

Planificación de  
**Estructuras de Datos y Algoritmos I**



Código/s: R-212

**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:	3º [LCC], 3º [LCC]		
Carga horaria:	105 hs. / 7 hs. semanales	Formato curricular:	Asignatura
Escuela:	Ciencias Exactas y Naturales	Departamento:	Ciencias de la Computación
Docente responsable:	SEVERINO GUIMPEL, Federico		

**Programa Sintético**

Introducción al paradigma de lenguajes imperativos. Introducción a la algoritmia: Definición, clasificación y orden de un algoritmo, instrucciones algorítmicas, expresiones. Complejidad de los algoritmos: cotas de complejidad, medidas asintóticas, ecuaciones en recurrencia. Programación en el lenguaje C. Métodos de ordenación. Técnicas de búsqueda en estructuras dinámicas.

**Espacios Curriculares Relacionados**

Previos Aprobados:	R-123 - Programación II
Simultaneos Recomendados:	R-213 - Lenguajes Formales y Computabilidad, R-211 - Álgebra lineal
Posteriores:	R-221 - Complementos de Matemática I, R-222 - Arquitectura del Computador, R-312 - Estructuras de Datos y Algoritmos II

**Vigencia desde 2024**

\_\_\_\_\_  
Firma Profesor

\_\_\_\_\_  
Fecha

\_\_\_\_\_  
Firma Aprob. Escuela

\_\_\_\_\_  
Fecha

Con el aval del Consejo Asesor:

## Fundamentación

La asignatura se ubica en el primer semestre del 2º año de la carrera Lic. en Cs. de la Computación. Esta actividad curricular continua la formación algorítmica desarrollada por el estudiante durante el primer año en la materia correlativa Programación II.

El estudio de estructuras de datos y diseño de algoritmos es fundamental para una carrera en Ciencias de la Computación. El incremento en los volúmenes de información a manipular, administrar y acceder en las aplicaciones obliga al uso de estructuras que minimicen los costos de acceso a la información y la elección adecuada de las mismas.

Esta materia se focaliza en el estudio profundo de almacenamiento y sus diferentes tipos de acceso considerando los costos asociados a las diferentes operaciones.

## Resultados del aprendizaje

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

RA1 Elegir una representación eficiente de la información en la creación de programas informáticos.

RA2 Describir, utilizar e implementar distintas estructuras de datos dinámicas.

RA3 Aplicar competencias de programación en la creación de programas informáticos.

RA4 Analizar el costo de un algoritmo tanto espacial como temporalmente.

RA5 Utilizar herramientas de desarrollo de software colaborativas para el trabajo en equipo.

## 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	Bajo	RA1, RA3
CGT2-Concepción, diseño y desarrollo de proyectos de informática	Bajo	RA2, RA3, RA4
CGT3-Gestión, planificación, ejecución y control de proyectos de informática	Bajo	RA5
CGS1-Fundamentos para el desempeño en equipos de trabajo	Bajo	RA5
CGS5-Fundamentos para el aprendizaje continuo	Bajo	RA3

## Programa Analítico

### Unidad 1

- 1.1 Lenguajes imperativos, algoritmos y estructuras de datos.
- 1.2 Los lenguajes de programación. El proceso de compilación.
- 1.3 Estructuras de datos y tipos de datos abstractos.
- 1.4 Introducción a la complejidad algorítmica
- 1.5 Herramientas para verificar errores de memoria.
- 1.6 Herramientas de control de código fuente y versionado con Git.
- 1.7 Asignación de tareas usando Jira o similar.

### Unidad 2:

- 2.1 Arreglos dinámicos
- 2.2 Estructuras de datos dinámicas y tipos abstractos compuestos

- 2.3 Listas.
- 2.4 Listas simplemente enlazadas.
- 2.5 Listas doblemente enlazadas.
- 2.6 Listas Circulares
- 2.7 Listas como arreglos.
- 2.8 Listas generales

#### Unidad 3:

- 3.1 El TAD Pila
- 3.2 El TAD Cola
- 3.3 Implementación del TAD Pila usando arreglos
- 3.4 Implementación del TAD Pila usando listas
- 3.5 Implementación del TAD Cola usando arreglos
- 3.6 Implementación del TAD Cola usando listas

