



1. Datos Generales de la asignatura

Nombre de la asignatura:	Estructura de datos
Clave de la asignatura:	DAD-2409
SATCA¹:	2-3-5
Carrera:	Ingeniería en Desarrollo de Aplicaciones.

2. Presentación

Caracterización de la asignatura

La asignatura de estructura de datos proporciona al perfil del egresado habilidades para la selección y aplicación de algoritmos y estructuras de datos para el desarrollo e implementación de programas que permitan la solución de problemas de diferentes contextos y estructuras, permite ayudar al alumno a controlar la información y determinar las mejores estructuras a seleccionar para completar los retos a los que este se enfrente, permite conocer conceptos como la complejidad computacional que ayuda al alumno a entender la diferencia de rendimiento entre diferentes algoritmos, entender las diferencias en tiempo de ejecución de los mismos y la diferencia entre las distintas implementaciones internas en memoria de las estructuras.

Previamente, se debe de tener conocimiento del paradigma orientado a objetos y de los fundamentos de programación, además esta asignatura realiza aportaciones a las competencias siguientes del perfil de egreso:

- Analiza, diseña y construye aplicaciones web, móvil o de cómputo en la nube que satisfacen requerimientos de los usuarios, mediante la aplicación de principios, herramientas, prácticas, metodologías y tecnologías emergentes.
- Utiliza modelos de ciencias básicas, ciencias sociales y del área computacional para resolver problemas profesionales de manera ética y sostenible.

Es difícil señalar cuáles son las asignaturas impactadas por medio de estructura de datos porque la naturaleza de los trabajos o proyectos que se efectúen en ellas determinará la necesidad del uso de los conceptos o métodos vistos dentro de esta asignatura.

La asignatura se basa en la utilización de cualquier lenguaje que soporte el paradigma orientado a objetos, donde se realizará una serie de programas explicativos sobre los cuales el alumno aplicará conocimientos para generar soluciones y productos pertinentes.

¹ Sistema de Asignación y Transferencia de Créditos Académicos



Intención didáctica

Esta asignatura está organizada en cinco temas, distinguiéndose en ella, claramente, dos apartados: Primero, la implementación de las estructuras de datos lineales y no lineales a través del manejo de memoria estática y dinámica; segundo, el análisis de los métodos de ordenamiento de datos internos para considerar su eficiencia en la aplicación de soluciones computacionales.

Se inicia el curso con el tratamiento de los tipos de datos abstractos. Para estudiar cada uno de ellos, es necesario aplicar el concepto de modularidad, analizando la forma en que se gestiona la memoria para almacenarlos. Se realiza además un estudio sobre el análisis de la complejidad y eficiencia de los algoritmos, lo cual permitirá determinar cuáles son los más eficientes para solucionar un problema.

El segundo tema aborda la definición, mecanismos y características de la recursividad, aplicando éstos a la creación de procedimientos, así como el análisis de las ventajas y desventajas de estas soluciones recursivas. Los estudiantes identifican dichas características de la recursividad y ejemplifican el caso de las Torres de Hanói, Serie de Fibonacci y Factorial entre otros para comprender mejor el mecanismo recursivo.

El tercer tema trata sobre las estructuras lineales: listas, pilas y colas. La representación de pilas y colas puede darse a través de vectores (memoria estática) o apuntadores y/o referencias (memoria dinámica).

Se analizan también otras variantes como el caso de colas circulares, colas de prioridad, listas simples y doblemente enlazadas. Los estudiantes desarrollan aplicaciones para resolver problemas que requieran de estos tipos de estructuras.

El cuarto tema se refiere a las estructuras no lineales conocidas como árboles y grafos que permiten dar solución a problemas más complejos a través de la recursividad y la utilización de memoria dinámica.

Se analizan los recorridos típicos de árboles binarios, búsquedas, entre otros, así como el algoritmo del viajero para operaciones con grafos.

En el quinto tema, los estudiantes identifican diferentes algoritmos de ordenamiento, los cuales aumentan la dificultad para implementar, los primeros explicados serán simples en su implementación, pero ineficientes, pasando por los algoritmos más simples como son el de burbuja, inserción y selección, pasando a algoritmos complejos de implementar, pero de mejor eficiencia en el tiempo como son: QuickSort, MergeSort y ShellSort.

