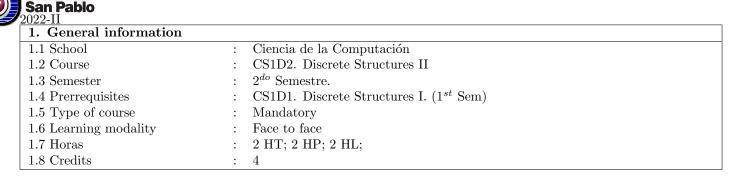
San Pablo Catholic University (UCSP) Undergraduate Program in Computer Science SILABO

CS1D2. Discrete Structures II (Mandatory)



2. Professors

Universidad Católica

3. Course foundation

In order to understand the advanced computational techniques, the students must have a strong knowledge of the Various discrete structures, structures that will be implemented and used in the laboratory in the programming language...

4. Summary

1. Digital Logic and Data Representation 2. Basics of Counting 3. Graphs and Trees

5. Generales Goals

- That the student is able to model computer science problems using graphs and trees related to data structures.
- That the student applies efficient travel strategies to be able to search data in an optimal way.
- That the student uses the various counting techniques to solve 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)

- i) An ability to use the techniques, skills, and modern computing tools necessary for computing practice. (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

 Set cardinality and counting Sum and product rule Inclusion-exclusion principle Arithmetic and geometric progressions The pigeonhole principle Permutations and combinations Mag 	bly counting arguments, including sum and proor rules, inclusion-exclusion principle and arith ic/geometric progressions [Familiarity] bly the pigeonhole principle in the context of nal proof [Familiarity] npute permutations and combinations of a se interpret the meaning in the context of the pa
 Set cardinality and counting Sum and product rule Inclusion-exclusion principle Arithmetic and geometric progressions The pigeonhole principle Permutations and combinations Mag 	rules, inclusion-exclusion principle and arith ic/geometric progressions [Familiarity] ly the pigeonhole principle in the context of nal proof [Familiarity] npute permutations and combinations of a se interpret the meaning in the context of the pa
 Pascal's identity The binomial theorem Solving recurrence relations An example of a simple recurrence relation, such as Fibonacci numbers Other examples, showing a variety of solutions Basic modular arithmetic 	lar application [Familiarity] o real-world applications to appropriate countin- nalisms, such as determining the number of way arrange people around a table, subject to con- ints on the seating arrangement, or the number vays to determine certain hands in cards (eg, house) [Familiarity] re a variety of basic recurrence relations [Family] lyze a problem to determine underlying recu- re relations [Familiarity] form computations involving modular arithmet

UNIT 3: Graphs and Trees (40) Competences: a	
 Trees Properties Traversal strategies Undirected graphs Directed graphs Weighted graphs Spanning trees/forests Graph isomorphism 	 Illustrate by example the basic terminology of graph theory, and some of the properties and special cases of each type of graph/tree [Familiarity] Demonstrate different traversal methods for trees and graphs, including pre, post, and in-order traversal of trees [Familiarity] Model a variety of real-world problems in computer science using appropriate forms of graphs and trees, such as representing a network topology or the organization of a hierarchical file system [Familiarity] Show how concepts from graphs and trees appear in data structures, algorithms, proof techniques (structural induction), and counting [Familiarity] Explain how to construct a spanning tree of a graph [Familiarity] Determine if two graphs are isomorphic [Familiarity]
Readings: Johnsonbaugh (1999)	

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

Grimaldi, R. (1997). Matemáticas Discretas y Combinatoria. Addison Wesley Iberoamericana.
Grimaldi, R. (2003). Discrete and Combinatorial Mathematics: An Applied Introduction. 5 ed. Pearson.
Johnsonbaugh, Richard (1999). Matemáticas Discretas. Prentice Hall, México.
Rosen, Kenneth H. (2007). Discrete Mathematics and Its Applications. 7 ed. Mc Graw Hill.