



### 1. Datos Generales de la asignatura

<b>Nombre de la asignatura:</b>	Matemáticas discretas
<b>Clave de la asignatura:</b>	AEF-24119
<b>SATCA<sup>1</sup>:</b>	3-2-5
<b>Carrera:</b>	Ingeniería en Desarrollo de Aplicaciones, Ingeniería en Inteligencia Artificial

### 2. Presentación

#### Caracterización de la asignatura

"Matemáticas discretas" desempeña un papel crucial en el currículo de primer semestre para los estudiantes de las ingenierías relacionadas con las ciencias de la computación, al establecer fundamentos matemáticos claves para la comprensión de los conceptos y elaboración de algoritmos específicos de cada área. Esta asignatura enriquece a los alumnos con habilidades para el análisis, modelado y resolución de problemas computacionales, a través del uso de teorías matemáticas discretas, que son indispensables en la práctica profesional. La asignatura aporta a los estudiantes los conocimientos necesarios no sólo para el diseño eficiente de algoritmos, sino también para la innovación en las estructuras de datos, abarcando desde la lógica proposicional y estructuras algebraicas hasta la teoría de grafos. Con un enfoque eminentemente práctico, asegura que los estudiantes adquieran una comprensión aplicada de las matemáticas discretas en las ciencias de la computación.

Adicionalmente, esta asignatura establece conexiones temáticas con futuros cursos esenciales, preparando a los estudiantes en aspectos fundamentales, entre otros: Programación, donde los fundamentos de lógica y estructuras algebraicas son críticos para entender la programación y el desarrollo de software. Algoritmos, aplicando la teoría de grafos y optimización para el diseño y análisis de algoritmos. Aprendizaje automático, preparando a los estudiantes con los principios para el análisis de grandes volúmenes de datos.

#### Intención didáctica

En el marco del modelo curricular del TecNM, se busca abordar los temas fundamentales de manera que se fomente un aprendizaje activo y profundo por parte de los estudiantes. El objetivo es combinar la teoría y la práctica en el tratamiento de los temas, adaptándose a la complejidad de cada concepto para asegurar una comprensión completa y aplicable. Las actividades educativas se centrarán en el desarrollo de habilidades genéricas como el pensamiento crítico, la resolución de problemas, el trabajo en equipo y el uso efectivo de la tecnología, aspectos fundamentales en el ámbito profesional.

<sup>1</sup> Sistema de Asignación y Transferencia de Créditos Académicos



En este contexto, el profesor juega un papel crucial como facilitador y guía del proceso formativo, utilizando estrategias didácticas que fomenten la participación estudiantil, la investigación, el análisis crítico y la aplicación práctica del conocimiento adquirido. Se pondrá énfasis en proyectos integradores y casos de estudio que permitan a los alumnos vincular los conceptos matemáticos con sus aplicaciones reales, promoviendo así una educación integral que prepare a los futuros profesionales para enfrentar los desafíos del sector.

### 3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Observaciones
Tecnológico Nacional de México 23 de octubre del 2023	Representantes del Instituto Tecnológico de Chetumal	Presentación de la propuesta de la carrera de Ingeniería en Desarrollo de Aplicaciones.
Instituto Tecnológico de Querétaro Campus Norte del 19 al 22 de marzo 2024.	Representantes de los Institutos Tecnológicos de: Celaya, Chetumal, Ensenada, La Zona Olmeca, Querétaro, Villahermosa.  Tecnológicos Superiores de: Huetamo, Mario Molina Pasquel y Henríquez (unidad Mascota), Purhépecha.  Representante de Ciencias Básica de los Institutos de: Celaya, Morelia y CIIDET.	Diseño y/o desarrollo curricular de la carrera de Ingeniería en Desarrollo de Aplicaciones.
Tecnológico Nacional de México del 22 al 24 de abril del 2024.	Representantes de los Institutos Tecnológicos de Chetumal y Villahermosa.	Contraste y ajuste de las asignaturas Ing. en Desarrollo de Aplicaciones con respecto a las de Ing. en Inteligencia Artificial, Ing. en Ciberseguridad e Ingeniería en Ciencia de Datos.
Tecnológico Nacional de México del	Representantes de los Institutos Tecnológicos de: Celaya, Chetumal, Ensenada, La Zona Olmeca, Villahermosa.	Consolidación curricular de la carrera de Ingeniería en Desarrollo de Aplicaciones.



