

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



**CS1D2. Estructuras Discretas II (Obligatorio)**

**1. Información general**

1.1 Escuela	:	Ciencia de la Computación
1.2 Curso	:	CS1D2. Estructuras Discretas II
1.3 Semestre	:	2 <sup>do</sup> Semestre.
1.4 Prerrequisitos	:	CS1D1. Estructuras Discretas I. (1 <sup>er</sup> 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**

- Luis Fernando Díaz Basurco <ldiaz@ucsp.edu.pe>  
– Master en Matemática, Pontificia Universidad Católica del Perú, Perú, 1990.

**3. Fundamentación del curso**

Para entender las técnicas computacionales avanzadas, los estudiantes deberán tener un fuerte conocimiento de las diversas estructuras discretas, estructuras que serán implementadas y usadas en laboratorio en el lenguaje de programación.

**4. Resumen**

1. Lógica Digital y Representación de Datos 2. Fundamentos de conteo 3. Árboles y Grafos

**5. Objetivos Generales**

- Que el alumno sea capaz de modelar problemas de ciencia de la computación usando grafos y árboles relacionados con estructuras de datos.
- Que el alumno aplique eficientemente estrategias de recorrido para poder buscar datos de una manera óptima.
- Que el alumno utilice las diversas técnicas de conteo para resolver 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**)
- i) Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (**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**

<b>UNIDAD 1: Lógica Digital y Representación de Datos (10)</b>	
<b>Competencias: a,i</b>	
<b>Contenido</b>	<b>Objetivos Generales</b>
<ul style="list-style-type: none"> <li>• Retículo: Tipos y propiedades.</li> <li>• Álgebras booleanas.</li> <li>• Funciones y expresiones booleanas.</li> <li>• Representación de las funciones booleanas: Disjuntiva normal y conjuntiva normal.</li> <li>• Puertas Lógicas.</li> <li>• Minimización del Circuito.</li> </ul>	<ul style="list-style-type: none"> <li>• Explicar la importancia del álgebra booleana como una unificación de la teoría de conjuntos y la lógica proposicional [Evaluar].</li> <li>• Explicar las estructuras algebraicas del retículo y sus tipos [Evaluar].</li> <li>• Explicar la relación entre el retículo y el conjunto de ordenadas y el uso prudente para demostrar que un conjunto es un retículo [Evaluar].</li> <li>• Explicar las propiedades que satisfacen un álgebra booleana [Evaluar].</li> <li>• Demostrar si una terna formada por un conjunto y dos operaciones internas es o no Álgebra booleana [Evaluar].</li> <li>• Encuentra las formas canónicas de una función booleana [Evaluar].</li> <li>• Representar una función booleana como un circuito booleano usando puertas lógicas[Evaluar].</li> <li>• Minimizar una función booleana [Evaluar].</li> </ul>
<b>Lecturas:</b> Rosen (2007), Grimaldi (2003)	

<b>UNIDAD 2: Fundamentos de conteo (40)</b>	
<b>Competencias: a</b>	
<b>Contenido</b>	<b>Objetivos Generales</b>
<ul style="list-style-type: none"> <li>• Técnicas de Conteo: <ul style="list-style-type: none"> <li>– Conteo y cardinalidad de un conjunto</li> <li>– Regla de la suma y producto</li> <li>– Principio de inclusión-exclusión</li> <li>– Progresión geométrica y aritmética</li> </ul> </li> <li>• Principio de las casillas.</li> <li>• Permutaciones y combinaciones: <ul style="list-style-type: none"> <li>– Definiciones básicas</li> <li>– Identidad de Pascal</li> <li>– Teorema del binomio</li> </ul> </li> <li>• Resolviendo relaciones de recurrencia: <ul style="list-style-type: none"> <li>– Un ejemplo de una relación de recurrencia simple, como los números de Fibonacci</li> <li>– Otros ejemplos, mostrando una variedad de soluciones</li> </ul> </li> <li>• Aritmetica modular basica</li> </ul>	<ul style="list-style-type: none"> <li>• Aplicar argumentos de conteo, incluyendo las reglas del producto y de la suma, principio de inclusión-exclusión y progresiones aritméticas/geométricas [Familiarizarse]</li> <li>• Aplicar el principio de las casillas en el contexto de una demostración formal [Familiarizarse]</li> <li>• Calcular permutaciones y combinaciones en un conjunto, e interpreta su significado en el contexto de una aplicación en particular [Familiarizarse]</li> <li>• Mapear aplicaciones del mundo real a formalismos de conteo adecuados, como el determinar el número de formas de acomodar a un conjunto de personas alrededor de una mesa, sujeto a restricciones en la disposición de los asientos, o en el número de maneras de determinar ciertas manos en juegos de cartas (ejm. una casa llena) [Familiarizarse]</li> <li>• Resolver una variedad de relaciones de recurrencia básicas [Familiarizarse]</li> <li>• Analizar un problema para determinar las relaciones de recurrencia implícitas [Familiarizarse]</li> <li>• Realizar cálculos que involucran aritmética modular [Familiarizarse]</li> </ul>
<b>Lecturas:</b> Grimaldi (2003)	

UNIDAD 3: Árboles y Grafos (40)	
Competencias: a	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> <li>• Árboles. <ul style="list-style-type: none"> <li>– Propiedades</li> <li>– Estrategias de recorrido</li> </ul> </li> <li>• Grafos no dirigidos</li> <li>• Grafos dirigidos</li> <li>• Grafos ponderados</li> <li>• Árboles de expansión/bosques.</li> <li>• Isomorfismo en grafos.</li> </ul>	<ul style="list-style-type: none"> <li>• Ilustrar mediante ejemplos la terminología básica de teoría de grafos, y de alguna de las propiedades y casos especiales de cada tipo de grafos/árboles [Familiarizarse]</li> <li>• Demostrar diversos métodos de recorrer árboles y grafos, incluyendo recorridos pre, post e inorden de árboles [Familiarizarse]</li> <li>• Modelar una variedad de problemas del mundo real en ciencia de la computación usando formas adecuadas de grafos y árboles, como son la representación de una topología de red o la organización jerárquica de un sistema de archivos [Familiarizarse]</li> <li>• Demostrar como los conceptos de grafos y árboles aparecen en estructuras de datos, algoritmos, técnicas de prueba (inducción estructurada), y conteos [Familiarizarse]</li> <li>• Explicar como construir un árbol de expansión de un grafo [Familiarizarse]</li> <li>• Determinar si dos grafos son isomorfos [Familiarizarse]</li> </ul>
<b>Lecturas:</b> Johnsonbaugh (1999)	

#### 8. Metodología

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. Evaluar

**Evaluación Continua 1** : 20 %

**Examen parcial** : 30 %

**Evaluación Continua 2** : 20 %

**Examen final** : 30 %

## References

- 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.