Peruvian Computing Society (SPC)
School of Computer Science
Sillabus 2021-I

1. COURSE

CS312. Advanced Data Structures (Mandatory)

2. GENERAL INFORMATION

2.1 Credits	:	4
2.2 Theory Hours	:	2 (Weekly)
2.3 Practice Hours	:	2 (Weekly)
2.4 Duration of the period	:	16 weeks
2.5 Type of course	:	Mandatory
2.6 Modality	:	Face to face
2.7 Prerrequisites	:	CS212. Analysis and Design of Algorithms. (5^{th} Sem)

3. PROFESSORS

Meetings after coordination with the professor

4. INTRODUCTION TO THE COURSE

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.

5. 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. COMPETENCES

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**)

7. SPECIFIC COMPETENCES

a33) Analyze and apply the computational cost in a metric space.

a34) Analyze and apply multimensional access methods in georeferenced query problems.

a35) Analyze and apply multi-mensional access methods with temporal variation.

a36) Analyze the problem of high dimensions in the efficiency of a query.

b4) Identify and efficiently apply various algorithmic strategies and data structures for the solution of a problem given certain space and time constraints.

c16) Implement a spatial or metric data structure in an open database engine.

8. TOPICS

Unit 1: Basic techniques to implement data structuras (16)	
Competences Expected: a,b,c	
Topics	**Learning Outcomes**
• 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 : [Cua+04], [Knu07a], [Knu07b], [Gam+94], [Bjö18], [Dav18]	

Unit 2: Multidimensional access methods (16)	
Competences Expected: a,b,c	
Topics	**Learning Outcomes**
• 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 : [Sam06], [Gü98]	

Unit 3: Metric access methods (20)	
Competences Expected: a,b,c	
Topics	**Learning Outcomes**
• 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 : [Sam06], [Chá+01], [Tra+00], [Zez+07]	

Unit 4: Approximate access methods (20)	
Competences Expected: a,b,c	
Topics	**Learning Outcomes**
• 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 : [Sam06], [PI06], [Zez+07]	

Unit 5: Seminars (8)	
Competences Expected: a,b,c	
Topics	**Learning Outcomes**
• 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 : [Sam06], [Nav16], [Chá+01]	

9. WORKPLAN

9.1 Methodology

Individual and team participation is encouraged to present their ideas, motivating them with additional points in the different stages of the course evaluation.

9.2 Theory Sessions

The theory sessions are held in master classes with activities including active learning and roleplay to allow students to internalize the concepts.

9.3 Practical Sessions

The practical sessions are held in class where a series of exercises and/or practical concepts are developed through problem solving, problem solving, specific exercises and/or in application contexts.

10. EVALUATION SYSTEM

********* EVALUATION MISSING ********

11. BASIC BIBLIOGRAPHY

[Bjö18] Stefan Björnander. *C++17 By Example: Practical projects to get you up and running with C++17*. Packt Publishing, Feb. 2018.

[Chá+01] E. Chávez et al. "Proximity Searching in Metric Spaces". In: *ACM Computing Surveys* 33.3 (Sept. 2001), pp. 273–321.

[Cua+04] Ernesto Cuadros-Vargas et al. "Implementing data structures: An incremental approach". http://socios.spc.org.pe/ecuadros/cursos/pdfs/. 2004.

[Dav18] Doug Gregor David Vandevoorde Nicolai M. Josuttis. *C++ Templates: The Complete Guide*. Addison-Wesley Professional, Sept. 2018. URL: `http://informit.com/aw`.

[Gam+94] Erich Gamma et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. Computing Series. ISBN-10: 0201633612. Addison-Wesley Professional, Nov. 1994.

[Gü98] Volker Gaede and Oliver ünther. "Multidimensional Access Methods". In: *ACM Computing Surveys* 30.2 (1998), pp. 170–231.

[Knu07a] Donald Ervin Knuth. *The Art of Computer Programming, Fundamental Algorithms*. 3rd. Vol. I. 0-201-89683-4. Addison-Wesley, Feb. 2007.

[Knu07b] Donald Ervin Knuth. *The Art of Computer Programming, Sorting and Searching*. 2nd. Vol. II. 0-201-89685-0. Addison-Wesley, Feb. 2007.

[Nav16] Gonzalo Navarro. *Compact Data Structures*. Cambridge University Press, 2016. ISBN: 978-1107152380.

[PI06] Trevor Darrell PGregory Shakhnarovich and Piotr Indyk. *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice*. 1st. ISBN 0-262-19547-X. MIT Press, Mar. 2006.

[Sam06] Hanan Samet. *Foundations of Multidimensional and Metric Data Structures*. Illustrated. Elsevier/Morgan Kaufmann, Aug. 2006. ISBN: 9780123694461. URL: `http://books.google.com.pe/books?id=vO-NRRKHG84C`.

[Tra+00] C. Traina Jr et al. "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. Konstanz, Germany: Springer, Mar. 2000, pp. 51–65.

[Zez+07] Pavel Zezula et al. *Similarity Search: The Metric Space Approach*. 1st. ISBN-10: 0387291466. Springer, Nov. 2007.