#### 4. Competencia(s) a desarrollar

##### Competencia(s) específica(s) de la asignatura

En el curso de “matemáticas discretas” se pretende que los estudiantes adquieran habilidades específicas clave para su desarrollo profesional. Al finalizar este curso, se espera que los estudiantes logren:

- Entiende y aplica los conceptos esenciales de las matemáticas discretas para resolver eficazmente problemas computacionales, demostrando competencia en el manejo de estructuras algebraicas, teoría de grafos y lógica matemática aplicada.
- Analiza y evalúa algoritmos y estructuras de datos basados en principios matemáticos discretos, garantizando su efectividad y adecuación a los requisitos específicos en aplicaciones.
- Integra conocimientos de matemáticas discretas para que, en asignaturas posteriores, puedan mejorar el diseño y análisis de sistemas de diferentes áreas disciplinares.

Estas habilidades están diseñadas para que los estudiantes no sólo obtengan un conocimiento teórico detallado, sino que también puedan poner en práctica este conocimiento en el desarrollo, análisis y mejora de tecnologías, preparándose para enfrentar los desafíos del ámbito profesional con una base sólida y habilidades prácticas.

#### 5. Competencias previas

Para garantizar un adecuado desarrollo de las habilidades en la asignatura “matemáticas discretas” se consideran esenciales ciertas competencias previas que los estudiantes deben haber adquirido:

- *Contar con conocimientos básicos en matemáticas generales*, abarcando álgebra, geometría y funciones, lo cual facilita la comprensión y aplicación de conceptos matemáticos más avanzados.
- *Tener habilidades iniciales en el uso de tecnologías de la información*, incluyendo el manejo básico de software matemático y herramientas computacionales. Esto resulta crucial para la modelación, análisis y simulación de problemas matemáticos discretos.
- *Poseer habilidades mínimas de comunicación matemática tanto oral como escrita*. Estas facilitan el intercambio de ideas, formular hipótesis y presentar soluciones matemáticas.

Estas competencias previas son indispensables para que los estudiantes puedan enfrentar con éxito los desafíos planteados por la asignatura “matemáticas discretas”, logrando así un entendimiento profundo y aplicado de los conceptos matemáticos fundamentales.



## 6. Temario

No.	Temas	Subtemas
1	Lógica matemática y sus aplicaciones.	1.1. Lenguaje de la lógica proposicional (LLP): proposiciones simples y compuestas. 1.2. Formalización (traducción) de enunciados del lenguaje natural al LLP. 1.3. Conectivos lógicos y tablas de verdad tautologías, contradicciones, contingencias y equivalencias lógicas. 1.4. Enfoques semántico y sintáctico de la lógica. 1.5. Análisis de argumentos con los dos enfoques el semántico y el sintáctico. 1.6. Lenguaje de la lógica de predicados: cuantificadores y evaluación de predicado. 1.7. Formalización (traducción) de enunciados del lenguaje natural al lenguaje de la lógica de predicados. 1.8. Métodos de demostración. 1.9. Aplicaciones de la lógica matemática.
2	Conjuntos, relaciones y funciones en computación.	2.1. Conceptos básicos de conjuntos y subconjuntos. 2.2. Operaciones con conjuntos. 2.3. Características, propiedades y aplicaciones de los conjuntos. 2.4. Producto cartesiano y relación binaria. 2.5. Representación y propiedades de las relaciones. 2.6. Relaciones de equivalencia y órdenes parciales. 2.7. Funciones y su clasificación. funciones de pertenencia. 2.8. Aplicaciones de conjuntos, relaciones y funciones.



