



---

**PROGRAMA CURSO: ALGORITMOS Y ESTRUCTURA DE DATOS**  
I Semestre, 2016

---

**Datos Generales**

**Sigla:** IF-3001

**Nombre del curso:** Algoritmos y estructura de datos

**Tipo de curso:** Teórico-Práctico

**Número de créditos:** 4

**Número de horas semanales presenciales:** 8

**Número de horas semanales de trabajo independiente del estudiante:** 12

**Requisitos:** IF-2000 Programación I

**Correquisitos:** N/A

**Ubicación en el plan de estudio:** III Ciclo

**Horario del curso:** L: 5:00 – 8:50. J: 5:00 – 8:50

**Suficiencia:** No

**Tutoría:** No

---

**Datos del Profesor**

**Nombre:** Mci. Alejandro Ulate

**Correo electrónico:** [alejandro.ulate@ucrsi.info](mailto:alejandro.ulate@ucrsi.info)

**Horario de consulta:** V: 5:00 – 9:00 p.m.

---

**1. Descripción del curso**

En este curso se analiza en detalle el concepto de algoritmo, así como de sus propiedades. Se analiza la fuerte relación entre algoritmos y estructura de datos. Se hace un recorrido por algunos de los algoritmos y estructuras de datos más importantes de la teoría de la computación y se determina qué tipo de problemas pueden ser eficientemente resueltos en un computador. El estudiante desarrollará la programación de los algoritmos.

---

**2. Objetivo General**

Introducir al estudiante a la abstracción de problemas mediante el uso de algoritmos y estructuras de datos básicas, para la solución de problemas computacionales.

---



### 3. Objetivos específicos

Al finalizar el curso el o la estudiante estará en capacidad de:

- a. Analizar la complejidad y el orden de duración de un algoritmo.
- b. Estudiar las estructuras de datos básicas para la implementación de tipos de datos abstractas empleadas en aplicaciones computacionales.
- c. Fomentar la valoración de los algoritmos con base en el contexto de la solución computacional donde son utilizados.
- d. Implementar los algoritmos en un lenguaje de programación.

---

### 4. Contenidos

1. Introducción a la teoría de algoritmos
  - 1.1 Estudio del concepto de algoritmo
  - 1.2 Tiempo y orden de ejecución de los algoritmos
  - 1.3 Algoritmos recursivos
2. Tipos de algoritmos
  - 2.1 Concepto y características
    - 2.1.1 Algoritmos voraces
    - 2.1.2 Búsquedas exhaustivas
    - 2.1.3 Programación dinámica
    - 2.1.4 Algoritmos divide y vencerás
    - 2.1.5 Algoritmos probabilísticos
3. Tipos de datos abstractos (TDA)
  - 3.1 Conceptos básicos de los TDA
  - 3.2 Pilas
    - 3.2.1 Concepto, características e implementación de una pila
    - 3.2.2 Pilas con arreglos
  - 3.3 Colas
    - 3.3.1 Concepto, características e implementación de una cola
    - 3.3.2 Colas con arreglos
  - 3.4 Listas
    - 3.4.1 Concepto, características e implementación de una lista
    - 3.4.2 Listas simples, circulares, dobles, dobles circulares
    - 3.4.3 Pilas, colas y colas de prioridad basadas en listas
  - 3.5 Árboles
    - 3.5.1 Concepto y características de árboles
    - 3.5.2 Árboles binarios
    - 3.5.3 Balanceo de árboles



4. Procesamiento de hileras
  - 4.1 Concepto y características de comprensión y criptografía
  - 4.2 Compresión de archivos
    - 4.2.1 Algoritmo Hoffman
  - 4.3 Criptografía
    - 4.3.1 Algoritmos criptográficos
5. Algoritmos de ordenamiento
  - 5.1 Concepto y características de los algoritmos de ordenamiento
  - 5.2 Algoritmos elementales:
    - 5.2.1 Burbuja y burbuja mejorada
    - 5.2.2 Inserción
    - 5.2.3 Selección
  - 5.3 Algoritmos complejos
    - 5.3.1 Quick sort
    - 5.3.2 Radix sort
    - 5.3.3 Merge sort
    - 5.3.4 Shell sort
6. Algoritmos de búsqueda
  - 6.1 Concepto y características de las búsquedas
  - 6.2 Algoritmos elementales
    - 6.2.1 Secuencial
    - 6.2.2 Búsqueda binaria
7. Algoritmos para grafos
  - 7.1 Concepto y características de los grafos
    - 7.1.1 Dirigidos, no dirigidos, vértice, arista, entre otros
    - 7.1.2 Matriz adyacencia
  - 7.2 Recorridos
    - 7.2.1 Anchura, profundidad y ciclos
  - 7.3 Conectividad
    - 7.3.1 No conexo, conexo, fuertemente conexo y completo
  - 7.4 Árbol de expansión mínima
    - 7.4.1 Algoritmos de Kruskal y Prim
  - 7.5 El problema de la ruta más corta
    - 7.5.1 Algoritmos Dijkstra y Floyd

---

## **5. Metodología**

El curso es teórico práctico, donde el estudiante debe desarrollar problemas en grupo e individualmente, que permita poner en práctica los conocimientos adquiridos.

Las prácticas serán definidas por parte del profesor el cual dará únicamente los lineamientos generales a seguir. El o los problemas escogidos y su solución aplicada deben ser expuestos por los grupos respectivos.



