|  |
| --- |
| **Objetivo General:**  Al final del curso el alumno será capaz de diseñar y aplicar estructuras de datos lineales y recursividad para la solución de problemas computacionales.  **Objetivos específicos:**   1. Diseñar y aplicar listas, pilas y colas para la solución de problemas computacionales. 2. Comprender y programar las representaciones más comunes de estas estructuras de datos, es decir, usando memoria estática y dinámica. 3. Programar, a partir de su comprensión, los principales algoritmos de búsqueda y ordenamiento de datos. 4. Aplicar el concepto de recursividad en la creación de algoritmos.   **CONTENIDO SINTÉTICO.**   1. Administración de memoria.    1. Memoria estática vs. Memoria dinámica.    2. Alojamiento de memoria.    3. Liberación de memoria. 2. Listas.    1. Definición de lista.    2. Operaciones sobre listas.    3. Implementación estática.    4. Listas ligadas sencillas.    5. Listas doblemente ligadas.    6. Listas ligadas circulares. 3. Pilas.    1. Definición de Pila.    2. Operaciones sobre pilas: Push y Pop.    3. Implementación utilizando listas.    4. Implementación estática y dinámica. 4. Colas.    1. Definición de Cola.    2. Operaciones sobre colas: Enqueue y Dequeue.    3. Implementación utilizando listas.    4. Implementación estática y dinámica. 5. Algoritmos iterativos de búsqueda y ordenamiento.    1. Búsqueda secuencial y binaria.    2. Ordenamiento por: burbuja, inserción, selección. 6. Recursividad.    1. Concepto de recursividad y su aplicación en la programación.    2. Aplicación de la recursividad en métodos de búsqueda y ordenamiento: búsqueda binaria, mergesort, quicksort.   **MODALIDADES DE CONDUCCIÓN DEL PROCESO DE ENSEÑANZA-APRENDIZAJE.**  Clase teórico-práctica a cargo del profesor con participación activa del alumno y clase práctica en laboratorio.  El profesor diseñará experiencias de aprendizaje por problemas, con nivel de complejidad incremental, tanto en el aula como en el laboratorio. El alumno analizará los problemas planteados y aplicará los conceptos aprendidos durante el curso para la solución de los problemas tanto en clase como extra clase y las prácticas de laboratorio.  Las habilidades transversales que deberá adquirir el alumno, asociadas a esta UEA son las siguientes:  (Ht0) Lenguaje disciplinar: Profundizará en el uso de algún lenguaje de programación.  (Ht1) Autoaprendizaje: El profesor expondrá al alumno a problemas que impliquen un reto en el que, para resolverlos, debe integrar los conceptos del curso con conceptos aprendidos en la UEA Programación estructurada.  (Ht3) Comunicarse eficazmente de forma oral y escrita en español: El alumno deberá responder oralmente a preguntas de los temas vistos en clase y ser capaz de explicar conceptos de forma escrita en (una cuartilla) como parte importante de las prácticas de laboratorio.  (Ht4) Comprender perfectamente los textos técnicos en español: El alumno investigará fuera de clase un tema relacionado con el curso y lo explicará a sus compañeros. La explicación deberá ser breve (no más de 2 min.).  (Ht6) Comprender textos técnicos en inglés: Leerá un texto técnico en inglés y lo explicará en español. El texto deberá ser breve (aprox. una cuartilla).  Las habilidades disciplinares que deberá adquirir el alumno asociadas a esta UEA son:  (H1) Abstracción de sistemas.  (H2) Resolver problemas mediante algoritmos: El alumno analizará los problemas propuestos que, además de las estructuras básicas de control, involucran tanto estructuras de datos lineales como recursividad y los resolverá mediante algoritmos que implementará en algún lenguaje de programación.  Las actitudes que deberá mostrar el alumno son:  (A2) Perseverancia para resolver problemas mediante estructuras de datos lineales y recursividad.  (A3) Disciplina para aplicar los conocimientos adquiridos.  **MODALIDADES DE EVALUACIÓN.**  **Evaluación Global:**  Se ponderarán las siguientes actividades a criterio del profesor:   * Tareas individuales, que incluyen prácticas de laboratorio. * Evaluaciones periódicas. * Participación en el proceso de resolución de problemas, tanto en las sesiones teóricas como prácticas. * Evaluación terminal. * Exposición del tema que el alumno investigó en textos que están en español. * Exposición del contenido de un texto técnico en inglés (una cuartilla). * Solución de los retos presentados por el profesor en clase.   **Evaluación de Recuperación:**  El alumno deberá presentar una evaluación que contemple todos los contenidos de la unidad de enseñanza aprendizaje. A criterio del profesor, también se podrá solicitar un proyecto de cómputo.  Requiere inscripción previa a la UEA.  **BIBLIOGRAFÍA NECESARIA O RECOMENDABLE.**   1. Aho, A. V. Hopcroft, J. E. y Ullman, J. D. *Estructura de Datos y Algoritmos.* Addison-Wesley, México1998*.* 2. Watt, D. y Brown, D. *Java Collections, An introduction to abstract data types, data structures and algorithms*, John Wiley & Sons, New York, 2004. 3. Weiss, M. A. *Data structures and algorithm analysis in Java.* Addison-Wesley, Massachusetts, 2012. 4. Weiss, M. A. *Data structures and problem solving using C++.* Addison-Wesley, Massachusetts, 2000. 5. Weiss, M. A. *Data structures and problem solving using Java.* Addison-Wesley, Massachusetts, 2006. |