



Universidad Nacional Mayor de San Marcos
Escuela Profesional de
Ciencia de la Computación
Silabo del curso
Periodo Académico 2018-II

1. **Código del curso y nombre:** CS1D02. Estructuras Discretas II (Obligatorio)
2. **Créditos:** 4
3. **Horas de Teoría y Laboratorio:** 2 HT; 4 HP; (15 semanas)
4. **Docente(s)**

Atención previa coordinación con el profesor

5. Bibliografía

- [Gri03] R. Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. 5 ed. Pearson, 2003.
- [Gri97] R. Grimaldi. *Matemáticas Discretas y Combinatoria*. Addison Wesley Iberoamericana, 1997.
- [Joh99] Richard Johnsonbaugh. *Matemáticas Discretas*. Prentice Hall, México, 1999.
- [Ros07] Kenneth H. Rosen. *Discrete Mathematics and Its Applications*. 7 ed. Mc Graw Hill, 2007.

6. Información del curso

- (a) **Breve descripción 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.
- (b) **Prerrequisitos:** CS1D01. Estructuras Discretas I. (1^{er} Sem)
- (c) **Tipo de Curso:** Obligatorio
- (d) **Modalidad:** Presencial

7. Objetivos específicos del curso.

- 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

8. Contribución a los resultados (*Outcomes*)

- a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (**Familiarizarse**)
- b) Analizar problemas e identificar y definir los requerimientos computacionales apropiados para su solución. (**Usar**)
- i) Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (**Familiarizarse**)

9. Competencias (IEEE)

- C1.** La comprensión intelectual y la capacidad de aplicar las bases matemáticas y la teoría de la informática (*Computer Science*).⇒ **Outcome a**
- C5.** Capacidad para implementar algoritmos y estructuras de datos en el software.⇒ **Outcome b**
- CS2.** Identificar y analizar los criterios y especificaciones apropiadas a los problemas específicos, y planificar estrategias para su solución.⇒ **Outcome i**

10. Lista de temas a estudiar en el curso

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

11. Metodología y Evaluación

Metodología:

Sesiones Teóricas:

Las sesiones de teoría se llevan a cabo en clases magistrales donde se realizarán actividades que propicien un aprendizaje activo, con dinámicas que permitan a los estudiantes interiorizar los conceptos.

Sesiones de Laboratorio:

Para verificar que los alumnos hayan alcanzado el logro planteado para cada una de las unidades de aprendizaje, realizarán actividades que les permita aplicar los conocimientos adquiridos durante las sesiones de teoría y se les propondrá retos que permitan evaluar el desempeño de los alumnos.

Exposiciones individuales o grupales:

Se fomenta la participación individual y en equipo para exponer sus ideas, motivándolos con puntos adicionales en las diferentes etapas de la evaluación del curso.

Lecturas:

A lo largo del curso se proporcionan diferentes lecturas, las cuales son evaluadas. El promedio de las notas de las lecturas es considerado como la nota de una práctica calificada. El uso del campus virtual UTEC Online permite a cada estudiante acceder a la información del curso, e interactuar fuera de aula con el profesor y con los otros estudiantes.

Sistema de Evaluación:

12. Contenido

Unidad 1: Lógica Digital y Representación de Datos (10)	
Competencias esperadas: C1,C20	
Objetivos de Aprendizaje	Tópicos
<ul style="list-style-type: none"> • Explicar la importancia del álgebra booleana como una unificación de la teoría de conjuntos y la lógica proposicional [Evaluar]. • Explicar las estructuras algebraicas del retículo y sus tipos [Evaluar]. • Explicar la relación entre el retículo y el conjunto de ordenadas y el uso prudente para demostrar que un conjunto es un retículo [Evaluar]. • Explicar las propiedades que satisfacen un álgebra booleana [Evaluar]. • Demostrar si una terna formada por un conjunto y dos operaciones internas es o no Álgebra booleana [Evaluar]. • Encuentra las formas canónicas de una función booleana [Evaluar]. • Representar una función booleana como un circuito booleano usando puertas lógica[Evaluar]. • Minimizar una función booleana [Evaluar]. 	<ul style="list-style-type: none"> • Retículo: Tipos y propiedades. • Álgebras booleanas. • Funciones y expresiones booleanas. • Representación de las funciones booleanas: Disjuntiva normal y conjuntiva normal. • Puertas Lógicas. • Minimización del Circuito.
Lecturas : [Ros07], [Gri03]	

Unidad 2: Fundamentos de conteo (40)	
Competencias esperadas: C1	
Objetivos de Aprendizaje	Tópicos
<ul style="list-style-type: none"> • Aplicar argumentos de conteo, incluyendo las reglas del producto y de la suma, principio de inclusión-exclusión y progresiones aritméticas/geométricas [Familiarizarse] • Aplicar el principio de las casillas en el contexto de una demostración formal [Familiarizarse] • Calcular permutaciones y combinaciones en un conjunto, e interpreta su significado en el contexto de una aplicación en particular [Familiarizarse] • 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] • Resolver una variedad de relaciones de recurrencia básicas [Familiarizarse] • Analizar un problema para determinar las relaciones de recurrencia implícitas [Familiarizarse] • Realizar cálculos que involucran aritmética modular [Familiarizarse] 	<ul style="list-style-type: none"> • Técnicas de Conteo: <ul style="list-style-type: none"> – Conteo y cardinalidad de un conjunto – Regla de la suma y producto – Principio de inclusión-exclusión – Progresión geométrica y aritmética • Principio de las casillas. • Permutaciones y combinaciones: <ul style="list-style-type: none"> – Definiciones básicas – Identidad de Pascal – Teorema del binomio • Resolviendo relaciones de recurrencia: <ul style="list-style-type: none"> – Un ejemplo de una relación de recurrencia simple, como los números de Fibonacci – Otros ejemplos, mostrando una variedad de soluciones • Aritmetica modular basica
Lecturas : [Gri97]	

Unidad 3: Árboles y Grafos (40)	
Competencias esperadas: C1	
Objetivos de Aprendizaje	Tópicos
<ul style="list-style-type: none"> • 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] • Demostrar diversos métodos de recorrer árboles y grafos, incluyendo recorridos pre, post e inorden de árboles [Familiarizarse] • 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] • Demostrar como los conceptos de grafos y árboles aparecen en estructuras de datos, algoritmos, técnicas de prueba (inducción estructurada), y conteos [Familiarizarse] • Explicar como construir un árbol de expansión de un grafo [Familiarizarse] • Determinar si dos grafos son isomorfos [Familiarizarse] 	<ul style="list-style-type: none"> • Árboles. <ul style="list-style-type: none"> – Propiedades – Estrategias de recorrido • Grafos no dirigidos • Grafos dirigidos • Grafos ponderados • Árboles de expansión/bosques. • Isomorfismo en grafos.
Lecturas : [Joh99]	