

NÚMERO Y NOMBRE DE LA UNIDAD:**HORAS: (7/7) 14**

1. Lógica matemática.

OBJETIVO DE LA UNIDAD:

El estudiante utilizará las leyes básicas del razonamiento matemático, que le permitan estar capacitado en el desarrollo de algoritmos necesarios para un programa o sistema de programas.

TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
1.1 Proposiciones simples 1.2 Proposiciones compuestas 1.3 Operadores lógicos básicos 1.3.1 Conjunción 1.3.2 Disyunción 1.3.3 Negación 1.3.4 Condicional 1.4 Tablas de verdad 1.4.1 Generación de nuevos operadores lógicos 1.5 Tautologías 1.6 Contradicciones 1.7 Demostración de teoremas por tablas de verdad 1.8 Álgebra de proposiciones 1.8.1 Propiedades de idempotencia. 1.8.2 Propiedades conmutativas. 1.8.3 Propiedades asociativas. 1.8.4 Propiedades distributivas. 1.8.5 Leyes de DeMorgan 1.8.6 Propiedad del Condicional 1.9 Demostración de Teoremas Mediante Algebra de Proposiciones 1.10 Reglas de Inferencia Lógica 1.10.1 Modus Ponens 1.10.2 Modus Tollens 1.10.3 Regla de Adjunción 1.10.4 Regla de Simplificación 1.10.5 Modus Tollendo Ponens 1.10.6 Regla del Silogismo Hipotético 1.11 Demostración de Teoremas Mediante Reglas de Inferencia Lógica y Algebra de	<ul style="list-style-type: none"> Exposición por parte del profesor (estrategia de recepción). Discusiones facilitadas por el profesor (estrategia interpersonal). 	<ul style="list-style-type: none"> Identificar los operadores lógicos básicos y demostrar mediante tablas de verdad sus propiedades a fin de construir un Álgebra de proposiciones (DD). Demostrar teoremas mediante la aplicación del Álgebra de proposiciones y las Reglas básicas de inferencia lógica (DD). Identificar tres métodos básicos de demostración de teoremas y las situaciones en las que son aplicables: demostración por contradicción, demostración por vacuidad y demostraciones por el principio de inducción matemática (DD). Elaboración de ejercicios en clase (DD). 	<ul style="list-style-type: none"> Realizar ejercicios que comprendan el uso de tablas de verdad (DD). Demostrar teoremas mediante tablas de verdad y mediante Álgebra de proposiciones y reglas de inferencia lógica (DD). Demostrar teoremas que involucren la aplicación del principio de inducción matemática (DD).

<p>Proposiciones</p> <p>1.12 Demostración de teoremas por Contradicción</p> <p>1.13 Cuantificadores</p> <p> 1.13.1 Cuantificador existencial</p> <p> 1.13.2 Cuantificador universal</p> <p> 1.13.3 Negación de cuantificadores</p> <p> 1.13.4 Demostración de teoremas por vacuidad</p> <p>1.14 Inducción Matemática</p>			
--	--	--	--

<p>ESCENARIOS</p>	<p>ESTRATEGIAS DE EVALUACIÓN</p>	<p>RECURSOS DIDÁCTICOS Y/ O SOFTWARE</p>
<ul style="list-style-type: none"> • Aula 	<ul style="list-style-type: none"> • Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI). • Interrogatorio: Exámenes escritos (DD). • Demostración: Validación de Teoremas (DD, CE). 	<ul style="list-style-type: none"> • Pizarrón • Proyector de acetatos • Cañón de proyección de video • Equipo de cómputo

NÚMERO Y NOMBRE DE LA UNIDAD:**HORAS:(7/7) 14**

2. Teoría de Conjuntos

OBJETIVO DE LA UNIDAD:

El estudiante usará las operaciones y leyes básicas de la teoría de conjuntos, que le permitan reconocer la estructura subyacente de una formulación concisa del tema matemático en cuestión.

TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
2.1 Conjunto y pertenencia 2.2 Subconjuntos y el Axioma de extensión 2.3 El Axioma de especificación 2.4 Extensión y comprensión de un conjunto 2.5 Cardinalidad de un conjunto 2.5.1 Conjuntos finitos 2.5.2 Conjuntos infinitos 2.5.3 Conjuntos contables 2.5.4 Conjuntos no contables 2.5.5 Conjunto Universo y conjunto vacío 2.6 Diagramas de Venn – Euler. 2.7 Axiomatización de las Operaciones con conjuntos 2.7.1 Unión 2.7.2 Intersección 2.7.3 Complemento 2.7.4 Diferencia de conjuntos 2.7.5 Diferencia simétrica de conjuntos 2.7.6 Conjunto Potencia 2.7.7 Producto Cartesiano 2.8 Álgebra de conjuntos 2.8.1 Propiedades de idempotencia 2.8.2 Propiedades conmutativas 2.8.3 Propiedades asociativas 2.8.4 Propiedades distributivas 2.8.5 Leyes de DeMorgan 2.8.6 Propiedad de la diferencia de conjuntos 2.9 Demostración de Teoremas Mediante Algebra de Conjuntos 2.10 Familias de conjuntos	<ul style="list-style-type: none"> Exposición por parte del profesor (estrategia de recepción). Discusiones facilitadas por el profesor (estrategia interpersonal). 	<ul style="list-style-type: none"> Identificar y distinguir los conceptos de conjunto, subconjunto, conjunto universo, conjunto vacío, sus características y los tipos de cardinalidades que pueden existir en éstos (DD) Enlistar los operadores básicos de conjuntos y demostrar, a partir de sus definiciones, sus propiedades a fin de construir un Álgebra de conjuntos (DD) Demostrar teoremas mediante la aplicación del Álgebra de conjuntos (DD) Elaboración ejercicios en clase (DD). 	<ul style="list-style-type: none"> Resolver ejercicios aplicando las operaciones básicas de conjuntos (DD). Resolver ejercicios en los que se identifique la cardinalidad y contabilidad de conjuntos dados (DD). Demostrar teoremas mediante diagramas de Venn-Euler y mediante Álgebra de conjuntos (DD).

ESCENARIOS	ESTRATEGIAS DE EVALUACIÓN	RECURSOS DIDÁCTICOS Y/ O SOFTWARE
<ul style="list-style-type: none">Aula	<ul style="list-style-type: none">Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI).Interrogatorio: Exámenes escritos (DD).Demostración: Validación de Teoremas (DD, CE).	<ul style="list-style-type: none">PizarrónProyector de acetatosCañón de proyección de videoEquipo de cómputo

NÚMERO Y NOMBRE DE LA UNIDAD:**HORAS: (6/6) 12**

3. Retículas y álgebra Booleana

OBJETIVO DE LA UNIDAD:

El estudiante demostrará con el estudio del álgebra Booleana el caso en el que la estructura depende principalmente de dos operaciones binarias cerradas.

TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
3.1 Retículas 3.2 Diagrama de Hasse de una Retícula 3.3 Retículas complementarias 3.4 Caracterización de retículas complementarias mediante el diagrama de Hasse 3.5 Algebras de Boole y sus axiomas 3.6 Teoremas básicos <ul style="list-style-type: none"> 3.6.1 Propiedades de idempotencia 3.6.2 Propiedades de absorción 3.6.3 Propiedades asociativas 3.6.4 Propiedades de unicidad 3.6.5 Leyes de DeMorgan 3.7 Algebras de Boole como retículas 3.8 El teorema de representación de Stone 3.9 Algebra de Boole de 2^n elementos 3.10 Operadores lógicos como Algebra de Boole 3.11 Funciones Booleanas 3.12 Mapas de Karnaugh <ul style="list-style-type: none"> 3.12.1 Caso de dos variables 3.12.2 Caso de tres variables 3.12.3 Caso de cuatro variables 	<ul style="list-style-type: none"> • Exposición por parte del profesor (estrategia de recepción). • Discusiones facilitadas por el profesor (estrategia interpersonal). 	<ul style="list-style-type: none"> • Identificar mediante ejemplos el concepto de retícula y construir diagramas de Hasse para la representación de estas estructuras algebraicas (DD). • Enlistar los axiomas que definen a un Álgebra de Boole y demostrar, a partir de tales axiomas, sus propiedades básicas (DD). • Identificar la caracterización de las Álgebras de Boole como retículas (DD). • Construir el modelado de los operadores lógicos básicos (conjunción, disyunción y negación) como un Álgebra de Boole (DD). • Identificar el proceso de simplificación de funciones Booleanas, las cuales son consecuencia del Álgebra de Boole definida por los operadores lógicos, mediante mapas de Karnaugh (DD). 	<ul style="list-style-type: none"> • Resolver ejercicios en los que se identifique si un conjunto bajo una operación puede ser caracterizado como una retícula (DD). • Demostrar matemáticamente si un conjunto bajo operaciones dadas puede ser caracterizado como un Álgebra de Boole (DD). • Resolver ejercicios en los que se requiera la simplificación de funciones Booleanas de dos a cuatro variables (DD). • Demostrar teoremas que permitan justificar propiedades asociadas a conjuntos caracterizados como Álgebras de Boole (DD).