#### Unidad 4

- 4.1 Árboles
- 4.2 Árbol Binario
  - 4.2.1 Árboles de Expresiones Aritméticas
  - 4.2.2 Recorridos
- 4.3 Búsqueda Binaria
- 4.4 Árboles de Búsqueda Binaria
- 4.5 Colas de Prioridades
- 4.6 Heaps
- 4.7 Árboles AVL
- 4.8 Árboles Generales

#### Unidad 5

- 5.1 Tablas Hash
- 5.2 Funciones Hash
- 5.3 Colisiones
  - 5.3.1 Encadenamiento
    - Hashing con listas mezcladas
  - 5.3.2 Direccionamiento Abierto
    - Linear Probing
    - Hashing Doble

#### Unidad 6

- 6.1 Ordenación.
  - 6.1.1 ¿Por qué ordenar?
  - 6.1.2 El problema de ordenación
- 6.2 Ordenación por comparación
  - 6.2.1 Burbuja
  - 6.2.2 Selección
  - 6.2.3 Inserción
- 6.3 Divide and Conquer
- 6.4 Ordenación por mezcla
  - 6.3.1 Mergesort
  - 6.3.2 Quicksort
- 6.5 Backtracking
- 6.6 Algoritmos Greedy
- 6.7 Programación Dinámica

## Modalidades de enseñanza

Se dan clases teóricas en aula, clases prácticas en laboratorio y clases prácticas en aula donde se resuelven problemas para pensar, usando el concepto de aula invertida y se trabaja en grupos.

CGT1. Identificación, formulación y resolución de problemas de informática. Para esta competencia la modalidad de enseñanza es mediante es mediante la resolución de problemas en las clases de aula y laboratorios.

CGT2. Concepción, diseño y desarrollo de proyectos de informática. Para esta competencia la modalidad de enseñanza es mediante es mediante la resolución de problemas en las clases de aula y laboratorios.

CGT3. Gestión, planificación, ejecución y control de proyectos de informática. Para esta competencia la modalidad de enseñanza es mediante es mediante las clases teóricas.

CGS1. Fundamentos para el desempeño en equipos de trabajo. Para esta competencia la modalidad de enseñanza es mediante las resoluciones de ejercicios de la práctica realizados en grupos durante las clase de práctica en aula.

CGS5. Fundamentos para el aprendizaje continuo. Para esta competencia la modalidad de enseñanza es mediante la preparación del trabajo práctico final.

## Recursos

Las clases teóricas y de resolución de problemas se realizan en aula. Las clases de laboratorio se realizan en laboratorio informático. Para las clases de práctica y laboratorio se utiliza un proyector.

En el Campus virtual de la UNR se encuentra el material teórico, las prácticas, soluciones de ejercicios seleccionados y referencias bibliográficas.

Para la comunicación con los alumnos se utiliza un canal de Zulip que se encuentra instalado en un servidor del Departamento de Ciencias de la Computación.

## Actividades de Formación Práctica

Nº	Título	Descripción
1	Arreglos	Se introducen nociones de complejidad en la resolución de ejercicios sobre arreglos.
2	Listas	Listas Simplemente Enlazadas. Listas Doblemente Enlazadas. Listas Generales.
3	Pilas y Colas	Se realiza una implementación de Pilas y Colas generales usando arreglos y listas.
4	Árboles	Árboles Binarios. Árboles de Búsqueda Binaria. Árboles AVL.
4	Árboles (continuación): Heaps	Implementación de Heaps generales. Construcción de Heaps a partir de arreglos. Implementación de Colas de Prioridades.
5	Tablas Hash	Implementación de Tablas Hash generales usando diferentes criterios de resolución de colisiones.

8	Introducción a los algoritmos	Se introduce al estudiante en analizar y resolver problemas teniendo en cuenta la complejidad (costo) temporal y espacial de las soluciones. En general, queremos las soluciones óptimas en tiempo. Si hay varias, preferimos las que además tengan menor complejidad espacial.
9	Ordenamiento	Implementación de algoritmos de ordenación. Análisis

## Evaluación

La evaluación se realiza mediante 3 parciales teóricos-prácticos y la resolución de ejercicios de las prácticas, elegidos al azar, los cuales deben ser explicados por el/la estudiante.

