



Universidad de Ingeniería y Tecnología
Escuela Profesional de
Ciencia de la Computación
Silabo del curso – Periodo Académico 2018-I

1. **Código del curso y nombre:** CS1D1. Estructuras Discretas I

2. **Créditos:** 4

3. **Horas de Teoría y Laboratorio:** 2 HT; 4 HP;

4. **Docente(s)**

Dr. Jose Antonio Fiestas Iquira

- Dr. Ciencias Naturales, UNIHEIDELBERG, Alemania, 2006.
- Mag. Física, UNIHEIDELBERG, Alemania, 2002.

Dr. Jose Miguel Renom Andara

- Dr. Matemáticas, USB, Venezuela, 2016.

Atención previa coordinación con el profesor

5. **Bibliografía**

[Epp10] Susanna S. Epp. *Discrete Mathematics with Applications*. 4 ed. Brooks Cole, 2010.

[Gri03] R. Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. 5 ed. Pearson, 2003.

[Ros07] Kenneth H. Rosen. *Discrete Mathematics and Its Applications*. 7 ed. Mc Graw Hill, 2007.

[Sch12] Edward R. Scheinerman. *Mathematics: A Discrete Introduction*. 3 ed. Brooks Cole, 2012.

6. **Información del curso**

- (a) **Breve descripción 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.

(b) **Prerrequisitos:**

(c) **Tipo de Curso:** Obligatorio

7. **Competencias**

- 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.
- Determine 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.

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

- a) Aplicar conocimientos de computación y de matemáticas apropiadas para la disciplina. (**Usar**)

- i) Utilizar técnicas y herramientas actuales necesarias para la práctica de la computación. (**Evaluar**)
- j) Aplicar la base matemática, principios de algoritmos y la teoría de la Ciencia de la Computación en el modelamiento y diseño de sistemas computacionales de tal manera que demuestre comprensión de los puntos de equilibrio involucrados en la opción escogida. (**Usar**)

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**
- C20.** Posibilidad de conectar la teoría y las habilidades aprendidas en la academia a los acontecimientos del mundo real que explican su pertinencia y utilidad.⇒ **Outcome i,j**

10. Lista de temas a estudiar en el curso

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

11. Metodología y Evaluación

Sesiones Teóricas:

El desarrollo de las sesiones teóricas está focalizado en el estudiante, a través de su participación activa, resolviendo problemas relacionados al curso con los aportes individuales y discutiendo casos reales de la industria. Los alumnos desarrollarán a lo largo del curso un proyecto de aplicación de las herramientas recibidas en una empresa.

Sesiones de Laboratorio:

Las sesiones prácticas se desarrollan en laboratorio. Las prácticas de laboratorio se realizan en equipos para fortalecer su comunicación. Al inicio de cada laboratorio se explica el desarrollo de la práctica y al término se destaca las principales conclusiones de la actividad en forma grupal.

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:

La nota final F depende de varias notas intermedias.

- La nota T es el promedio, redondeado hacia arriba, de los exámenes cortos sobre nueve puntos. Esta nota es individual.
- La nota P es el promedio, redondeada hacia arriba, de los cuadernos de trabajo sobre nueve puntos. Esta nota es grupal.
- La nota E es la nota de los problemas de esfuerzo, que es un entero entre cero y dos. Esta nota es individual.

Para calcular la nota final F se debe ver el desempeño del estudiante en tres bandas de desempeño, desempeño alto, desempeño medio y desempeño bajo.

Desempeño alto: Si $\min(T, P) \geq 7$ entonces $F = T + P + E$.

Desempeño medio: Si $\min(T, P) < 7$ y $\min(T, P) \geq 4$ entonces $F = T + P$.

Desempeño bajo: Si $\min(T, P) < 4$ entonces $F = 2 * \min(T, P)$.

Para aprobar el curso hay que obtener 11 o más en la nota final F .

