

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 | : | Presencial |
| 1.7 horas | : | 2 HT; 4 HP; |
| 1.8 Créditos | : | 4 |
| 1.9 Plan | : | Plan Curricular 2016 |

2. Profesores

Titular

- Erick Gomez Nieto <emgomez@ucsp.edu.pe>
 - Doctor en Ciencia de la Computación y Matemática Computacional, Universidad de Sao Paulo - USP, Brasil, 2017.
 - Master en Ciencia de la Computación, Universidad de Sao Paulo - USP, Brasil, 2012.

Laboratorio

- Carlos Eduardo Atencio Torres <ceatencio@ucsp.edu.pe>
 - Master en Ciencia de la Computación, USP, Brasil, 2014.

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. Multidimensional Data 2. Multidimensional Acces Data Structures 3. Approximate Access Methods 4. Métodos de Acceso Aproximados 5. Clustering 6. Temporal Data Structures 7. Final Talks

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:

- 1) S.O. Analizar un problema computacional complejo y aplicar los principios computacionales y otras disciplinas relevantes para identificar soluciones. (**Usar**)
- 2) S.O. Diseñar, implementar y evaluar una solución basada en computación para cumplir con un conjunto determinado de requisitos computacionales en el contexto de las disciplinas del programa. (**Familiarizarse**)
- 6) S.O. Aplicar la teoría de la computación y los fundamentos del desarrollo de software para producir soluciones basadas en computación. . (**Familiarizarse**)

7. Contenido**UNIDAD 1: Multidimensional Data (16)****Competencias: 1,2****Contenido**

- Introducción al curso.
- Introducción a datos multidimensionales.
- Maldición de la dimensionalidad.

Objetivos Generales

- Introducir la trascendencia de la representación multidimensional de datos. [Usar]
- Entender la complejidad de lidiar con datos multidimensional y de alta dimensión.[Usar]
- Entender la maldición de la dimensionalidad, y su impacto en el indizado de grandes volúmenes de datos.[Usar]
- Presentar y discutir aplicaciones reales de datos multidimensionales en motores de búsqueda.[Usar]

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

UNIDAD 2: Multidimensional Acces Data Structures (16)**Competencias: 1,2****Contenido**

- Introducción a estructuras de datos espaciales.
- Estructuras espaciales, Quadtree, Octree y visualización.
- Kd-Tree.
- Introducción a R-Tress.
- R tree (Guttman).
- R+ tree.
- R* tree.
- Variación R*-tree y relación con paginación y tamaño de bloques.
- X-tree.

Objetivos Generales

- Introducir los fundamentos teóricos de estructuras de datos espaciales.
- Entender los beneficios y limitaciones de estructuras de datos espaciales basadas en árbol.
- Implementar diferentes estructuras de datos para el indizado de grandes volúmenes de datos.
- Entender los fundamentos e implementar estrategias de búsqueda como vecinos mas próximos y búsquedas por rango.

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

| UNIDAD 3: Approximate Access Methods (20) | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Competencias: 1,2 | |
| 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: Cuadros-Vargas et al. (2004), Knuth (2007a), Knuth (2007b), Gamma et al. (1994) | |

| UNIDAD 4: Métodos de Acceso Aproximados (20) | |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Competencias: 1,2 | |
| Contenido | Objetivos Generales |
| <ul style="list-style-type: none"> • Space Filling Curves: Hilbert curve y Z-order • Proyecciones y complejidad. • Locally sensitive hashing (LSH) | <ul style="list-style-type: none"> • Entender, conocer e implementar algunos métodos de acceso aproximados. • Entender la importancia de estos métodos de Acceso para la recuperación de información por similitud en entornos donde la escalabilidad sea una factor muy importante. |
| Lecturas: Samet (2006), PGregory Shakhnarovich and Indyk (2006), Zezula et al. (2007) | |

| UNIDAD 5: Clustering (8) | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Competencias: 1,2 | |
| Contenido | Objetivos Generales |
| <ul style="list-style-type: none"> • Introducción a Clustering. • Kmeans y DBScan. • Clustering Applications. • Clustering Ensemble. | <ul style="list-style-type: none"> • Introducir los fundamentos teóricos para el clustering de datos multidimensionales. • Implementar diferentes estrategias para el clustering de datos multidimensionales, como basados en partición, en jerarquía o en densidad. • Entender los fundamentos, aplicaciones e implementar ensambles de métodos de clustering. • Implementar ensambles de métodos de clustering con datos reales. |
| Lecturas: Cuadros-Vargas et al. (2004), Knuth (2007a), Knuth (2007b), Gamma et al. (1994) | |

| UNIDAD 6: Temporal Data Structures (8) | |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Competencias: 1,2 | |
| Contenido | Objetivos Generales |
| <ul style="list-style-type: none"> • Introducción a Estructuras de datos temporales. • Versionando la estructura de Datos. • Persistencia • Retroactividad | <ul style="list-style-type: none"> • Introducir los fundamentos teóricos de estructuras de datos temporales. • Entender, discutir e implementar Persistencia y sus tipos. • Entender, discutir e implementar Retroactividad y sus tipos. • Entender y discutir los beneficios y limitaciones entre persistencia y retroactividad. |
| Lecturas: Cuadros-Vargas et al. (2004), Knuth (2007a), Knuth (2007b), Gamma et al. (1994) | |

| UNIDAD 7: Final Talks (8) | |
|----------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Competencias: 1,2 | |
| Contenido | Objetivos Generales |
| <ul style="list-style-type: none"> • Seminarios de trabajo de investigación. | <ul style="list-style-type: none"> • Investigar sobre nuevos métodos para el indizado de grandes volúmenes de datos complejos. • Presentar y dirigir la discusión sobre métodos para indizados de Big Data investigado. |
| Lecturas: SCuadros2004Implementing, Knuth (2007a), Knuth (2007b), Gamma et al. (1994) | |

8. Metodología

1. El profesor del curso presentará clases teóricas de los temas señalados en el programa propiciando la intervención de los alumnos.
2. El profesor del curso presentará demostraciones para fundamentar clases teóricas.
3. El profesor y los alumnos realizarán prácticas
4. 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. Evaluar Sesiones Teóricas:

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.

Sesiones Prácticas:

Las sesiones prácticas se llevan en clase donde se desarrollan una serie de ejercicios y/o conceptos prácticos mediante planteamiento de problemas, la resolución de problemas, ejercicios puntuales y/o en contextos aplicativos.

Sistema de Evaluación:

La nota final se obtiene a través de:

| EVALUACIONES PERMANENTES | EVALUACIONES |
|--------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Evaluación Permanente 1 : 20 % Evaluación Permanente 2 : 20 % | Evaluación Parcial : 20 % Parcial Teórico : 70 % Parcial Práctico : 30 % Evaluación Final : 40 % Trabajo Final : 30 % Examen Final : 70 % |
| 40% | 60% |

Donde:

Evaluación Permanente: Comprende trabajos grupales, participación activa en clase, test de ejercicios.

- Permanente 1 (Semanas 1 - 9)
- Permanente 2 (Semanas 10 - 17)

Para aprobar el curso, el alumno debe obtener 11.5 o más en la nota final.

References

- Cuadros-Vargas, Ernesto et al. (2004). "Implementing data structures: An incremental approach". <http://socios.spc.org.pe/ecuadros/cursos/pdfs/>.
- 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.
- Zezula, Pavel et al. (Nov. 2007). *Similarity Search: The Metric Space Approach*. 1st. ISBN-10: 0387291466. Springer.