3	Sistemas numéricos, aritmética modular y sus aplicaciones en computación.	<p>3.1. Introducción a los sistemas numéricos (binario, octal, decimal, hexadecimal).</p> <p>3.2. Conversiones entre sistemas numéricos.</p> <p>3.3. Operaciones básicas (suma, resta, multiplicación y división).</p> <p>3.4. Aritmética modular.</p> <p>3.5. Aplicación de los sistemas numéricos y aritmética modular en ciencias de la computación.</p>
4	Álgebra booleana y circuitos lógicos.	<p>4.1. Fundamentos del álgebra booleana: teoremas y postulados.</p> <p>4.2. Optimización de expresiones booleanas.</p> <p>4.3. Representación de expresiones booleanas con circuitos lógicos.</p> <p>4.4. Minitérminos y maxitérminos en el diseño de circuitos.</p> <p>4.5. Aplicaciones del álgebra booleana</p>
5	Teoría de grafos y su relevancia en ciencias de la computación	<p>5.1. Elementos básicos de la teoría de grafos.</p> <p>5.2. Tipos de grafos y sus propiedades.</p> <p>5.3. Representación de grafos (matemática y computacional).</p> <p>5.4. Algoritmos de recorrido y búsqueda en grafos.</p> <p>5.5. Árboles y su caracterización.</p> <p>5.6. Árboles generadores.</p> <p>5.7. Búsqueda en profundidad (DFS).</p> <p>5.8. Algoritmo de Prim y algoritmo de Kruskal.</p> <p>5.9. Aplicaciones de la teoría de grafos.</p>

## 7. Actividades de aprendizaje de los temas

1. Lógica matemática y su aplicación.	
Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i></p> <ul style="list-style-type: none"> <li>● Analiza y aplica la lógica proposicional y de predicados, utilizando proposiciones simples y compuestas en diferentes contextos.</li> <li>● Utiliza operaciones lógicas y tablas de verdad para evaluar proposiciones y argumentos.</li> </ul>	<ul style="list-style-type: none"> <li>● Taller de lógica proposicional: ejercicios prácticos en grupos para crear y evaluar proposiciones simples y compuestas, usando tablas de verdad y discutiendo su aplicación en el razonamiento de sistemas.</li> </ul>



<ul style="list-style-type: none"> <li>● Identifica tautologías, contradicciones y contingencias, y utiliza equivalencias lógicas y reglas de inferencia para construir y evaluar argumentos lógicos.</li> <li>● Aplica la lógica de predicados, incluyendo cuantificadores y evaluación de predicados, para mejorar la precisión de modelos y algoritmos.</li> <li>● Emplea álgebra declarativa e inducción matemática en la formulación y demostración de propiedades de algoritmos.</li> </ul> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> <li>● Desarrolla habilidades de pensamiento crítico y analítico a través de la evaluación y construcción de argumentos lógicos.</li> <li>● Fomenta la capacidad de abstracción y modelado matemático para representar problemas y soluciones.</li> <li>● Potencia la habilidad para trabajar en equipo y comunicar efectivamente resultados de análisis lógicos y matemáticos.</li> <li>● Promueve el uso de herramientas tecnológicas para la simulación y evaluación de argumentos lógicos y modelos matemáticos.</li> </ul>	<ul style="list-style-type: none"> <li>● Estudio de caso de tautologías y contradicciones: análisis individual de casos donde los estudiantes identifiquen tautologías y contradicciones en argumentos lógicos, aplicando estos conceptos al diseño de sistemas expertos.</li> <li>● Proyecto de lógica de predicados: desarrollo de un miniproyecto grupal donde se aplique lógica de predicados a un problema específico, diseñando un sistema de inferencia basado en reglas y utilizando cuantificadores para mejorar la precisión del modelo.</li> <li>● Simulación de argumentos válidos e inválidos: utilización de software de lógica para simular y evaluar argumentos válidos e inválidos, fomentando la comprensión de cómo la lógica sustenta el razonamiento.</li> <li>● Aplicación de inducción matemática. Ejercicio individual donde los estudiantes utilicen la inducción matemática para probar la corrección de algoritmos aplicados, como la búsqueda o el ordenamiento, destacando la importancia de una base lógica sólida.</li> </ul>
<b>2. Conjuntos, relaciones y funciones en computación</b>	
<b>Competencias</b>	<b>Actividades de aprendizaje</b>
<p><i>Específica(s):</i></p> <ul style="list-style-type: none"> <li>● Entiende y aplica los conceptos básicos de conjuntos y subconjuntos, reconociendo su importancia en la estructuración de datos para computación.</li> <li>● Realiza operaciones con conjuntos y</li> </ul>	<ul style="list-style-type: none"> <li>● Exploración de conjuntos y subconjuntos: creación de actividades interactivas en grupos para que los alumnos identifiquen y organicen conjuntos y subconjuntos en diversos contextos, incluyendo bases de datos y estructuras de datos en programación.</li> </ul>



