproximados 5. Seminarios

5. Objetivos Generales

- Que el alumno entienda, diseñe, implemente, aplique y proponga estructuras de datos innovadoras para solucionar problemas relacionados al tratamiento de datos multidimensionales, recuperación de información por similitud, motores de búsqueda y otros problemas computacionales.

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. (**Familiarizarse**)
- 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. (**Familiarizarse**)
- j) Aplicar la base matemática, principios de algoritmos y la teoría de la CS en el modelamiento y diseño de sistemas. (**Familiarizarse**)

7. Contenido

UNIDAD 1: Técnicas Básicas de Implementación de Estructuras de Datos (16)

Competencias: a,b,c

Contenido	Objetivos Generales
<ul style="list-style-type: none">• Programación estructurada• Programación Orientada a Objetos• Tipos Abstractos de Datos• Independencia del lenguaje de programación del usuario de la estructura• Independencia de Plataforma• Control de concurrencia• Protección de Datos• Niveles de encapsulamiento (struct, class, namespace, etc)	<ul style="list-style-type: none">• Que el alumno entienda las diferencias básicas que involucran las distintas técnicas de implementación de estructuras de datos[Usar]• Que el alumno analice las ventajas y desventajas de cada una de las técnicas existentes[Usar]

Lecturas: Cuadros-Vargas et al. (2004), Knuth (2007a), Knuth (2007b), Gamma et al. (1994), Björnander (2018), David Vandevorde (2018)

UNIDAD 2: Métodos de Acceso Multidimensionales (16)

Competencias: a,b,c

Contenido	Objetivos Generales
<ul style="list-style-type: none">• Métodos de Acceso para datos puntuales• Métodos de Acceso para datos no puntuales• Problemas relacionados con el aumento de dimensión	<ul style="list-style-type: none">• Que el alumno entienda conozca e implemente algunos Métodos de Acceso para datos multidimensionales y espacio temporales[Usar]• Que el alumno entienda el potencial de estos Métodos de Acceso en el futuro de las bases de datos comerciales[Usar]

Lecturas: Samet (2006), Gaede and ünther (1998)

UNIDAD 3: Métodos de Acceso Métrico (20)

Competencias: a,b,c

Contenido	Objetivos Generales
<ul style="list-style-type: none">• Métodos de Acceso Métrico para distancias discretas• Métodos de Acceso Métrico para distancias continuas	<ul style="list-style-type: none">• Que el alumno entienda conozca e implemente algunos métodos de acceso métrico[Usar]• Que el alumno entienda la importancia de estos Métodos de Acceso para la Recuperación de Información por Similitud[Usar]

Lecturas: Samet (2006), Chávez et al. (2001), Traina Jr et al. (2000), Zezula et al. (2007)

UNIDAD 4: Métodos de Acceso Aproximados (20)	
Competencias: a,b,c	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> • Space Filling Curves • Locality Sensitive Hashing 	<ul style="list-style-type: none"> • Que el alumno entienda conozca e implemente algunos métodos de acceso aproximados[Usar] • Que el alumno entienda la importancia de estos Métodos de Acceso para la Recuperación de Información por Similitud en entornos donde la Escalabilidad sea una factor muy importante[Usar]
Lecturas: Samet (2006), PGregory Shakhnarovich and Indyk (2006), Zezula et al. (2007)	

UNIDAD 5: Seminarios (8)	
Competencias: a,b,c	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> • Métodos de Acceso Espacio Temporal • Estructuras de Datos con programación genérica 	<ul style="list-style-type: none"> • Que el alumno pueda discutir sobre los últimos avances en métodos de acceso para distintos dominios de conocimiento[Usar]
Lecturas: Samet (2006), Chávez et al. (2001)	

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

- Björnander, Stefan (Feb. 2018). *C++17 By Example: Practical projects to get you up and running with C++17*. Packt Publishing.
- Chávez, E. et al. (Sept. 2001). “Proximity Searching in Metric Spaces”. In: *ACM Computing Surveys* 33(3), pp. 273–321.
- Cuadros-Vargas, Ernesto et al. (2004). “Implementing data structures: An incremental approach”. <http://socios.spc.org.pe/ecuadros/cursos/pdfs/>.
- David Vandevoorde Nicolai M. Josuttis, Doug Gregor (Sept. 2018). *C++ Templates: The Complete Guide*. Addison-Wesley Professional.
- Gaede, Volker and Oliver ünther (1998). “Multidimensional Access Methods”. In: *ACM Computing Surveys* 30(2), pp. 170–231.
- Gamma, Erich et al. (Nov. 1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Computing Series. ISBN-10: 0201633612. Addison-Wesley Professional.

- Knuth, Donald Ervin (Feb. 2007a). *The Art of Computer Programming, Fundamental Algorithms*. 3rd. Vol. I. 0-201-89683-4. Addison-Wesley.
- Knuth, Donald Ervin (Feb. 2007b). *The Art of Computer Programming, Sorting and Searching*. 2nd. Vol. II. 0-201-89685-0. Addison-Wesley.
- PGregory Shakhnarovich, Trevor Darrell and Piotr Indyk (Mar. 2006). *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*. 1st. ISBN 0-262-19547-X. MIT Press.
- Samet, Hanan (Aug. 2006). *Foundations of Multidimensional and Metric Data Structures*. Illustrated. Elsevier/Morgan Kaufmann. ISBN: 9780123694461.
- Traina Jr, C. et al. (Mar. 2000). "Slim-Trees: High Performance Metric Trees Minimizing Overlap between Nodes". In: *Advances in Database Technology - EDBT 2000, 6th International Conference on Extending Database Technology*. Vol. 1777. Lecture Notes in Computer Science. Springer: Konstanz, Germany, pp. 51–65.
- Zeuzula, Pavel et al. (Nov. 2007). *Similarity Search: The Metric Space Approach*. 1st. ISBN-10: 0387291466. Springer.