

Planificación de
Estructuras de Datos y Algoritmos II



Código/s: R-312

Identificación y características del Espacio Curricular

Carrera/s:	Licenciatura en Ciencias de la Computación		
Plan de Estudios:	2010, TO2024	Carácter:	Obligatoria
Bloque/Campo:		Área:	Algoritmos y Lenguajes
Régimen de cursado:	Cuatrimestral		
Cuatrimestre:	5º [LCC], 5º [LCC]		
Carga horaria:	120 hs. / 8 hs. semanales	Formato curricular:	Asignatura
Escuela:	Ciencias Exactas y Naturales	Departamento:	Ciencias de la Computación
Docente responsable: JASKELIOFF, Mauro			

Programa Sintético

Introducción, diseño de algoritmos, complejidad, ordenación, estructuras de datos elementales, tablas hash (diccionarios), árboles binarios de búsqueda, árboles rojo-negro. Programación dinámica, algoritmos golosos (greedy), b-trees heaps (binomial - fibonacci), conjuntos disjuntos, algoritmos de grafos.

Espacios Curriculares Relacionados

Previos Aprobados: R-212 - Estructuras de Datos y Algoritmos I, R-221 - Complementos de Matemática I

Simultaneos Recomendados:

Posteriores: R-322 - Análisis de Lenguajes de Programación, R-324 - Teoría de Base de Datos, R-411 - Ingeniería del Software I, R-413 - Introducción a la Inteligencia Artificial, R-422 - Compiladores

Vigencia desde 2024

Firma Profesor

Fecha

Firma Aprob. Escuela

Fecha

Con el aval del Consejo Asesor:

Fundamentación

Los algoritmos y estructuras de datos constituyen la esencia de la disciplina vinculada con la carrera. Es por ello que materias que aborden esta temática son fundamentales. En particular en esta materia se estudiarán las estructuras de datos persistentes, a diferencia de Estructuras de Datos y Algoritmos I, que se focaliza en estructuras efímeras. Las estructuras de datos persistentes han cobrado una enorme importancia debido a que son más fáciles de verificar y programar correctamente, incluso en presencia de programas paralelos. Las estructuras persistentes ocurren naturalmente en un lenguaje funcional puro, por lo que se optó por enseñarlas usando este estilo de programación.

Dentro de la carrera los aportes concretos de la asignatura son:

- Enseñanza de programación funcional y de los beneficios de la abstracción.
- Enseñanza de diseño, implementación, verificación, y evaluación de algoritmos.

La materia se encuentra en el primer cuatrimestre de tercer año.

Resultados del aprendizaje

Al finalizar el cursado los/las estudiantes serán capaces de:

RA1 Utilizar y diseñar estructuras de datos funcionales y tipos abstractos de datos.
RA2 Resolver problemas básicos de programación funcional.
RA3 Aplicar técnicas de verificación de programas funcionales.
RA4 Entender la noción de complejidad de algoritmos para programas secuenciales y paralelos y aplicarla a problemas puntuales.
RA5 Identificar las metas y responsabilidades individuales y colectivas y actuar de acuerdo a ellas.
RA6 Comunicar en forma clara y usando lenguaje preciso y técnico los conceptos fundamentales de las estructuras de datos y la programación funcional.

Competencias / Ejes transversales y Resultados del Aprendizaje

Competencia/Eje transversal al que tributa	Nivel	Resultados del Aprendizaje
CGT1-Identificación, formulación y resolución de problemas de informática	Medio	RA1 RA2
CGT4-Utilización de técnicas y herramientas de aplicación en la informática	Medio	RA3 RA4
CGS1-Fundamentos para el desempeño en equipos de trabajo	Bajo	RA5
CGS2-Fundamentos para la comunicación efectiva	Bajo	RA6

Programa Analítico

Unidad 1: Complejidad

1.1 Orden de una sucesión. Notación $O()$. Definición. Propiedades.

1.2 Modelos de Costos. Aplicación al análisis de algoritmos. Ejemplos.

1.3 Técnicas de resolución y cálculo de orden en recurrencias. Método de sustitución. Árboles de Recurrencia. Regla de suavidad. Teorema Maestro.