ESCENARIOS	ESTRATEGIAS DE EVALUACIÓN	RECURSOS DIDÁCTICOS Y/ O SOFTWARE
<ul style="list-style-type: none">Aula	<ul style="list-style-type: none">Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI).Interrogatorio: Exámenes escritos (DD).Demostración: Validación de Teoremas (DD, CE).	<ul style="list-style-type: none">PizarrónProyector de acetatosCañón de proyección de videoEquipo de cómputo

NÚMERO Y NOMBRE DE LA UNIDAD:**HORAS: (7/7) 14**

4. Preliminares de la teoría de grafos

OBJETIVO DE LA UNIDAD:

El estudiante utilizará, a diferencia de las gráficas continuas de los primeros cursos de álgebra, la estructura finita de un grafo, y su importancia en el uso de análisis de relaciones.

TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
4.1 Grafos, seudógrafos, multígrafos y grafos dirigidos 4.2 Complemento de un grafo. 4.3 Rutas y circuitos 4.3.1 Rutas simples 4.3.2 Rutas cerradas 4.3.3 Circuitos 4.3.4 Vías 4.4 Grafos completos. 4.5 Grafos regulares. 4.6 Grafos bipartitas. 4.7 Grafos Eulerianos y Hamiltonianos. 4.8 Grafos planos y grafos planares 4.9 Teorema de Kuratowski 4.10 Grafos Isomorfos 4.10.1 Invariantes 4.11 Grafos Homeomorfos 4.12 Árboles 4.13 Coloraciones en grafos y grafos planos 4.14 Número cromático de un grafo 4.15 Fórmula de Euler 4.16 Representaciones Computacionales de Grafos 4.16.1 Matrices de adyacencias 4.16.2 Listas de adyacencias	<ul style="list-style-type: none"> Exposición por parte del profesor (estrategia de recepción). Discusiones facilitadas por el profesor (estrategia interpersonal). 	<ul style="list-style-type: none"> Enlistar las definiciones, características, tipos y aplicación de los grafos y resolver ejemplos que comprendan el uso de éstos (DD). Identificar condiciones necesarias y/o suficientes para identificar grafos Eulerianos, Hamiltonianos, Isomorfos, Planos o colorables. Resolver ejemplos de aplicaciones de tales condiciones (DD). Construir matrices de adyacencias y listas de adyacencias para la representación de grafos (DD). Resolver ejercicios en clase de aplicaciones de la fórmula de Euler y el número cromático (DD). 	<ul style="list-style-type: none"> Resolver ejercicios en los que se requiera representar un problema dado mediante un grafo (DD). Enlistar el conjunto de propiedades asociadas a un grafo dado a fin de caracterizarlo (DD). Resolver ejercicios en los que se aplique el Teorema de Kuratowski a un grafo a fin de determinar su posible planaridad y representarlo en su forma plana si es posible (DD).

ESCENARIOS	ESTRATEGIAS DE EVALUACIÓN	RECURSOS DIDÁCTICOS Y/ O SOFTWARE
<ul style="list-style-type: none">Aula	<ul style="list-style-type: none">Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI).Interrogatorio: Exámenes escritos (DD).Demostración: Validación de Teoremas (DD, CE).	<ul style="list-style-type: none">PizarrónProyector de acetatosCañón de proyección de videoEquipo de cómputo