<p>analiza sus características y propiedades para resolver problemas computacionales.</p> <ul style="list-style-type: none"> <li>● Comprende y utiliza relaciones y funciones, incluyendo el producto cartesiano y relaciones binarias, para modelar y resolver problemas.</li> <li>● Aplica el conocimiento de conjuntos, relaciones y funciones en el diseño y análisis de algoritmos, identificando sus aplicaciones prácticas.</li> </ul> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> <li>● Desarrolla habilidades analíticas y de pensamiento crítico al explorar y aplicar conceptos matemáticos complejos en situaciones reales.</li> <li>● Fomenta la colaboración y comunicación efectiva a través del trabajo en equipo y la presentación de soluciones a problemas.</li> <li>● Promueve la autogestión del aprendizaje mediante la investigación y exploración de aplicaciones de conjuntos, relaciones y funciones.</li> <li>● Potencia la habilidad para utilizar herramientas tecnológicas en la simulación y modelado de problemas relacionados con la asignatura.</li> </ul>	<ul style="list-style-type: none"> <li>● Taller de operaciones con conjuntos: sesión práctica individual donde los estudiantes resuelvan problemas utilizando operaciones como unión, intersección, diferencia y complemento con conjuntos. Estas operaciones se aplicarán en la filtración y manipulación de datos.</li> <li>● Análisis de relaciones y funciones: desarrollo de un proyecto en equipos reducidos donde los alumnos modelen relaciones matemáticas para resolver problemas específicos, por ejemplo, el emparejamiento de datos o la clasificación. Deberán explicar cómo este modelo se implementa en un algoritmo.</li> <li>● Aplicaciones prácticas: realización de un miniproyecto investigativo donde los estudiantes exploren y presenten cómo se utilizan conjuntos, relaciones y funciones en algoritmos actuales, por ejemplo, su uso en redes neuronales o sistemas de recomendación, etc. Se utilizará software matemático o programación para demostrar ejemplos prácticos.</li> </ul>
<p><b>3. Sistemas numéricos, aritmética modular y sus aplicaciones en computación</b></p>	
<p>Competencias</p>	<p>Actividades de aprendizaje</p>
<p><i>Específica(s):</i></p> <ul style="list-style-type: none"> <li>● Comprende los fundamentos y la importancia de los sistemas numéricos binario, octal, decimal y hexadecimal en el contexto de la computación.</li> <li>● Realiza conversiones eficientes entre diferentes sistemas numéricos, aplicando métodos matemáticos</li> </ul>	<ul style="list-style-type: none"> <li>● Introducción a los sistemas numéricos: creación de un mapa conceptual en grupo que ilustra los diferentes sistemas numéricos y sus aplicaciones en computación. Este trabajo fomentará la investigación y el aprendizaje colaborativo.</li> <li>● Conversiones entre sistemas numéricos: desarrollo de una serie de ejercicios prácticos</li> </ul>



<p>precisos.</p> <ul style="list-style-type: none"> <li>● Aplica operaciones básicas dentro de diferentes sistemas numéricos para resolver problemas específicos en computación.</li> <li>● Identifica y utiliza aplicaciones prácticas de los sistemas numéricos en la computación y desarrollo de algoritmos.</li> </ul> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> <li>● Desarrolla habilidades de razonamiento lógico y matemático mediante la resolución de problemas relacionados con sistemas numéricos.</li> <li>● Fomenta la capacidad de análisis y síntesis al convertir y operar con números en distintos sistemas numéricos.</li> <li>● Promueve el trabajo colaborativo a través de actividades grupales que implican discusión y resolución de problemas.</li> <li>● Impulsa el uso eficiente de herramientas tecnológicas para la simulación y aplicación de conceptos aprendidos en contextos de computación.</li> </ul>	<p>individuales y en equipos donde los estudiantes conviertan números entre diferentes sistemas numéricos. Se podría incluir el uso de software matemático para verificar resultados.</p> <ul style="list-style-type: none"> <li>● Operaciones básicas: implementación de un taller práctico donde los estudiantes, en parejas, realicen operaciones básicas en diferentes sistemas numéricos y presenten sus aplicaciones en problemas reales de computación, utilizando plataformas de programación.</li> <li>● Aplicaciones en computación. Desarrollo de un mini proyecto donde los estudiantes apliquen los sistemas numéricos en un contexto, por ejemplo, puede ser el desarrollo de un algoritmo simple que use representaciones binarias para la toma de decisiones o la manipulación de datos. Este proyecto se presentará al final del tema, fomentando la investigación, la aplicación práctica de conocimientos y el uso de tecnología.</li> </ul>
--	---

