



Análisis de Algoritmos y Estructuras de Datos

1. Características generales

Nombre:	Análisis de Algoritmos y Estructuras de Datos
Sigla:	CI-0116
Créditos:	4
Horas lectivas:	5
Requisitos:	CI-0113 Programación II, CI-0111 Estructuras Discretas
Correquisitos:	ninguno
Clasificación:	Curso propio
Ciclo de carrera:	II ciclo, II año
Docente(s):	Allan Berrocal Rojas
Datos de contacto:	correo: allan.berrocal@ucr.ac.cr oficina: 3-28
Grupo:	01 y 03
Semestre y año:	I ciclo 2025
Horario y lugar de clases:	Grupo 01 L 7-9:50, J 7-8:50, aula 203 Grupo 03 L 13-15:50, J 13-14:50, aula 305
Horario y lugar de consulta:	L 10:00-11:30, oficina 3-28 J 9:00-11:30, oficina 3-28
Asistente:	Grupo 01: Pendiente Grupo 03: Pendiente
Modalidad:	presencial

2. Descripción

Existen diversas formas de solucionar computacionalmente un problema, lo que conduce al dilema de escoger la más adecuada. Hay diferentes aspectos que se pueden evaluar para hacer tal escogencia. En este curso se enseña a calcular la complejidad espacio-temporal de una solución, con el fin de resolver un problema eficientemente. También se enseñan modelos matemáticos, tipos de datos abstractos, estructuras de datos y





algoritmos clásicos, haciendo especial énfasis en la complejidad espacio-temporal de las implementaciones.

3. Objetivos

Objetivos general

El objetivo general del curso es que el estudiante aprenda familias de algoritmos, tipos de datos abstractos y estructuras de datos asociados a modelos matemáticos, y analice su complejidad espacio-temporal para compararlos en términos de su eficiencia.

Objetivos específicos

Durante este curso el estudiante desarrollará habilidades para:

1. Usar modelos matemáticos, tipos de datos abstractos, estructuras de datos y algoritmos clásicos para resolver problemas.
2. Demostrar la correctitud de un algoritmo iterativo, aplicando las técnicas correspondientes.
3. Analizar el comportamiento asintótico del tiempo de ejecución y del espacio requerido por un algoritmo o una estructura de datos para determinar su eficiencia.
4. Diseñar e implementar estructuras de datos y algoritmos eficientes, para hacer un uso apropiado de los recursos computacionales.
5. Diseñar e implementar algoritmos eficientes para resolver problemas de optimización.

4. Contenidos

Los ejes temáticos del curso y los objetivos a los que contribuyen se muestran en la tabla que sigue.

Objetivos	Eje temático	Desglose
1,4	Modelos matemáticos, tipos de datos abstractos y estructuras de datos	Definiciones, propiedades y diferencias entre ellos.
2	Demostración de la correctitud de algoritmos iterativos	Invariante de un ciclo. Pasos de inicialización, mantenimiento y terminación.





3	Complejidad espacio-temporal de un algoritmo.	Cálculo del orden de duración de algoritmos recursivos, iterativos y con llamados a métodos o funciones, en el peor caso, caso promedio y mejor caso. Resolución de relaciones de recurrencia.
1,3,4	Modelos y tipos de datos abstractos	Listas, pilas, colas, árboles, conjuntos, diccionarios, colas de prioridad, relaciones y grafos. Operadores básicos, implementaciones, algoritmos asociados, aplicaciones, complejidad espacio-temporal de los algoritmos y sus implementaciones.
1,3,4	Estructuras de datos	Arreglos, listas enlazadas, árboles parcialmente ordenados, árboles de búsqueda binaria, árboles balanceados, tablas de dispersión. Estructuras de datos para la implementación de árboles, grafos y relaciones. Complejidad espacio-temporal.
1,3,4	Algoritmos de ordenamiento	Ordenamiento de burbuja, ordenamiento por selección, inserción, mezcla, montículos, residuos y ordenamiento rápido (quicksort), y su complejidad espacio-temporal.
1,3,5	Técnicas de resolución de problemas	Divide y vencerás, búsqueda exhaustiva, programación dinámica, algoritmos ávidos, y su complejidad temporal.
1,3,4	Algoritmos para grafos	Recorridos en anchura y en profundidad, árboles recubridores mínimos, rutas más cortas, y su complejidad espacio-temporal.

5. Metodología

Este curso utiliza una metodología híbrida que combina clases magistrales con lecturas extra-clase, discusiones y ejercicios prácticos que apoyan el proceso de aprendizaje. La parte teórica será guiada por la persona docente mediante presentaciones temáticas y actividades en clase. Por tanto se requiere de una participación activa de las personas estudiantes durante las clases.

Los contenidos vistos en clase y las lecturas asignadas descritas en la bibliografía serán evaluadas por medio de exámenes cortos (que pueden ser o no anunciados con anticipación), prácticas dentro y fuera de horario del curso, laboratorios y exámenes.