## 6. Evaluación

<i>Descripción</i>	<i>Porcentaje</i>
Examen Parcial I	25%
Examen Parcial II	25%
Quices, tareas cortas, exposiciones, otros	25%
2 proyectos programados	25%
<b>Total:</b>	<b>100%</b>

### Consideraciones sobre la evaluación

- ✓ Los Quices se harán sin previo aviso en cualquier momento de la lección y no se harán reposiciones (excepto por las disposiciones establecidas en el reglamento de Régimen Académico Estudiantil).
- ✓ Las fechas de las evaluaciones del cronograma están sujetas a cambios dependiendo del avance de los temas vistos.
- ✓ Los rubros de calificación de cada una de las evaluaciones serán entregadas junto con la especificación de la misma.
- ✓ Las tareas programadas se realizarán individualmente o en grupos establecidos por el profesor.
- ✓ Las tareas programadas deben ir acompañadas de la respectiva documentación interna. No se recibirá la misma posterior a la entrega.
- ✓ Las tareas programadas deben ser entregadas a la hora y fecha indicadas en el enunciado del proyecto. El no entregar cualquier evaluación a tiempo provocará que se pierda el 100% de la nota. En casos muy justificados y previamente aprobados por el profesor se podrá recibir un trabajo hasta 24 horas después de la hora de entrega, en este caso perderán el 40% del valor del trabajo.
- ✓ Para los demás aspectos de evaluación no se aceptarán entregas después de la fecha y hora solicitada.
- ✓ Para las tareas programadas se realizará una comprobación individual para determinar la participación de los integrantes.
- ✓ Para todas las entregas sólo se recibirán aquellas cuyo contenido sea exclusivamente de la asignación respectiva.
- ✓ Mantener celulares en modo silencioso o apagado durante las lecciones.
- ✓ La aparición de un celular durante un examen o quiz anulará automáticamente el mismo.
- ✓ La comprobación de que alguna tarea, laboratorio, proyecto o examen es una copia hará que se apliquen las sanciones que contemple el reglamento de Régimen Académico Estudiantil. Consultar en: [http://cu.ucr.ac.cr/normativ/regimen\\_academico\\_estudiantil.pdf](http://cu.ucr.ac.cr/normativ/regimen_academico_estudiantil.pdf).
- ✓ Se utilizará la plataforma moodle.ucrso.info o la que el profesor designe.



## 7. Cronograma.

	SEMANA	TEMA
1	07 marzo al 12 marzo	Presentación curso y entrega carta al estudiante Tema 1: Introducción a la teoría de algoritmos
2	14 marzo al 19 Marzo	Tema 2: Tipos de algoritmos
3	21 marzo al 26 Marzo	<b>Semana Santa</b>
4	30 marzo al 02 Abril	Tema 3: Tipos de datos abstractos (Pilas)
5	28 marzo – 02 abril	Tema 3: Tipos de datos abstractos (Colas)
6	04 abril al 09 abril	Tema 3: Tipos de datos abstractos (Listas)
7	11 abril al 16 abril	Tema 3: Tipos de datos abstractos (Listas) <b>11 de abril feriado</b>
8	18 abril al 23 abril	Tema 3: Tipos de datos abstractos (Árboles)
9	25 abril al 30 abril	Tema 3: Tipos de datos abstractos (Árboles) <b>Semana Universitaria</b>
10	2 mayo al 7 mayo	Tema 3: Tipos de datos abstractos (Árboles) <b>I Examen</b> <b>Entrega I Tarea Programada</b>
11	9 mayo al 14 mayo	Tema 4: Procesamiento de hileras Tema 5: Algoritmos de ordenamiento
12	16 mayo al 21 mayo	Tema 5: Algoritmos de ordenamiento
13	23 mayo al 28 mayo	Tema 6: Algoritmos de búsqueda
14	30 mayo al 4 junio	Tema 7: Algoritmos para grafos
15	6 junio al 11 junio	Tema 7: Algoritmos para grafos
16	13 junio al 18 junio	Tema 7: Algoritmos para grafos
17	20 junio al 25 junio	<b>II Examen</b> <b>Entrega II Tarea Programada</b>
18	27 junio al 2 julio	<b>Entrega de promedios</b>
19	04 julio al 9 de julio	<b>Ampliación</b>

\*El cronograma está sujeto a cambios durante el semestre, los cuales serán informados durante las lecciones.

### Bibliografía

1. Allen Weiss, Mark. Estructura de Datos en Java, 4ta edición. Addison Wesley. Madrid. 2013.
2. Michael T. Goodrich; Roberto Tamassia. Data Structures and Algorithms in Java. 4 ed. John Wiley & Sons, Inc.
3. Drozdek, Adam. Estructura de datos y algoritmos en Java. Thomson, Mexico. 2007.
4. Deitel y Deitel. Java: How to program? 5 ed. Prentice Hall. 2003.
5. Joyanes y Zahonero. Fundamentos de Programación - Algoritmos, Estructuras de Datos y Objetos. Editorial Mc Graw Hill, tercera edición 2004.
6. Joyanes, Luis. Programación en Java2. Algoritmos, Estructuras de Datos y Programación Orientada a Objetos. Editorial Mc Graw Hill, primera edición, 2002.
7. Martí, Ortega y Verdero. Estructuras de Datos y Métodos Algorítmicos – Ejercicios resueltos. Editorial Pearson Prentice – Hall, 2003.
8. Heileman, Gregory. Estructuras de datos, algoritmos, programación orientada a objetos. McGraw Hill. 1998.