La intención del sexto tema es que el estudiante conozca otras estrategias para almacenar y recuperar los datos, así como fortalecer la seguridad de la información que se administra, se estudia el sexto tema encargado precisamente de los métodos de recuperación de información.

Al finalizar la asignatura se habrá adquirido las bases para evaluar e implementar soluciones por medio de estructuras.



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 Básicas 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 27 al 31 de mayo de 2024	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
Idea soluciones eficaces de problemas, aplicando conceptos y métodos de estructuras de datos, para el logro del uso eficiente de los recursos computacionales y el tiempo invertidos en ella.

5. Competencias previas

Aplica los conceptos básicos de lógica matemática, relaciones, grafos y árboles para usarlos en modelos que resuelvan problemas computacionales usando un lenguaje de programación orientado a objetos.



6. Temario

No.	Temas	Subtemas
1	Introducción a las estructuras de datos.	1.1. Clasificación de las estructuras de datos. 1.2. Tipos de datos abstractos (TDA). 1.3. Ejemplos de TDA's. 1.4. Manejo de memoria. 1.4.1. Memoria estática. 1.4.2. Memoria dinámica. 1.5. Análisis de algoritmos. 1.5.1. Complejidad en el tiempo. 1.5.2. Complejidad en el espacio. 1.5.3. Eficiencia de los algoritmos.
2	Recursividad.	2.1. Definición. 2.2. Procedimientos recursivos. 2.3. Ejemplos de casos recursivos.
3	Estructuras lineales.	3.1. Listas. 3.1.1. Operaciones básicas. 3.1.2. Tipos de listas: simplemente enlazadas, doblemente enlazadas y circulares. 3.2. Pilas. 3.2.1. Operaciones básicas. 3.2.2. Implementación en memoria estática. 3.2.3. Implementación en memoria dinámica. 3.3. Colas. 3.3.1. Operaciones básicas. 3.3.2. Implementación en memoria estática. 3.3.3. Implementación en memoria dinámica.
4	Estructuras no lineales.	4.1. Árboles binarios. 4.1.1. Operaciones sobre árboles binarios. 4.1.2. Búsquedas sobre árboles binarios. 4.2. Árboles binarios balanceados. 4.2.1. Operaciones sobre ABB. 4.2.2. Búsquedas sobre ABB. 4.3. Grafos. 4.3.1. Operaciones sobre grafos simples. 4.3.2. Implementación de grafos. 4.3.3. Búsquedas en grafos.
5	Métodos de ordenamiento.	5.1. Algoritmos básicos. 5.1.1. Burbuja. 5.1.2. Inserción. 5.1.3. Selección. 5.2. Algoritmos de implementación compleja. 5.2.1. ShellSort. 5.2.2. MergeSort. 5.2.3. QuickSort.

6	Métodos de búsqueda.	6.1. Búsqueda secuencial. 6.2. Búsqueda binaria. 6.3. Búsqueda por funciones de HASH.
---	----------------------	---