El estudiante alcanza la condición de Regular, si aprueba con nota mayor o igual a 6 los 3 parciales y las resoluciones de las prácticas presentadas son correctas. Caso contrario, obtiene la condición de Libre.

Para las o los estudiantes que hayan alcanzado la condición Regular, el examen final consiste en la entrega de un Trabajo Práctico final integrador. El enunciado de este trabajo práctico cambia cada año.

Luego de la aprobación del TP se realiza un coloquio donde se debe justificar las decisiones tomadas en el desarrollo del TP y se evalúan temas dados en la materia.

Para las o los estudiantes en condición Libre el examen final consiste en la resolución de un examen final práctico en computadora (que equivale a un mini proyecto de TP final) y un coloquio sobre temas dados en la materia.

Resultado de Aprendizaje	Actividades/Modalidad de Enseñanza	Modalidad de Evaluación
RA1	Clases de teoría y de práctica	Ejercicios de parciales y trabajo final
RA2	Clases de teoría, práctica en aula y laboratorio	Ejercicios de parciales
RA3	Clases de teoría y clases de laboratorio	Ejercicios de parciales y trabajo final
RA4	Clases de práctica y clases de laboratorio	Ejercicios de parciales
RA5	Clases de laboratorio	Coloquio trabajo final

## Bibliografía básica

Autores (Apellido, Inicial nombre)	Año de edición	Título de la obra	Editorial o Revista	Ejemplares disponibles o sitio web
Thomas Cormen	2009	Introduction to algorithms	MIT	4
Alfred Aho, John Hopcroft y Jeffrey Ullman.	1999	Estructuras de Datos y Algoritmos	Addison-Wesley	3
Brian Kernighan, Dennis Ritchie	1991	El lenguaje de programación C	Prentice Hall	6

## Bibliografía complementaria

Autores (Apellido, Inicial nombre)	Año de edición	Título de la obra	Editorial o Revista	Ejemplares disponibles o sitio web
------------------------------------	----------------	-------------------	---------------------	------------------------------------

## Distribución de la carga horaria

**Presenciales**

Teóricas		45 Hs.
Prácticas	Formación Experimental	20 Hs.
	Resolución de Problemas vinculados a la Profesión	20 Hs.
	Resolución de Problemas y Ejercicios	
	Actividades de Proyecto y Diseño	20 Hs.
	Formación en la Práctica Profesional	
Evaluaciones		
<b>Total</b>		<b>105 Hs.</b>

**Dedicadas por el alumno fuera de clase**

	Preparación Teórica	15 Hs.
	Preparación Práctica	30 Hs.
	Elaboración y redacción de informes, trabajos, presentaciones, etc.	10 Hs.
<b>Total</b>		<b>55 Hs.</b>

**Cronograma de actividades**

Semana	Unidad	Tema	Actividad
1	1.1 al 1.3, 1.6	Presentación de la materia. Introducción a los diferentes temas que se abordarán. Presentación de herramientas colaborativas.	1, C1
2	1.4, 1.5, 1.7 y 2.1	Introducción a la complejidad de problemas. Arreglos dinámicos. Presentación de herramientas colaborativas.	A1, C1
3	2.2 a 2.4	¿Cómo implementar estructuras dinámicas? Listas Simplemente Enlazadas	2
4	2.4 a 2.8	Listas Doblemente Enlazadas. Listas Circulares. Listas Generales	2
5	3	Pilas y Colas	3
6	6.1 y 6.2	Motivación e introducción a los Algoritmos de Ordenación	A2
7	4.1 al 4.2	Introducción a Árboles. Árboles binarios	4
8	4.3 a 4.4	Búsqueda Binaria. ABB	4
9	4.5 a 4.8	Heaps. Árboles AVL.	4
10	5.1 y 5.2	Introducción y motivación de las Tablas Hash.	5
11	5.3	Colisiones. Dificultades y ventajas de cada	5
12	6.3 y	Divide and Conquer. Más algoritmos de ordenación	A2
13	6.5 y 6.6	Backtracking. Algoritmos Greedy	A3
14	6.7	Programación dinámica	A3

15		Repaso general de la materia y presentación del trabajo final	
----	--	---	--