**4. Álgebra booleana y Circuitos lógicos**

Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i></p> <ul style="list-style-type: none"> <li>● Domina los fundamentos del álgebra booleana, incluyendo teoremas y postulados, para la conceptualización y análisis de problemas lógicos.</li> <li>● Optimiza expresiones booleanas para la eficiencia en la representación y procesamiento de</li> </ul>	<ul style="list-style-type: none"> <li>● Taller de fundamentos del álgebra booleana: ejercicios grupales para explorar teoremas y postulados del álgebra booleana, incluyendo la construcción de tablas de verdad y la simplificación de expresiones booleanas.</li> <li>● Optimización de expresiones booleanas: sesión de prácticas donde los estudiantes, individualmente, aplican técnicas de</li> </ul>



<p>datos, crucial en el desarrollo de algoritmos.</p> <ul style="list-style-type: none"> <li>● Diseña y representa circuitos lógicos a partir de expresiones booleanas, aplicando conocimientos de minitérminos y maxitérminos.</li> <li>● Aplica el álgebra booleana en la creación de modelos computacionales, destacando su utilidad en la fundamentación de sistemas.</li> </ul> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> <li>● Fomenta habilidades de razonamiento lógico y análisis crítico al trabajar con estructuras booleanas y circuitos lógicos.</li> <li>● Desarrolla competencias en diseño y modelado a través de la creación de circuitos lógicos que solucionan problemas específicos.</li> <li>● Potencia la colaboración y comunicación efectiva mediante el trabajo en equipo para el diseño y evaluación de circuitos lógicos.</li> <li>● Promueve el uso eficiente de herramientas tecnológicas para simular y evaluar circuitos y modelos booleanos.</li> </ul>	<p>simplificación como mapas de Karnaugh para optimizar expresiones booleanas, evaluando su impacto en la eficiencia computacional.</p> <ul style="list-style-type: none"> <li>● Diseño de circuitos lógicos: proyecto en pequeños grupos para diseñar y representar circuitos lógicos a partir de expresiones booleanas dadas, haciendo uso de minitérminos y maxitérminos, y luego simular su funcionamiento utilizando software especializado.</li> <li>● Creación de modelos computacionales: actividad final donde los estudiantes aplican los conocimientos adquiridos para desarrollar un modelo computacional, por ejemplo, basado en álgebra booleana que resuelva un problema específico. Este modelo se presentará y discutirá en clase, destacando la aplicación práctica del álgebra booleana en sistemas.</li> </ul>
<b>5. Teoría de grafos y su relevancia en ciencias de la computación</b>	
Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i></p> <ul style="list-style-type: none"> <li>● Comprende los elementos básicos y propiedades de la teoría de grafos aplicándolos en diferentes contextos.</li> <li>● Diferencia y utiliza diversos tipos de grafos para la representación eficaz de estructuras de datos complejas.</li> <li>● Implementa representaciones matemáticas y computacionales de</li> </ul>	<ul style="list-style-type: none"> <li>● Introducción a la teoría de grafos: sesiones prácticas en grupo donde los estudiantes construyen y clasifican diferentes tipos de grafos, identificando sus propiedades únicas mediante la utilización de software de grafos.</li> <li>● Laboratorio de representación de grafos: actividad individual en la que los estudiantes aplican técnicas matemáticas y computacionales para representar grafos,</li> </ul>



