

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



**CS1D1. Estructuras Discretas I (Obligatorio)**

**1. Información general**

1.1 Escuela	:	Ciencia de la Computación
1.2 Curso	:	CS1D1. Estructuras Discretas I
1.3 Semestre	:	1 <sup>er</sup> Semestre.
1.4 Prerrequisitos	:	Ninguno
1.5 Condición	:	Obligatorio
1.6 Modalidad de aprendizaje	:	Virtual
1.7 horas	:	2 HT; 4 HP;
1.8 Créditos	:	4

**2. Profesores**

**3. Fundamentación del curso**

Las estructuras discretas proporcionan los fundamentos teóricos necesarios para la computación. Estos fundamentos no sólo son útiles para desarrollar la computación desde un punto de vista teórico como sucede En el curso de la teoría computacional, pero también es útil para la práctica de la informática; En particular en aplicaciones tales como verificación, Criptografía, métodos formales, etc.

**4. Resumen**

1. Funciones, relaciones y conjuntos 2. Lógica básica 3. Técnicas de demostración 4. Representación de Datos

**5. Objetivos Generales**

- Aplicar Correctamente conceptos de matemáticas finitas (conjuntos, relaciones, funciones) para representar datos de problemas reales.
- Modelar situaciones reales descritas en lenguaje natural, usando lógica proposicional y lógica predicada.
- Determinar las propiedades abstractas de las relaciones binarias.
- Elegir el método de demostración más apropiado para determinar la veracidad de una propuesta y construir argumentos matemáticos correctos.
- Interpretar soluciones matemáticas a un problema y determinar su fiabilidad, ventajas y desventajas.
- Expresar el funcionamiento de un circuito electrónico simple usando álgebra booleana.

**6. Contribución a los resultados (*Outcomes*)**

Esta disciplina contribuye al logro de los siguientes resultados de la carrera:

- 1) Analizar un problema computacional complejo y aplicar los principios computacionales y otras disciplinas relevantes para identificar soluciones. (**Evaluar**)
- 6) Aplicar fundamentos de teoría de ciencias de la computación y desarrollo de software para producir soluciones basados en computación. (**Evaluar**)

**7. Contenido**

<b>UNIDAD 1: Funciones, relaciones y conjuntos (22)</b>	
<b>Competencias:</b>	
<b>Contenido</b>	<b>Objetivos Generales</b>
<ul style="list-style-type: none"> <li>• Conjuntos: <ul style="list-style-type: none"> <li>– Diagramas de Venn</li> <li>– Unión, intersección, complemento</li> <li>– Producto Cartesiano</li> <li>– Potencia de conjuntos</li> <li>– Cardinalidad de Conjuntos finitos</li> </ul> </li> <li>• Relaciones: <ul style="list-style-type: none"> <li>– Reflexividad, simetria, transitividad</li> <li>– Relaciones de equivalencia</li> <li>– Relación de orden parcial y conjuntos parcialmente ordenados</li> <li>– Elementos extremos de un conjunto parcialmente ordenado</li> </ul> </li> <li>• Funciones: <ul style="list-style-type: none"> <li>– Suryecciones, inyecciones, biyecciones</li> <li>– Inversas</li> <li>– Composición</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• Explicar con ejemplos la terminología básica de funciones, relaciones y conjuntos [Evaluar]</li> <li>• Realizar las operaciones asociadas con conjuntos, funciones y relaciones [Evaluar]</li> <li>• Relacionar ejemplos prácticos para conjuntos funciones o modelos de relación apropiados e interpretar la asociación de operaciones y terminología en contexto [Evaluar]</li> </ul>
<b>Lecturas:</b> Grimaldi (2003), Rosen (2007), Velleman (2006)	