2. Estructuras de Datos Persistentes

- 2.1 Introducción a la programación funcional. Sintaxis. Técnicas de programación funcional.
- 2.2 Programación con listas persistentes.
- 2.3 Tipos de datos algebraicos. Árboles binarios de búsqueda. Operaciones de búsqueda, inserción y borrado. Ventajas y limitaciones.
- 2.4 Árboles balanceados. Árboles rojo-negro. Invariante roja-negra. Propiedades.
- 2.5 Heaps. Invariante. Operaciones sobre heaps. Leftist heaps. Propiedades.

Unidad 3: Abstracción y Verificación.

- 3.1 Tipos abstractos de datos. Ejemplos.
- 3.2 Especificación algebraica. Especificación mediante modelos.
- 3.3 Verificación de propiedades. Razonamiento ecuacional. Inducción matemática y estructural

Unidad 4: Programación en paralelo.

- 4.1 Algoritmos básicos de programación en paralelo. Análisis de complejidad.
- 4.2 Programando con secuencias en paralelo. TAD secuencias. Implementaciones y análisis de costos.
- 4.3 Programando con tablas en paralelo. TAD tabla. Implementaciones y análisis de costos.
- 4.4 Programando conjuntos en paralelos. Implementaciones y análisis de costos.
- 4.5 Búsquedas en paralelo. Búsqueda primero a lo ancho. Patrón de programación Map-reduce.

Modalidades de enseñanza

Se dan clases teóricas, clases de práctica en las que se resuelven ejercicios en papel, y clases de laboratorio, donde se resuelven problemas programando en computadora. Se realizan trabajos prácticos en grupo, los cuáles deben presentar y defender.

CGT1. Identificación, formulación y resolución de problemas de informática (RA1 RA2). Para esta competencia la modalidad de enseñanza es mediante la resolución de ejercicios en papel, y ejercicios que se resuelven con una computadora en laboratorio.

CGT4. Utilización de técnicas y herramientas de aplicación en la informática (RA3 RA4). Para esta competencia la modalidad de enseñanza es mediante la resolución de ejercicios en papel, y ejercicios que se resuelven con una computadora en laboratorio.

CGS1. Fundamentos para el desempeño en equipos de trabajo (RA5). Para esta competencia la modalidad de enseñanza es mediante trabajos prácticos grupales. Estos trabajos prácticos deben presentarse mediante un informe escrito, y una defensa oral, en las que deben responder preguntas de los docentes. Parte de la evaluación de la defensa es respecto a la organización, división de responsabilidades individuales y colectivas y funcionamiento del grupo en pos de una meta.

CGS2. Fundamentos para la comunicación efectiva (RA6). Para esta competencia la modalidad de enseñanza es mediante la preparación de informes escritos de los trabajos prácticos grupales, así como la defensa oral del mismo.

Recursos

Las clases teóricas y las prácticas en papel se realizan en aula. Para las clases teóricas se requiere un proyector multimedia. Para los laboratorios se requiere un laboratorio de informática.

Todo el material de estudio se sube al campus virtual (comunidades) provisto por la Universidad para que todos

los alumnos tengan acceso al material. También se utiliza el campus virtual para la entrega de los trabajos prácticos. Para la comunicación se utiliza un canal de Zulip que corre en un servidor del Departamento de Ciencias de la Computación.

Actividades de Formación Práctica

Las actividades de formación práctica consisten en trabajos prácticos integradores que contienen una parte de programación, y otra parte de resolución en papel.

Nº	Título	Descripción
1	Práctica 0	Ejercicios sobre complejidad asintótica (notación $O()$)
2	Práctica 1	Ejercicios sobre resolución y cotas de recurrencias.
3	Práctica 2	Ejercicios sobre programación funcional básica.
4	Práctica 3	Ejercicios sobre tipos de datos algebraicos y estructuras persistentes.
5	Práctica 4	Ejercicios sobre tipos abstractos de datos y verificación de propiedades.
6	TP1: Estructura de Datos Persistente.	Los alumnos implementan un tipo de abstracto de diccionario usando una estructura de datos persistente.
7	Práctica 5	Ejercicios de programación paralela.
8	Práctica 6	Ejercicios de programación en paralelo con secuencias.
9	TP2: Implementación de Secuencias	Los alumnos deben proveer dos implementaciones del TAD secuencias y analizar sus costos (trabajo y profundidad).
10	Práctica 7	Ejercicios sobre programación en paralelo, con tablas, conjuntos, y búsquedas (BFS).

