

Estructuras de Datos — Programa y cronograma 2026-2

Programa y cronograma del curso para el calendario académico 2026-2 de la Facultad de Ingeniería y Ciencias, con los horarios de la sección B.

Identificación

Campo	Valor
Nombre	Estructuras de Datos
Código	300CIP009
Sección	B
Semestre	2026-2
Docente sección B	Carlos Andrés Delgado Saavedra
Horario	Miércoles 17:00–19:00 y viernes 14:00–17:00
Salón	PL-3.1
Periodo	27 de julio al 29 de noviembre de 2026
Lenguajes	C (ANSI) y C++

Textos guía

1. R. Thareja, *Data Structures Using C*, 2018.
2. N. Kalicharan, *Data Structures in C*, 2008 (1.^a ed.).
3. R. Sebesta, *Concepts of Programming Languages*, 2015.
4. T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, *Introduction to Algorithms*, 4.^a ed. (CLRS).

Descripción

El curso introduce al estudiante al modelamiento e implementación de problemas de ingeniería usando estrategias de abstracción de datos. Se enfoca en la descomposición de un programa en el componente de datos y el componente algorítmico, presenta el costo y los beneficios de estructuras dinámicas y estáticas, y cierra con la escogencia de la estructura apropiada para modelar problemas actuales. En la asignatura: (i) se apropian y aplican conocimientos de ciencias de la computación; (ii) se implementan algoritmos correctos y eficientes; (iii) se analizan algoritmos a través de demostraciones y de la interpretación de resultados experimentales; y (iv) se justifican decisiones de diseño e implementación.

Objetivos de aprendizaje

- **OA1** Apropiar conocimientos de computación a través de la participación activa en el diseño e implementación de estructuras de datos simples para el modelamiento de problemas de ingeniería (resultados 1.1, 1.3, 1.4).
- **OA2** Diseñar e implementar soluciones a problemas de ingeniería mediante el uso de herramientas, considerando impactos sociales y medioambientales (2.4, 2.6, 2.7).
- **OA3** Describir ideas con argumentos precisos y basados en evidencia (3.2, 3.4 — material técnico en inglés).
- **OA5** Liderar y participar en equipos disciplinares en proyectos de pequeña escala (5.1, 5.2).

Estrategias didácticas

Asignatura orientada por problemas algorítmicos de la literatura y de arenas de programación públicas. Clase magistral, aprendizaje basado en problemas y aprendizaje basado en simulaciones (servidor de juzgamiento automático con retroalimentación). Los problemas se formulan mayormente en inglés; eventualmente hay comprobaciones de lectura.

Evaluación

	Parcial 1	Parcial 2	Proyecto	Tareas y otros
Porcentaje	30 %	30 %	30 %	10 %

- El proyecto es individual y tiene dos entregas: un avance funcional (10 %) y la entrega final con sustentación (20 %).
- *Tareas y otros*: el promedio de las cinco tareas vale 8 %; los quices (comprobaciones de lectura), los ejercicios en clase y la participación completan el 2 %.
- Las tareas son individuales, con problemas conceptuales y de implementación.
- Los parciales se hacen en la sesión del viernes.

Calendario académico 2026-2 (FIC)

Hito	Fecha
Inicio de clases	Lunes 27 de julio (semana 1)
Última fecha de cancelación	Sábado 7 de noviembre (semana 14)
Fin del período académico	Sábado 28 de noviembre (semana 17)
Notas definitivas en Secretaría	Miércoles 2 de diciembre

Festivo que afecta el curso: viernes 7 de agosto (Batalla de Boyacá).

Cronograma de sesiones

Miércoles de 2 horas (17:00–19:00) y viernes de 3 horas (14:00–17:00).

#	Fecha	Día	Tema	Entregas
1	29/07	Mi	Presentación del curso; repaso de programación básica	
2	31/07	Vi	Lenguajes de programación; proceso de compilación; paradigmas	
3	05/08	Mi	Introducción a la complejidad computacional	
—	07/08	Vi	Festivo (Batalla de Boyacá)	
—	12/08	Mi	Sin clase	
—	14/08	Vi	Sin clase	
4	19/08	Mi	Análisis de complejidad computacional	
5	21/08	Vi	Análisis de complejidad computacional: ejercicios	
6	26/08	Mi	C/C++: manejo de memoria	
7	28/08	Vi	C/C++: punteros y arreglos; arreglos bidimensionales	
8	02/09	Mi	C/C++: paso de parámetros por referencia	

#	Fecha	Día	Tema	Entregas
9	04/09	Vi	Introducción a la abstracción de datos; tipos abstractos de datos	Tarea 1 (do 06/09)
10	09/09	Mi	Repaso POO	
11	11/09	Vi	TAD Lista, Pila y Cola: contrato y ejemplos de uso	
12	16/09	Mi	Ejercicios con los TAD Lista, Pila y Cola	Tarea 2 (do 13/09)
—	18/09	Vi	PARCIAL 1	
13	23/09	Mi	STL de C++	
14	25/09	Vi	STL de C++: iteradores, algoritmos y ejercicios	Tarea 3 (do 04/10)
15	30/09	Mi	Implementaciones del TAD Lista: estáticas	
16	02/10	Vi	Implementaciones del TAD Lista: enlazadas	
17	07/10	Mi	Implementaciones del TAD Pila	Tarea 4 (ma 13/10)
18	09/10	Vi	Implementaciones del TAD Cola	
19	14/10	Mi	Implementaciones: ejercicios; repaso para el parcial 2	
—	16/10	Vi	PARCIAL 2	
20	21/10	Mi	Contenedores <code>list</code> , <code>deque</code> , <code>stack</code> , <code>queue</code>	
21	23/10	Vi	Contenedores (continuación); TAD tabla de direccionamiento directo	
22	28/10	Mi	Tablas de direccionamiento directo y tablas hash	Entrega 1 del proyecto (vi 30/10); Tarea 5 (do 01/11)
23	30/10	Vi	Tablas hash; contenedor <code>map</code>	
24	04/11	Mi	Contenedor <code>map</code> : cierre y ejercicios	
25	06/11	Vi	Introducción a árboles (TAD árbol)	
26	11/11	Mi	Montículos (<i>heaps</i>)	
27	13/11	Vi	TAD cola de prioridad; <code>priority_queue</code> ; contenedor <code>set</code> ; estructuras de datos en Python	
28	18/11	Mi	Sustentaciones del proyecto	Entrega final del proyecto (lu 16/11)
29	20/11	Vi	Sustentaciones del proyecto (continuación)	

Notas del cronograma:

- Las fechas de los exámenes y del proyecto pueden ajustarse según los eventos y disposiciones de la Universidad.
- Cinco tareas y un proyecto con dos entregas: un avance funcional y la entrega final con sustentación.