NÚMERO Y NOMBRE DE LA UNIDAD: 5. Algoritmos y leguajes de programación de alto nivel		HORAS: (5/5) 10	
OBJETIVO DE LA UNIDAD: El estudiante usará el concepto de algoritmo y su importancia en el diseño e implementación de los lenguajes de programación.			
TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
5.1 Definición y características de Algoritmos 5.2 Representación de algoritmos 5.3 Diseño de algoritmos 5.4 Implantación de lenguajes de programación de alto nivel 5.5 Diseño de un lenguaje de programación de alto nivel 5.5.1 Pseudocódigo 5.6 Componentes de un lenguaje de programación de alto nivel 5.7 Compiladores	Exposición por parte del profesor (estrategia de recepción). Discusiones facilitadas por el profesor (estrategia interpersonal). Trabajo individual o grupal por parte de los estudiantes (estrategia de selección).	<ul style="list-style-type: none"> • Identificar el concepto de algoritmo y sus características básicas. Además resolver ejemplos de problemas que pueden ser modelados mediante algoritmos (DD). • Enlistar las fases básicas detrás del diseño de un algoritmo, sus representaciones gráficas y su implantación mediante un lenguaje de programación de alto nivel haciendo énfasis en la codificación mediante pseudocódigo (DD). • Identificar los componentes básicos de un compilador y la manera en que éste permite la conversión de un algoritmo implementado en un lenguaje de alto nivel a una implementación en bajo nivel a ser ejecutada por un equipo de cómputo (DD). • Resolver ejercicios en clase (DD). 	Realizar ejercicios diseñando algoritmos e implementándolos en un lenguaje de programación de alto nivel o bien mediante pseudocódigo (DD).

ESCENARIOS	ESTRATEGIAS DE EVALUACIÓN	RECURSOS DIDÁCTICOS Y/ O SOFTWARE
<ul style="list-style-type: none">Aula	<ul style="list-style-type: none">Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI).Interrogatorio: Exámenes escritos (DD).Demostración: Validación de Teoremas (DD, CE).	<ul style="list-style-type: none">PizarrónProyector de acetatosCañón de proyección de videoEquipo de cómputo

NÚMERO Y NOMBRE DE LA UNIDAD: 6. Métodos y análisis de algoritmos		HORAS: (3/3) 6	
OBJETIVO DE LA UNIDAD: El estudiante aplicará los principales métodos en el diseño de algoritmos, llevando a cabo el análisis e implementación de los mismos, para la resolución de problemas aplicados.			
TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
6.1 El método de estructuras iterativas 6.2 Estructuras recursivas 6.2.1 Enfoque “divide y vencerás”	<ul style="list-style-type: none"> Exposición por parte del profesor (estrategia de recepción). Discusiones facilitadas por el profesor (estrategia interpersonal). Trabajo individual o grupal por parte de los estudiantes (estrategia de selección). 	<ul style="list-style-type: none"> Identificar mediante la solución de ejemplos, la aplicación de estructuras recursivas e iterativas y su implementación en un lenguaje de programación haciendo énfasis en la codificación mediante pseudocódigo (DD). Elaborar ejercicios en clase (DD). 	Resolver ejercicios de diseño de algoritmos aplicando los conceptos de recursividad, estructuras iterativas o el enfoque “divide y vencerás” (DD).

ESCENARIOS	ESTRATEGIAS DE EVALUACIÓN	RECURSOS DIDÁCTICOS Y/ O SOFTWARE
<ul style="list-style-type: none"> Aula 	<ul style="list-style-type: none"> Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI). Interrogatorio: Exámenes escritos (DD). Demostración: Validación de Teoremas (DD, CE). 	<ul style="list-style-type: none"> Pizarrón Proyector de acetatos Cañón de proyección de video Equipo de cómputo

NÚMERO Y NOMBRE DE LA UNIDAD:

7. Limitaciones y robustez

HORAS: (10/10) 20**OBJETIVO DE LA UNIDAD:**

El estudiante usará los conceptos de limitación y robustez como consecuencia del diseño y análisis de algoritmos.