<p>grafos, facilitando su análisis y manipulación en algoritmos.</p> <ul style="list-style-type: none"> <li>● Desarrolla y aplica algoritmos de recorrido y búsqueda en grafos, esenciales en el procesamiento y análisis de datos.</li> <li>● Identifica y explora aplicaciones de la teoría de grafos como el análisis de redes sociales, sistemas de recomendación y problemas de optimización.</li> </ul> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> <li>● Fomenta habilidades de pensamiento crítico y solución de problemas mediante la aplicación de teoría de grafos en diferentes contextos.</li> <li>● Desarrolla la capacidad para trabajar en equipo y comunicar soluciones complejas de manera efectiva, especialmente en proyectos que involucran estructuras de datos avanzadas.</li> <li>● Impulsa la creatividad e innovación en el diseño de soluciones basadas en grafos.</li> <li>● Promueve el aprendizaje autónomo y el uso efectivo de recursos tecnológicos para la simulación y análisis de grafos.</li> </ul>	<p>utilizando herramientas de programación como Python o software específico de grafos.</p> <ul style="list-style-type: none"> <li>● Taller de algoritmos en grafos: ejercicios en equipos donde los estudiantes implementan y comparan algoritmos de recorrido y búsqueda, como DFS (búsqueda en profundidad) y BFS (búsqueda en anchura), analizando su eficiencia y aplicabilidad en problemas reales.</li> <li>● Proyecto de aplicaciones de grafos: proyecto final grupal en el que los estudiantes eligen una aplicación específica de la teoría de grafos, desarrollan un modelo o algoritmo relacionado y presentan sus resultados, demostrando cómo la teoría de grafos contribuye a soluciones innovadoras.</li> </ul>
--	--

## 8. Práctica(s)

### **Práctica 1: Aplicación de lógica matemática en sistemas computacionales**

Objetivo: comprender la importancia de la lógica matemática en el desarrollo y fundamentación de sistemas.

Actividades sugeridas: individualmente o en pequeños grupos, los estudiantes desarrollarán un sistema de inferencia simple o un conjunto de reglas para un sistema experto, utilizando lógica proposicional o de predicados. Deberán demostrar cómo su sistema puede aplicarse en un escenario especificado.

Competencias desarrolladas: innovación y creatividad, habilidades de programación, comprensión profunda de la lógica aplicada en Ciencias de la Computación.



### **Práctica 2: Modelado de problemas con conjuntos y relaciones**

Objetivo: aplicar teorías de conjuntos y relaciones para modelar y resolver problemas relacionados con estructuras de datos y su manipulación.

Actividades sugeridas: en equipos, los estudiantes identificarán y modelarán un problema de clasificación o agrupación de datos usando conjuntos, subconjuntos y relaciones. Presentarán sus soluciones mediante diagramas y una breve exposición sobre la aplicabilidad.

Competencias desarrolladas: capacidad de abstracción y modelado, trabajo en equipo y comunicación efectiva, análisis y síntesis de información.

### **Práctica 3: Conversiones y operaciones en sistemas numéricos**

Objetivo: reforzar la comprensión de los sistemas numéricos y su aplicabilidad en las ciencias de la computación.

Actividades sugeridas: los estudiantes realizarán conversiones entre distintos sistemas numéricos (binario, decimal, octal, hexadecimal) y aplicarán operaciones básicas en cada sistema. Esta actividad culminará con la implementación de un pequeño programa que realice conversiones automáticamente.

Competencias desarrolladas: razonamiento lógico matemático, aplicación de conocimientos teóricos en prácticas de programación, desarrollo de habilidades en el uso de tecnologías de la información.

### **Práctica 4: Diseño y simulación de circuitos lógicos**

Objetivo: consolidar el entendimiento del álgebra booleana y su aplicación en el diseño de circuitos lógicos y modelos computacionales.

Actividades sugeridas: los estudiantes usarán software de simulación de circuitos para diseñar y probar circuitos lógicos basados en especificaciones dadas, analizando su relevancia en la resolución de problemas.

Competencias desarrolladas: habilidades en el diseño y análisis de sistemas lógicos, aplicación práctica de teorías matemáticas, manejo de herramientas tecnológicas especializadas.

### **Práctica 5: Grafos y árboles**

Objetivo: analizar algoritmos para diferentes problemas que se modelan con grafos y árboles.

Actividades sugeridas: los estudiantes usarán software de simulación de algoritmos para analizar su relevancia en la resolución de problemas.

Competencias desarrolladas: habilidades en análisis de algoritmos, aplicación práctica de teorías matemáticas, manejo de herramientas tecnológicas especializadas.

Estas prácticas integran los contenidos de la asignatura de Matemáticas Discretas promoviendo el desarrollo de competencias tanto específicas como genéricas, y preparando a los estudiantes para enfrentar desafíos profesionales en el campo de diferentes áreas de las Ciencias de la Computación.