Se utilizará la plataforma institucional [Mediación Virtual](#) (MV) como aula virtual para efectos de comunicar y distribuir diversos materiales y recursos tales como





lecturas, asignaciones, prácticas, quices y exámenes, entre otros. Es responsabilidad de las personas estudiantes registrarse en dicha plataforma, así como revisar las noticias y actividades publicadas ahí. El uso de este entorno virtual complementa las actividades sincrónicas y las instrucciones emitidas por la persona docente durante las horas lectivas. El correo electrónico institucional y el entorno de MV serán medios de comunicación oficiales entre la persona docente y las personas estudiantes

6. Evaluación

Valor	Rubro	Descripción
45%	3 exámenes	Serán tres exámenes de 15% cada uno. Las fechas exactas se comunicarán con antelación durante el semestre.
40%	Tareas programadas	Serán al menos tres tareas programadas con entregas parciales donde las personas estudiantes resuelven problemas en relación con los algoritmos y estructuras de datos estudiados en el curso. Estas tareas requieren la aplicación de buenas prácticas de programación, así como la capacidad de análisis y síntesis sobre el desempeño obtenido al ejecutar los programas.
10%	Quices	Se realizan durante las lecciones y en cualquier momento durante el transcurso de la lección. Solo se repondrán en los casos que establece el Reglamento de Régimen Académico Estudiantil en su Artículo 24.
5%	Tareas cortas	Se realizan varios trabajos durante el semestre. La descripción, objetivos y criterios de evaluación serán anunciados en Mediación Virtual

Lineamientos:

1. Toda asignación se considera tardía si se entrega después de las 6:00 a.m. posterior a la fecha de entrega. Si la entrega tiene 24 o menos horas de retraso se le aplicará una penalización de 50% del valor. Después de 24 horas de retraso, la asignación será calificada con cero.
2. Todas las evaluaciones deben ser entregadas tal y como se indica en el respectivo enunciado.
3. Solo se repondrán evaluaciones bajo el procedimiento descrito en el artículo 24 de Reglamento de Régimen Académico Estudiantil de la UCR.
4. Todas las evaluaciones son estrictamente individuales excepto aquellas en cuyo enunciado se especifique algo distinto.
5. En todos los trabajos y las evaluaciones, se calificará la redacción, ortografía, estructura y contenido.
6. Todo código fuente que no compile será calificado con cero.
7. En toda asignación se evaluarán las buenas prácticas de programación, como identificadores significativos, indentación del código fuente, apego





a una convención de estilo, documentación de interfaces e implementaciones de subrutinas, y reutilización de código. Serán castigadas malas prácticas de programación como redundancia de código, fugas y accesos inválidos de memoria, condiciones de carrera, espera activa, y otras practicas contrarias a lo estudiado en el curso.

8. A discreción de la persona docente, en algunas evaluaciones se podrían ofrecer rubros opcionales por crédito extra.
9. Es ilegal presentar como propio, material parcial o totalmente creado por otras personas u obtenido de fuentes de información, como por ejemplo de libros o de Internet, sin la debida referencia a la fuente de producción intelectual.
10. Asimismo, es prohibido suplantar, parcial o totalmente, una respuesta propia, que por su naturaleza debe ser original, por una elaborada por otra persona o generada por herramientas automatizadas de generación de respuestas.
11. No se permite entregar como trabajo propio ninguna solución (código fuente, trabajo escrito, etc.) que haya sido generada total o parcialmente por un modelo de inteligencia artificial incluso si la persona estudiante declara que lo ha utilizado. Si se detecta el uso de tales herramientas se procede como se indica en el siguiente punto.
12. En cualquier asignación en que se sospeche de plagio (incluso si es parcial), o que se pueda constatar que fue generada total o parcialmente por un modelo de inteligencia artificial será calificada con una nota de cero. Naturalmente, se aplicará el debido proceso estipulado en el Reglamento de Orden y Disciplina de los Estudiantes de la Universidad de Costa Rica ([consultar](#)).

7. Cronograma

Semana	Fecha	Actividad
1	10 de marzo	Inicio de lecciones
6	14 de abril	Semana Santa
7	21 de abril	Semana Universitaria
8	28 de abril	Primer examen
13	2 de junio	Segundo examen
17	30 de junio	Tercer examen
17	3 de julio	Fin de lecciones
18-19	7-17 de julio	Pruebas de ampliación

8. Bibliografía

1. Cormen, T. H., Leiserson, C. E., Rivest, R. L. y Stein, C. Introduction to Algorithms, IV ed. MIT Press, 2022.
2. Aho, A. V., Ullman, J. D. y Hopcroft, J. E. Data Structures and Algorithms. Addison-Wesley, 1983.





3. Brassard, G. y Bratley, P. Fundamentos de Algoritmos. Prentice Hall. 1997.
4. Heileman, G. Estructuras de Datos, Algoritmos y Programación Orientada a Objetos. McGraw Hill. 1998.
5. Sedgewick, R. Algoritmos en C++. Pearson Education. 1995.

9. Recursos estudiantiles

Para información sobre recursos estudiantiles disponibles en la UCR, incluyendo el Sistema de Bibliotecas y la normativa universitaria vigente, favor visitar la página:

<https://www.ecci.ucr.ac.cr/vida-estudiantil/servicios-institucionales-para-estudiantes/guia-de-recursos-estudiantiles-de-la-ucr>