<b>UNIDAD 2: Lógica básica (14)</b>	
<b>Competencias:</b>	
<b>Contenido</b>	<b>Objetivos Generales</b>
<ul style="list-style-type: none"> <li>• Lógica proposicional.</li> <li>• Conectores lógicos.</li> <li>• Tablas de verdad.</li> <li>• Forma normal (conjuntiva y disyuntiva)</li> <li>• Validación de fórmula bien formada.</li> <li>• Reglas de inferencia proposicional (conceptos de modus ponens y modus tollens)</li> <li>• Logica de predicados: <ul style="list-style-type: none"> <li>– Cuantificación universal y existencial</li> </ul> </li> <li>• Limitaciones de la lógica proposicional y de predicados (ej. problemas de expresividad)</li> </ul>	<ul style="list-style-type: none"> <li>• Convertir declaraciones lógicas desde el lenguaje informal a expresiones de lógica proposicional y de predicados [Usar]</li> <li>• Aplicar métodos formales de simbolismo proposicional y lógica de predicados, como el cálculo de la validez de formulas y cálculo de formas normales [Usar]</li> <li>• Usar reglas de inferencia para construir demostraciones en lógica proposicional y de predicados [Usar]</li> <li>• Describir como la lógica simbólica puede ser usada para modelar situaciones o aplicaciones de la vida real, incluidos aquellos planteados en el contexto computacional como análisis de software (ejm. programas correctores ), consulta de base de datos y algoritmos [Familiarizarse]</li> <li>• Aplicar demostraciones de lógica formal y/o informal, pero rigurosa, razonamiento lógico para problemas reales, como la predicción del comportamiento de software o solución de problemas tales como rompecabezas [Usar]</li> <li>• Describir las fortalezas y limitaciones de la lógica proposicional y de predicados [Usar]</li> </ul>
<b>Lecturas:</b> Rosen (2007), Grimaldi (2003), Velleman (2006)	

UNIDAD 3: Técnicas de demostración (14)	
Competencias:	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> <li>• Nociones de implicancia, equivalencia, conversión, inversa, contrapositivo, negación, y contradicción</li> <li>• Estructura de pruebas matemáticas.</li> <li>• Demostración directa.</li> <li>• Refutar por contraejemplo.</li> <li>• Demostración por contradicción.</li> <li>• Inducción sobre números naturales.</li> <li>• Inducción estructural.</li> <li>• Inducción leve y fuerte (Ej. Primer y Segundo principio de la inducción)</li> <li>• Definiciones matemáticas recursivas.</li> <li>• Conjuntos bien ordenados.</li> </ul>	<ul style="list-style-type: none"> <li>• Identificar la técnica de demostración utilizada en una demostración dada [Evaluar]</li> <li>• Describir la estructura básica de cada técnica de demostración (demostración directa, demostración por contradicción e inducción) descritas en esta unidad [Usar]</li> <li>• Aplicar las técnicas de demostración (demostración directa, demostración por contradicción e inducción) correctamente en la construcción de un argumento sólido [Usar]</li> <li>• Determine que tipo de demostración es la mejor para un problema dado [Evaluar]</li> <li>• Explicar el paralelismo entre ideas matemáticas y/o inducción estructural para la recursión y definir estructuras recursivamente [Familiarizarse]</li> <li>• Explicar la relación entre inducción fuerte y débil y dar ejemplos del apropiado uso de cada uno [Evaluar]</li> <li>• Enunciar el principio del buen-orden y su relación con la inducción matemática [Familiarizarse]</li> </ul>
<b>Lecturas:</b> Rosen (2007), Vel06, Scheinerman (2012), Velleman (2006)	

UNIDAD 4: Representación de Datos (10)	
Competencias:	
Contenido	Objetivos Generales
<ul style="list-style-type: none"> <li>• Representaciones numéricas: signo magnitud, punto flotante.</li> <li>• Representaciones de otros objetos: conjuntos, relaciones, funciones</li> </ul>	<ul style="list-style-type: none"> <li>• Conocer las formas de representación numérica como signo magnitud y punto flotante. [Evaluar].</li> <li>• Llevar a cabo operaciones aritméticas utilizando las distintas formas de representación. [Evaluar].</li> <li>• Conocer el estándar de punto flotante IEEE-754 [Familiarizarse].</li> </ul>
<b>Lecturas:</b> Rosen (2007), Grimaldi (2003), Velleman (2006)	

8. Metodología
<p>El profesor del curso presentará clases teóricas de los temas señalados en el programa propiciando la intervención de los alumnos.</p> <p>El profesor del curso presentará demostraciones para fundamentar clases teóricas.</p> <p>El profesor y los alumnos realizarán prácticas</p> <p>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.</p>

## 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.

Rosen, Kenneth H. (2007). *Discrete Mathematics and Its Applications*. 7 ed. Mc Graw Hill.

Scheinerman, Edward R. (2012). *Mathematics: A Discrete Introduction*. 3 ed. Brooks Cole.

Velleman, Daniel J. (2006). *How to Prove It: A Structured Approach*. Ed. by Cambridge University Pres. 2nd. ISBN: 978-0521675994.