7. Actividades de aprendizaje de los temas

1. Introducción a las estructuras de datos	
Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i> Comprende las diferentes estructuras de datos, su clasificación y forma de manipularlas para buscar la manera más eficiente de resolver problemas.</p> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> Resolución de problemas. Trabajo en equipo. Gestión de proyectos. Aprender a aprender. 	<ul style="list-style-type: none"> En equipos de trabajo, realizar las actividades seleccionadas de la siguiente lista: Elaborar un cuadro sinóptico sobre las diferentes estructuras de datos, su representación abstracta y su utilización en un lenguaje de programación orientado a objetos. Resolver el planteamiento de un problema pensado por el catedrático, que aplique más de una estructura de datos en un lenguaje de programación orientado a objetos. Resolver una serie de prácticas en las que se trabajen pequeñas aplicaciones de las diversas estructuras de datos analizadas durante el tema. Asignar un proyecto de dificultad moderada, que para su solución use estructuras de datos.
2. Recursividad	
Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i> Aplica la recursividad en la solución de problemas informáticos simples.</p> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> Resolución de problemas. Trabajo en equipo. Gestión de proyectos. Aprender a aprender. 	<p>En equipos de trabajo, realizar las actividades seleccionadas de la siguiente lista:</p> <ul style="list-style-type: none"> Redactar una definición propia del concepto de recursividad después de consultar en diferentes fuentes bibliográficas. Trasladar un catálogo de problemas iterativos a recursivos, donde distinga el segmento recursivo y la condición de salida, elaborar un reporte de práctica de ejercicios. Desarrollar programas en los que haga uso de la recursividad en programación, además entregue un reporte con los resultados de cada corrida y sus propias conclusiones. Resolver problemas simples en los que se utilice la recursividad para su solución.
3. Estructuras lineales	
Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i> Aplica estructuras de datos lineales para solución de problemas informáticos simples usando un lenguaje de programación orientado a objetos.</p>	<ul style="list-style-type: none"> En equipos de trabajo, realizar las actividades seleccionadas de la siguiente lista: Consultar las características y aplicación de las estructuras lineales en diversas fuentes bibliográficas y realizar un informe.

<p>Genérica(s):</p> <ul style="list-style-type: none"> Resolución de problemas. Trabajo en equipo. Gestión de proyectos. Aprender a aprender. 	<ul style="list-style-type: none"> Programar las operaciones básicas de pilas y colas de manera estática (usando arreglos) entregando los códigos correspondientes junto a sus corridas. Utilizar las clases predefinidas para el manejo de pilas, colas y listas enlazadas(dinámicas) y describir en un texto la diferencia de hacerlo con arreglos. Emplear las estructuras lineales en la elaboración de códigos para la resolución de problemas elaborando un reporte. Implementar desde cero sus propias clases para estructuras de datos dinámicas, entregando un reporte con sus implementaciones y corridas respectivas.
4. Estructuras no lineales	
Competencias	Actividades de aprendizaje
<p>Específica(s): Aplica estructuras no lineales en la solución de problemas concretos.</p> <p>Genérica(s):</p> <ul style="list-style-type: none"> Resolución de problemas. Trabajo en equipo. Realiza su trabajo de manera autónoma y autorregulada. Introduce recursos y experiencias que promueven un pensamiento crítico. 	<p>En equipos de trabajo, realizar las actividades seleccionadas de la siguiente lista:</p> <ul style="list-style-type: none"> Elaborar un cuadro sinóptico o esquema con la clasificación de los árboles, los grafos y sus aplicaciones. Implementar las operaciones básicas de inserción, eliminación, búsqueda y recorridos en un árbol binario. Desarrollar aplicaciones con grafos y elaborar un reporte de prácticas con los códigos elaborados y las corridas correspondientes.
5. Métodos de ordenamiento	
Competencias	Actividades de aprendizaje
<p>Específica(s): Idea algoritmos de ordenamiento específicos, a partir de los que se estudiaron en el tema o que haya estudiado, para su inclusión en aplicaciones que requieran su uso en condiciones especiales.</p> <p>Genérica(s):</p> <ul style="list-style-type: none"> Resolución de problemas. Trabajo en equipo. Realiza su trabajo de manera autónoma y autorregulada. Introduce recursos y experiencias que promueven un pensamiento crítico. 	<p>En equipos de trabajo, realizar las actividades seleccionadas de la siguiente lista:</p> <ul style="list-style-type: none"> Consultar en fuentes bibliográficas los diversos algoritmos de ordenamiento y generar un cuadro comparativo mostrando su notación BigO. Resolver problemas donde se justifique la necesidad de utilizar métodos de ordenamiento. Realizar ejercicios donde se elaboren programas de los diferentes métodos de ordenamiento y entregar el código respectivo junto a las corridas en un reporte.