12. Contenido

| Unidad 1: Funciones, relaciones y conjuntos (13) | |
|--|---|
| Competences esperadas: C1,C20 | |
| Objetivos de Aprendizaje | Tópicos |
| <ul style="list-style-type: none"> • Explicar con ejemplos la terminología básica de funciones, relaciones y conjuntos [Evaluar] • Realizar las operaciones asociadas con conjuntos, funciones y relaciones [Evaluar] • Relacionar ejemplos prácticos para conjuntos funciones o modelos de relación apropiados e interpretar la asociación de operaciones y terminología en contexto [Evaluar] | <ul style="list-style-type: none"> • Conjuntos: <ul style="list-style-type: none"> – Diagramas de Venn – Unión, intersección, complemento – Producto Cartesiano – Potencia de conjuntos – Cardinalidad de Conjuntos finitos • Relaciones: <ul style="list-style-type: none"> – Reflexividad, simetria, transitividad – Relaciones equivalentes, ordenes parciales • Funciones: <ul style="list-style-type: none"> – Suryecciones, inyecciones, biyecciones – Inversas – Composición |
| Lecturas : [Gri03], [Ros07] | |

| Unidad 2: Lógica básica (14) | |
|---|--|
| Competences expected: C1,C20 | |
| Objetivos de Aprendizaje | Tópicos |
| <ul style="list-style-type: none"> • Convertir declaraciones lógicas desde el lenguaje informal a expresiones de lógica proposicional y de predicados [Usar] • Aplicar métodos formales de simbolismo proposicional y lógica de predicados, como el cálculo de la validez de formulas y cálculo de formas normales [Usar] • Usar reglas de inferencia para construir demostraciones en lógica proposicional y de predicados [Usar] • 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] • 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] • Describir las fortalezas y limitaciones de la lógica proposicional y de predicados [Usar] | <ul style="list-style-type: none"> • Lógica proposicional. • Conectores lógicos. • Tablas de verdad. • Forma normal (conjuntiva y disyuntiva) • Validación de fórmula bien formada. • Reglas de inferencia proposicional (conceptos de modus ponens y modus tollens) • Logica de predicados: <ul style="list-style-type: none"> – Cuantificación universal y existencial • Limitaciones de la lógica proposicional y de predicados (ej. problemas de expresividad) |
| Lecturas : [Ros07], [Gri03] | |

| Unidad 3: Técnicas de demostración (14) | |
|--|--|
| Competences esperadas: C1,C20 | |
| Objetivos de Aprendizaje | Tópicos |
| <ul style="list-style-type: none"> • Identificar la técnica de demostración utilizada en una demostración dada [Evaluar] • Describir la estructura básica de cada técnica de demostración (demostración directa, demostración por contradicción e inducción) descritas en esta unidad [Usar] • Aplicar las técnicas de demostración (demostración directa, demostración por contradicción e inducción) correctamente en la construcción de un argumento solido [Usar] • Determine que tipo de demostración es la mejor para un problema dado [Evaluar] • Explicar el paralelismo entre ideas matemáticas y/o inducción estructural para la recursión y definir estructuras recursivamente [Familiarizarse] • Explicar la relación entre inducción fuerte y débil y dar ejemplos del apropiado uso de cada uno [Evaluar] • Enunciar el principio del buen-orden y su relación con la inducción matemática [Familiarizarse] | <ul style="list-style-type: none"> • Nociones de implicancia, equivalencia, conversión, inversa, contrapositivo, negación, y contradicción • Estructura de pruebas matemáticas. • Demostración directa. • Refutar por contraejemplo. • Demostración por contradicción. • Inducción sobre números naturales. • Inducción estructural. • Inducción leve y fuerte (Ej. Primer y Segundo principio de la inducción) • Definiciones matemáticas recursivas. • Conjuntos bien ordenados. |
| Lecturas : [Ros07], [Epp10], [Sch12] | |

| Unidad 4: Lógica Digital y Representación de Datos (19) | |
|---|--|
| Competences 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]. • Conocer 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]. • Conocer 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"> • Órdenes parciales y Conjuntos parcialmente ordenados. • Elementos extremos de un conjunto parcialmente ordenado. • Reticulo: Tipos y propiedades. • Álgebras booleanas. • Funciones y expresiones booleanas. • Representación de las funciones booleanas: Disjuntiva normal y forma conjunta. • Puertas Lógicas. • Minimización del Circuito. |
| Lecturas : [Ros07], [Gri03] | |