Evaluación

Se regulariza mediante la aprobación de dos trabajos prácticos grupales y dos parciales individuales. Los alumnos tienen la posibilidad de recuperar uno de los dos parciales. Mediante los parciales se evalúan los resultados de aprendizaje RA1, RA2, RA3, y RA4.

Para aprobar los trabajos prácticos los alumnos deben presentar un trabajo escrito que es evaluado teniendo en cuenta el contenido técnico, pero también la estructura, organización, sintaxis, y claridad conceptual. Posteriormente hay un coloquio donde los docentes interrogan sobre contenidos específicos del documento, y sobre la división y organización del trabajo grupal. Esto se refleja en planillas que conforman documentación de evaluación del trabajo. Mediante estos trabajos prácticos se evalúan los resultados de aprendizaje RA5 y RA6.

El examen final consiste de una parte práctica con ejercicios en papel y un examen teórico oral donde deben exponer un tema de la asignatura y responder preguntas sobre el mismo. En caso que el alumno haya aprobado los parciales de la materia con promedio mayor a ocho, se considera que la parte práctica del examen está aprobada y sólo se evalúa el examen teórico oral. Para el caso de los alumnos libres, la parte práctica del examen final se extiende con ejercicios adicionales de manera de poder evaluar en forma más completa el temario de la asignatura.

Resultado de Aprendizaje	Actividades/Modalidad de Enseñanza	Modalidad de Evaluación
RA1	Resolución de ejercicios y práctica en laboratorio	Modalidad de Evaluación.
RA2	Resolución de ejercicios y práctica en laboratorio	Ejercicios escritos (Parcial y Final)

RA3	Resolución de ejercicios en papel	Ejercicios escritos (Parcial y Final)
RA4	Resolución de ejercicios en papel	Ejercicios escritos (Parcial y Final)
RA5	Trabajo práctico con informe y presentación oral.	Coloquio.
RA6	Trabajo práctico con informe y presentación oral	Coloquio de los TPs. Informe escrito y examen oral final

Bibliografía básica

Autores (Apellido, Inicial nombre)	Año de edición	Título de la obra	Editorial o Revista	Ejemplares disponibles o sitio web
Cormen T. H.; Leiserson, C. E.; Rivest, R. L.; Stein C.	2022	Introduction to Algorithms	MIT Press	4
Hutton, G.	2016	Programming in Haskell	Cambridge University Press	1
Brassard, G.; Bratley, P.	1995	Fundamentals of Algorithmics	Pearson	1

Bibliografía complementaria

Autores (Apellido, Inicial nombre)	Año de edición	Título de la obra	Editorial o Revista	Ejemplares disponibles o sitio web
Bird, R.	2000	Introducción a la Programación Funcional Usando Haskell	Prentice Hall	4

Distribución de la carga horaria

Presenciales

Teóricas		42 Hs.
Prácticas	Formación Experimental	
	Resolución de Problemas vinculados a la Profesión	20 Hs.
	Resolución de Problemas y Ejercicios	50 Hs.
	Actividades de Proyecto y Diseño	
	Formación en la Práctica Profesional	
Evaluaciones		8 Hs.
	Total	120 Hs.

Dedicadas por el alumno fuera de clase

	Preparación Teórica	10 Hs.
	Preparación Práctica	30 Hs.
	Elaboración y redacción de informes, trabajos, presentaciones, etc.	10 Hs.
	Total	50 Hs.

Cronograma de actividades

Semana	Unidad	Tema	Actividad
1	1	Introducción a la materia. Unidad 1.1	Práctica 0
2	1	Modelos de costos 1.2	Práctica 0
3	1	Resolución de recurrencias. 1.3	Práctica 1
4	2	Introducción a la programación Funcional. 2.1	Práctica 2
5	2	Tipos de datos. Listas persistentes. 2.2	Práctica 2
6	2	Estructuras funcionales. 2.3	Práctica 3
7	2		Práctica 3
8	3	3.1 y 3.2	Práctica 4 TP1
9	3	3.3	Práctica 4 TP1
10	3		Parcial 1
11	4	4.1	Práctica 5
12	4	4.2	Práctica 6 TP2
13	4	4.3, 4.4 y 4.5	Práctica 7
14			Parcial 2
15			Recuperatorio