6. Métodos de Búsqueda	
Competencias	Actividades de aprendizaje
<p><i>Específica(s):</i> Idea algoritmos de búsqueda específicos, a partir de los que se estudiaron en el tema, para su inclusión en aplicaciones que requieran su uso en condiciones especiales.</p> <p><i>Genérica(s):</i></p> <ul style="list-style-type: none"> Resolución de problemas. Trabajo en equipo. Realiza su trabajo de manera autónoma y autorregulada. Introduce recursos y experiencias que promueven un pensamiento crítico. 	<p>En equipos de trabajo, realizar las actividades seleccionadas de la siguiente lista:</p> <ul style="list-style-type: none"> Consultar en las fuentes bibliográficas los diversos algoritmos de búsqueda. Práctica de ejercicios. Implementar en un lenguaje de programación los métodos de búsqueda para un conjunto de datos generados aleatoriamente. Realizar ejercicios donde se elaboren programas de los diferentes métodos de búsqueda y entregar el código respectivo junto a las corridas en un reporte.

8. Práctica(s)

- Desarrollar una aplicación donde se utilicen tipos de datos abstractos para comprobar el comportamiento estático y dinámico, se sugiere un sistema de almacenamiento de productos realizado con archivos.
- Desarrollar una aplicación que resuelva mediante la técnica de recursividad un problema específico como puede ser el triángulo de Pascal recursivo, las torres de Hanói, el algoritmo de MCD.
- Hacer una aplicación que simule una lista de espera para la asignación de mesas en un restaurante.
- Crear una aplicación que simule un sistema de comandas para un restaurante.
- Desarrollar una aplicación que simule la salida de aviones en un aeropuerto usando colas de prioridad.
- Llevar a cabo una aplicación empleando los conceptos de árboles binarios.
- Implementar una búsqueda genérica en grafos que seleccione un camino entre diferentes nodos representando los estados de la república o municipios de su estado.
- Implementar una búsqueda óptima en grafos pesados que seleccione el mejor camino entre diferentes nodos representando los estados de la república o municipios de su estado.
- Aplicar los métodos de búsqueda a un conjunto de datos aleatorios y determinar su eficiencia.
- Implementar un diccionario simple de inglés a español con tablas hash.
- Realizar una aplicación de búsqueda de estudiantes implementando la búsqueda binaria.



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, esta 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

Las técnicas, herramientas y/o instrumentos sugeridos que permiten obtener el producto del desarrollo las actividades de aprendizaje: códigos, exposición, mapas mentales o conceptuales, resúmenes, reportes de prácticas, cuadros comparativos e informes.

Las técnicas, herramientas y/o instrumentos recomendados, que, como docente, me permite constatar el logro o desempeño de las competencias del estudiante: rúbricas, listas de cotejo, listas de verificación y guías de observación.



11. Fuentes de información

1. Campesato, O. (2023). Data Structures in Java. Mercury Learning and Information.
2. Dale, N., Joyce, D. T., & Weems, C. (2016). Object-Oriented Data Structures Using Java 4th Edición. Jones & Bartlett Learning.
3. Drozdek, A. (2007) Estructuras de datos y algoritmos con Java. México: Cengage Learning Editores. Educación.
4. Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). Data Structures and Algorithms in Java (6a ed.). John Wiley & Sons.
5. Guardati, S. (2007) Estructura de Datos Orientada a Objetos Algoritmos con C++, Primera Edición. México: Prentice Hall.
6. Hu, Y. (2019). Easy Learning Data Structures & Algorithms Java Practice: Graphically learn data structures and algorithms better than before (Easy Learning Programming for beginner Book 3). Independently published.
7. Hu, Y. (2020). Fácil Aprendizaje Estructuras De Datos y Algoritmos Java: Aprenda fácilmente estructuras de datos y algoritmos gráficamente Pasta blanda. Independently Published.
8. Jain, H. (2022). Problem Solving in Data Structures & Algorithms Using Java. Hemant Jain.
9. Martínez, R. y Quiroga, E. (2004) Estructura de Datos Referencia práctica con orientación a objetos. México: Ed. Thomson.
10. Myers, D. S. (2024). Data Structures and Algorithms in Java: A Project-Based Approach. Cambridge University Press.
11. Yang Hu(2022). Fácil Aprendizaje Estructuras De Datos y Algoritmos Javascript:Independently published.