San Pablo Catholic University (UCSP)
Undergraduate Program in
Computer Science
SILABO

Universidad Católica San Pablo
2021-I

CS312. Advanced Data Structures (Mandatory)

1. General information

1.1 School	:	Ciencia de la Computación
1.2 Course	:	CS312. Advanced Data Structures
1.3 Semester	:	6^{to} Semestre.
1.4 Prerrequisites	:	CS212. Algorithm Analysis and Design. (5^{th} Sem)
1.5 Type of course	:	Mandatory
1.6 Learning modality	:	Virtual
1.7 Horas	:	2 HT; 2 HP; 2 HL;
1.8 Credits	:	4

2. Professors

Lecturer

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

Practice

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

3. Course foundation

Algorithms and data structures are a fundamental part of computer science that allow us to organize information more efficiently, so it is important for every professional in the area to have a solid background in this regard.
In the course of advanced data structures our goal is for the student to know and analyze complex structures, such as Multidimensional Access Methods, Spatio-Temporal Access Methods and Metric Access Methods, Compact Data Structures, etc.

4. Summary

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. Generales Goals

- That the student understands, designs, implements, applies and Propose innovative data structures to solve problems related to the handling of multidimensional data, retrieval of information by similarity, search engines and other computational problems.

6. Contribution to Outcomes

This discipline contributes to the achievement of the following outcomes:

a) An ability to apply knowledge of mathematics, science. (**Familiarity**)

b) An ability to design and conduct experiments, as well as to analyze and interpret data. (**Usage**)

c) An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability. (**Familiarity**)

j) Apply the mathematical basis, principles of algorithms and the theory of Computer Science in the modeling and design of computational systems in such a way as to demonstrate understanding of the equilibrium points involved in the chosen option. (**Familiarity**)

7. Content

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

Competences: a,b,c

Content	**Generales Goals**
• Structured Programming • Object-oriented programming • Abstract Data Types • Independence of the user programming language of the structure • Platform Independence • Concurrency control • Data Protection • Encapsulation levels (struct, class, namespace, etc)	• That the student understands the basic differences that involve the different techniques of implementation of data structures[Usage] • That the student analyze the advantages and disadvantages of each of the existing techniques[Usage]

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

UNIT 2: Métodos de Acceso Multidimensionales (16)

Competences: a,b,c

Content	**Generales Goals**
• Access Methods for Point Data • Access Methods for non-point data • Problems with dimension enhancement	• That the student understands to know and implement some Access Methods for multidimensional data and temporal space[Usage] • That the student understands the potential of these Access Methods in the future of commercial databases[Usage]

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

UNIT 3: Métodos de Acceso Métrico (20)	
Competences: a,b,c	
Content	**Generales Goals**
• Metric Access Methods for discrete distances • Metric Access Methods for Continuous Distances	• That the student understands to know and implement some methods of metric access[Usage] • That the student understands the importance of these Access Methods for Information Retrieval by similarity[Usage]
Readings: Samet (2006), Chávez et al. (2001), Traina Jr et al. (2000), Zezula et al. (2007)	

UNIT 4: Métodos de Acceso Aproximados (20)	
Competences: a,b,c	
Content	**Generales Goals**
• Space Filling Curves • Locality Sensitive Hashing	• That the student understands to know and implement some approximate access methods[Usage] • That the student understands the importance of these Access Methods for Information Retrieval by Similarity in environments where Scalability is a very important factor [Usage]
Readings: Samet (2006), PGregory Shakhnarovich and Indyk (2006), Zezula et al. (2007)	

UNIT 5: Seminarios (8)	
Competences: a,b,c	
Content	**Generales Goals**
• Access Methods Temporary Space • Generic Data Structures	• That the student can discuss the latest advances in access methods for different domains of knowledge [Usage]
Readings: Samet (2006), Navarro (2016), Chávez et al. (2001)	

8. Methodology

El profesor del curso presentará clases teóricas de los temas señalados en el programa propiciando la intervención de los alumnos.

El profesor del curso presentará demostraciones para fundamentar clases teóricas.

El profesor y los alumnos realizarán prácticas

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. Assessment

Continuous Assessment 1 : 20 %

Partial Exam : 30 %

Continuous Assessment 2 : 20 %

Final exam : 30 %

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.

Navarro, Gonzalo (2016). *Compact Data Structures*. Cambridge University Press. ISBN: 978-1107152380.

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.

Zezula, Pavel et al. (Nov. 2007). *Similarity Search: The Metric Space Approach*. 1st. ISBN-10: 0387291466. Springer.