

Programa Regular

Complejidad Temporal, Estructuras de Datos y Algoritmos.

Modalidad de la Asignatura: Teórico-práctica.

Carga horaria: 4 hs.

Objetivos:

Los estudiantes lograrán diseñar y desarrollar programas correctos y eficientes. Se propone:

Proporcionar al estudiante más experiencia en el campo de la programación mediante la realización de prácticas.

Conocer mejor cómo es un lenguaje de programación, en particular un lenguaje orientado a objetos. Con este objetivo se estudiarán aspectos como son la estructura de control, el tipo de datos, la gestión de memoria y los mecanismos de abstracción de un lenguaje de estas características.

Conocer nuevas técnicas de programación. En particular, el uso de la memoria dinámica y las estructuras de datos enlazadas, que están en la base de muchas aplicaciones.

Introducir herramientas de diseño de algoritmos y la ingeniería algorítmica como selección de las estructuras de datos y de las técnicas algorítmicas más adecuadas para la resolución de un problema concreto.

Contenidos: Presentación del concepto de complejidad temporal. Expresar un algoritmo en término de $T(n)$. Cálculo del $T(n)$ para algoritmos iterativos y recursivos. Uso de tipos genéricos de datos para la resolución de problemas. Abstracción de datos mediante la especificación de interfaces comunes, herencia e implementaciones intercambiables entre sí. Presentación de las estructuras de datos: listas, pilas, colas, colas de prioridad, árboles y grafos. Implementaciones de las estructuras de datos: listas, colas, pilas, árboles y grafos. Algoritmos fundamentales: recorridos en arboles y grafos, caminos mínimos en grafos, árboles de expansión.

Unidades temáticas:

Unidad I

Análisis de Algoritmos: Eficiencia, introducción al concepto $T(n)$, casos, notación asintótica Big-Oh, regla de la suma y del producto, cálculo del $T(n)$ para algoritmos iterativos y recursivos.

Unidad II

Encapsulamiento y Abstracción: Introducción al concepto de tipo de datos genérico, especificación de las interfaces de listas, pilas y colas, implementaciones intercambiables para listas, pilas y colas.

Unidad III

Estructuras Arbóreas: Definición, descripción, terminología, representaciones, recorridos y tiempo de ejecución de los mismos para Árboles Generales, Árboles Binarios, Árboles Binarios de Búsqueda, Árboles AVL. Implementación de colas de prioridades mediante Heap.

Unidad IV

Grafos: Terminología, representación con lista y matriz de adyacencias, recorridos DFS y BFS, sort topológico, caminos de costo mínimo y árbol de expansión mínimo.

Bibliografía:

- Allen Weiss, M. Estructuras de datos en Java. Editorial Addison Wesley (ISBN 9788478290352). Año 2000.
- Aho, A.; Hopcroft, J.; Ullman, J. Estructuras de datos y algoritmos. Editorial Addison Wesley (ISBN 9789684443457). Año 1999.
- Waite, M.; Lafore, R. Data structures and algorithms in Java. Editorial Kindle (ISBN 978-0672324536). Ed. 2°. Año 2002.

Propuesta didáctica: Las clases se desarrollarán en el Laboratorio de Informática. Se organizarán en modalidades teórico-prácticas con soporte de presentaciones digitales y prácticas en función de cada clase.

En las clases se presentan los contenidos teóricos y se van resolviendo en forma conjunta ejemplos que ayuden a comprender los nuevos conceptos introducidos.

La formación práctica está basada en la resolución de problemas tipo y de problemas abiertos de ingeniería, cuyas resoluciones se realizan principalmente en las computadoras, utilizando aplicaciones de uso en la industria que permitan un contacto directo con las tecnologías actuales.

En cuanto a los problemas abiertos de ingeniería, se realizarán trabajos relacionados con la implementación de una aplicación determinada, que haga uso

de las herramientas de diseño de algoritmos que provee la programación orientada a objetos. Cada trabajo debe incluir un detalle de los problemas encontrados, las formas de solucionarlos, las condiciones de ejecución, formato de los datos de entrada e ideas o sugerencias para realizar una versión mejorada del mismo. La realización de los trabajos implica la aplicación de los conocimientos adquiridos hasta entonces, lo cual constituye la base formativa para que el estudiante adquiera las habilidades que le permitan encarar proyectos y diseños de ingeniería.

Los estudiantes deben realizar entregas de al menos un trabajo integrador durante el desarrollo de la cursada el cual será revisado por los docentes y devuelto a los estudiantes para su corrección.

El material correspondiente a las clases teóricas, así como los documentos de la práctica se encontrarán disponibles a través de un grupo Web al cual los estudiantes tendrán acceso. Este mecanismo también será utilizado para realizar consultas simples.

Actividades extra-áulicas: Se establecerán guías de actividades prácticas para que el estudiante pueda ejercitar, a fin de consolidar los conceptos aprendidos en clase.

Evaluación: La evaluación integradora de las instancias teórico-prácticas se realiza a través de un parcial teórico-práctico en máquina. Además, los estudiantes deberán desarrollar un trabajo final donde se integren los temas vistos en la materia. Las clases son obligatorias ya que implican participación y debate que forman parte de la evaluación.