## 9. Proyecto de asignatura

El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:

- **Fundamentación:** marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo.
- **Planeación:** con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.
- **Ejecución:** consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.
- **Evaluación:** es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

## 10. Evaluación por competencias

- **Proyectos integradores:** evaluación basada en el desarrollo de proyectos que aplican conceptos de la asignatura a problemas reales de inteligencia artificial. Estos proyectos permitirán a los estudiantes demostrar su capacidad para integrar conocimientos y habilidades en la creación de soluciones innovadoras, evaluando tanto el proceso de desarrollo como el resultado final.
- **Presentaciones orales:** cada estudiante o grupo presentará los resultados de sus proyectos o investigaciones ante la clase. Esta actividad evalúa la capacidad de comunicar ideas complejas de manera efectiva, el uso de soportes visuales adecuados y la habilidad para responder preguntas y recibir retroalimentación constructiva.
- **Exámenes escritos:** se realizarán exámenes que incluyan problemas aplicados, preguntas teóricas y casos prácticos para evaluar la comprensión de conceptos fundamentales de la teoría de grafos, álgebra booleana, lógica matemática y su aplicación en Ciencias de la Computación. Los exámenes escritos permitirán medir el conocimiento individual de cada estudiante de manera objetiva.
- **Evaluación Continua de Tareas y Ejercicios:** a lo largo del semestre, se asignarán tareas y ejercicios que cubran diferentes temas de la asignatura. Esta evaluación continua facilita el seguimiento del progreso del estudiante, permitiendo identificar áreas de fortaleza y oportunidades de mejora, además de fomentar la práctica constante de los conceptos aprendidos.



Estos métodos de evaluación están diseñados para proporcionar una visión integral del desempeño de los estudiantes, fomentando el aprendizaje activo, la aplicación práctica de conocimientos y el desarrollo de habilidades clave para su futuro profesional en las ciencias de la computación.

## 11. Fuentes de información

1. Biggs, N. L. (2002). *Discrete Mathematics* (2ª ed.). Oxford University Press.
2. Chartrand, G., & Zhang, P. (2012). *A First Course in Graph Theory*. Dover Publications.
3. Dossey, J. A., Otto, A. D., Spence, L. E., & Eynden, C. V. (2006). *Discrete Mathematics* (5ª ed.). Pearson.
4. Epp, S. S. (2010). *Discrete Mathematics with Applications* (4ª ed.). Brooks Cole.
5. Gersting, J. L. (2007). *Mathematical Structures for Computer Science* (7ª ed.). W. H. Freeman.
6. Goodaire, E. G., & Parmenter, M. M. (2002). *Discrete Mathematics with Graph Theory* (3ª ed.). Prentice Hall.
7. Grimaldi, R. P. (2003). *Matemáticas discretas y combinatoria* (5ª ed.). Wiley.
8. Gross, J. L., & Yellen, J. (2005). *Graph Theory and Its Applications* (2ª ed.). Chapman and Hall/CRC.
9. Harary, F. (1994). *Graph Theory*. Addison-Wesley Publishing Company.
10. Johnsonbaugh, R. (2018). *Discrete Mathematics* (9ª ed.). Pearson.
11. Knuth, D. E. (2011). *The Art of Computer Programming, Volume 4A: Combinatorial Algorithms, Part 1*. Addison-Wesley Professional.
12. Kolman, B., Busby, R. C., & Ross, S. (2008). *Discrete Mathematical Structures* (6ª ed.). Pearson.
13. Lipson, M. (2000). *Discrete Mathematics Demystified*. McGraw-Hill Education.
14. Liu, C. L. (2008). *Elements of Discrete Mathematics* (4ª ed.). McGraw-Hill Higher Education.
15. Rosen, K. H. (2012). *Matemáticas discretas y sus aplicaciones* (7ª ed.). McGraw-Hill Education.
16. Scheinerman, E. R. (2012). *Mathematics: A Discrete Introduction* (3ª ed.). Cengage Learning.
17. Sedgewick, R., & Wayne, K. (2011). *Algorithms* (4ª ed.). Addison-Wesley.
18. Truss, J. K. (1999). *Discrete Mathematics for Computer Scientists*. Addison Wesley.
19. Tucker, A. (2004). *Applied Combinatorics* (5ª ed.). Wiley.
20. Wilson, R. J. (2002). *Introduction to Graph Theory* (5ª ed.). Pearson.