TEMAS Y SUBTEMAS	ESTRATEGIAS DE INSTRUCCIÓN *	EXPERIENCIAS DE APRENDIZAJE	
		Con Docente	Independientes**
7.1 Máquinas de Turing 7.2 Funciones Computables 7.3 Notación Asintótica 7.3.1 Notación Theta 7.3.2 Notación O-Grande 7.3.3 Notación Omega-Grande 7.3.4 Propiedades de las Notaciones 7.4 Definición de Tiempo de ejecución 7.5 Análisis de complejidad 7.5.1 Modelo de máquina de Memoria de Acceso Aleatorio 7.5.2 Análisis de complejidad de algoritmos Iterativos 7.5.3 Análisis de complejidad de algoritmos Recursivos 7.5.4 Solución de Recurrencias 7.5.5 Solución de Recurrencias por el Teorema Maestro 7.6 Clasificación de los problemas 7.6.1 Definición y ejemplos de problemas P 7.6.2 Definición y ejemplos de problemas NP-Complejos 7.6.3 Definición y ejemplos de problemas NP-difíciles 7.6.4 Definición y ejemplos de problemas no computables 7.6.5 El Problema de la Parada	<ul style="list-style-type: none"> Exposición por parte del profesor (estrategia de recepción). Discusiones facilitadas por el profesor (estrategia interpersonal). 	<ul style="list-style-type: none"> Identificar la importancia de las máquinas de Turing y las funciones computables como el fundamento matemático detrás de la teoría de la complejidad de algoritmos (DD). Enlistar y memorizar los conceptos de notación asintótica y sus tres variantes (theta, o-grande y omega grande) como una herramienta para la expresión de tiempos de ejecución de algoritmos (DD). Identificar mediante la presentación de ejemplos de algoritmos, la determinación de la complejidad de los mismos, en cuanto al tiempo, espacio y el número de pasos para su ejecución (DD). Enlistar mediante ejemplos la clasificación de problemas de acuerdo a los tiempos de ejecución de los algoritmos que los solucionan. También se identificará al Problema de la Parada como un ejemplo de un problema no computable (DD). 	

ESCENARIOS	ESTRATEGIAS DE EVALUACIÓN	RECURSOS DIDÁCTICOS Y/ O SOFTWARE
<ul style="list-style-type: none"> Aula 	<ul style="list-style-type: none"> Resolución de Casos: Tareas y Ejercicios en clase (DD, CE y DI). Interrogatorio: Exámenes escritos (DD). Demostración: Validación de Teoremas (DD, CE). 	<ul style="list-style-type: none"> Pizarrón Proyector de acetatos Cañón de proyección de video Equipo de cómputo

** Incluir el desarrollo de habilidades de investigación en caso de ser pertinente.*

*** Desarrollo de proyectos de investigación*

REFERENCIAS BIBLIOGRÁFICAS

BÁSICA:

Aho, Alfred V. (1974). *The design and Analysis of Computer Algorithms*. Edit. Addison Wesley. Reading, Mass. (Clásico)

Johnsonbaugh, Richard. (1999). *Matemáticas Discretas*. México: Edit. Prentice Hall.

Wirth, Niklaus. (1987). *Algoritmos y estructuras de datos*. México: Edit. Prentice Hall.

COMPLEMENTARIA:

Bradley, J. (1988). *Introduction to Discrete Mathematics*. USA: Edit. Addison Wesley.

Rosen, K. H. (1991). *Discrete Mathematics and its Applications*. USA: Edit. Mc Graw Hill.

SWOKOWSKY, E.W. (1983). *Álgebra Universitaria*. México: Edit. CECSA.

Grimaldi, R. P. (2003). *Discrete and Combinatorial Mathematics*. USA: Edit. Addison Wesley.

Cormen, T. H. (2001). *Introduction to Algorithms*. USA: Edit. MIT-Press.

Levitin, A. V. (2002). *Introduction to the Design and Analysis of Algorithms*. USA: Edit. Addison Wesley.

Douglas, B. W. (2000). *Introduction to Graph Theory*. USA: Edit. Prentice Hall.



UNIVERSIDAD DEL VALLE DE MÉXICO
PROGRAMA DE ESTUDIO DE LICENCIATURA
PRAXIS MES XXI

ASIGNATURA: MATEMÁTICAS DISCRETAS Y ALGORÍTMICA

CLAVE: 532515

PERFIL DOCENTE							
NIVEL DE ESCOLARIDAD	PROFESIÓN	EXPERIENCIA PROFESIONAL			EXPERIENCIA DOCENTE		
		ÁREA	ACTIVIDADES	AÑOS	NIVEL EDUCATIVO	ASIGNATURAS	AÑOS Y/O SEMESTRES
Maestría o Doctorado	Matemático o Ciencias de la computación	Programación Investigación	Análisis y diseño de algoritmos	2	Educación Superior (Licenciatura o Posgrado)	Matemáticas Discretas Análisis y diseño de Algoritmos Teoría de la Computación Matemáticas Finitas	2 años

OTROS CONOCIMIENTOS DESEABLES:

Contar con al menos 2 cursos de una Maestría y / o Doctorado en el área de la asignatura.

Conocimiento suficiente de cuando menos una lengua extranjera.

Contar con asistencias recientes a congresos relacionados con el área que le permitan tener una visión actualizada.

Tener en su haber publicaciones recientes con arbitraje en revistas y/o congresos del área.

