



SÍLABO 2025-II

I. DATOS ADMINISTRATIVOS

1. Nombre de la Asignatura: **Estructuras de Datos y Algorítmica para Desarrollo de Software**
2. Código: I10401
3. Naturaleza: Teórica, laboratorio
4. Condición: Obligatorio
5. Requisitos: IF0302 Taller de Programación II
6. Nro. Créditos: 4.5
7. Nro. de horas: Teóricas=3 / Laboratorio=2
8. Semestre Académico: 2025-2
9. Docente: M. Sc. Eduardo Yauri Lozano
10. Correo Docente: Eduardo.yauri@urp.edu.pe

II. SUMILLA

El curso de Estructuras de Datos y Algorítmica corresponde al cuarto semestre de la formación de la Escuela Profesional de Ingeniería Informática es de naturaleza teórico-práctica y forma parte del área de formación fundamental del estudiante. Al finalizar la asignatura el estudiante tendrá los conocimientos y las habilidades para la resolución de problemas complejos mediante programas de computador utilizando eficazmente estructuras de datos y algoritmos.

Síntesis del contenido:

(1) Análisis de algoritmos y abstracción de datos. (2) Resolución de problemas y recursividad. (3) Listas, pilas y colas. (4) Tablas de dispersión de tipo abierta. (5) Conjuntos. (6) Árboles binarios y árboles n-arios. (7) Proyectos de Aplicación.

III. COMPETENCIAS GENÉRICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Resolución de problemas.
- Investigación científica y tecnológica.
- Autoaprendizaje

IV. COMPETENCIAS ESPECÍFICAS A LAS QUE CONTRIBUYE LA ASIGNATURA

- Soluciona problemas de ingeniería.
- Dominio de las ciencias.
- Experimentación y pruebas.
- Aprendizaje durante toda la vida.
- Práctica moderna de ingeniería.
- Gestión del Proyectos.
- Responsabilidad Ética
- Comunicación.
- Trabajo en equipo.

V. CONTRIBUCIÓN A LA INVESTIGACIÓN FORMATIVA

El curso de Estructura de Datos y Algoritmia contribuye significativamente a la investigación formativa al fomentar en los estudiantes una actitud analítica y metódica frente a la solución de problemas computacionales. A través del estudio comparativo de algoritmos, la evaluación de estructuras de datos y el análisis de su eficiencia en distintos contextos, los estudiantes desarrollan habilidades de indagación, reflexión crítica y validación de soluciones, que son fundamentales en la formación investigativa.

VI. CONTRIBUCIÓN A LA RESPONSABILIDAD SOCIAL

El curso de Estructura de Datos y Algoritmia contribuye a la responsabilidad social al desarrollar en los estudiantes la capacidad de diseñar soluciones algorítmicas eficientes, éticas y sostenibles para problemas del entorno. A través del análisis y modelado de datos, los estudiantes pueden abordar situaciones reales que afectan a la sociedad, como la optimización de recursos en instituciones públicas, la mejora de sistemas de



UNIVERSIDAD RICARDO PALMA
Oficina de Desarrollo Académico, Calidad y Acreditación
FACULTAD DE INGENIERÍA
Escuela Profesional de Ingeniería Informática

salud o educación, o la gestión de información en organizaciones sociales.

VII. LOGRO DE LA ASIGNATURA

Al finalizar la asignatura el estudiante tendrá los conocimientos y las habilidades para la resolución de problemas complejos mediante programas de computador utilizando eficazmente estructuras de datos y algoritmos.

VIII. PROGRAMACIÓN DE CONTENIDOS

UNIDAD I: ALGORITMOS, TIEMPO DE EJECUCIÓN DE UN PROGRAMA Y RECURSIVIDAD.	
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante analiza y aplica conceptos fundamentales de algoritmos, tipos de datos y estructuras de datos, así como técnicas de recursividad directa, indirecta y de cola, para resolver problemas computacionales de diversa complejidad, estimando su eficiencia mediante el cálculo de complejidad algorítmica. Criterios: Claridad en la representación de algoritmos, correcto uso de notación Big-O, selección adecuada del tipo de recursividad, justificación del enfoque algorítmico utilizado. Resultado: Desarrollo de algoritmos recursivos documentados, ejercicios resueltos con análisis de complejidad, y participación en actividades prácticas de retroalimentación.	
Semana	Contenido
1	Análisis de algoritmos y abstracción de datos: - Definición de algoritmos y características. - Representación de algoritmos y programas. - Tipos de datos primitivos, tipos de datos abstractos (TDA) y estructuras de datos.
2	Análisis de complejidad de Algoritmos: - Definiciones base de complejidad de algoritmos. - Medición del tiempo de ejecución de un programa. - Notación o mayúscula (Big - O). - Reglas para calcular la complejidad de un programa
3	Resolución de problemas y recursividad: - Concepto de recursividad. Problemas y soluciones recursivas. - La recursividad frente a la iteración y los problemas de la recursividad. - La recursividad de cola. Eliminación de la recursividad de cola. - Recursividad directa e indirecta. Uso de funciones auxiliares en la resolución de problemas recursivos. - Metodología para la resolución de problemas. - Resolución y cálculo de complejidad de problemas recursivos de baja complejidad. - Resolución de ejercicios prácticos de recursividad de baja complejidad.
4	Resolución de problemas y recursividad: - Resolución y cálculo de complejidad de problemas recursivos de baja y mediana complejidad. - Resolución de ejercicios prácticos de recursividad de baja y media complejidad



UNIVERSIDAD RICARDO PALMA
Oficina de Desarrollo Académico, Calidad y Acreditación
FACULTAD DE INGENIERÍA
Escuela Profesional de Ingeniería Informática

5	Resolución de problemas y recursividad: • Resolución y cálculo de complejidad de problemas recursivos de mediana y alta complejidad. • Resolución de ejercicios prácticos de recursividad de media y alta complejidad • Evaluación y retroalimentación del logro.
UNIDAD II: EL TDA LISTA Y LISTAS ESPECÍFICAS (PILA Y COLA)	
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante implementa y manipula estructuras de datos lineales, como listas, pilas y colas, utilizando tanto arreglos como listas enlazadas, comprendiendo su comportamiento, operaciones principales y aplicaciones, además de desarrollar soluciones a problemas representados mediante estos TDA. Criterios: Implementación funcional con estructuras adecuadas, reutilización de listas enlazadas en la creación de pilas y colas, correcta ejecución de operaciones (insertar, eliminar, recorrer), análisis lógico de casos prácticos. Resultado: Programas funcionales que resuelven problemas utilizando listas, pilas y colas; ejercicios resueltos con documentación del código; evaluación de logro individual y retroalimentación formativa.	
Semana	Contenido
6	El TDA Lista: • Definición del TDA Lista. • Implementación de listas mediante arreglos. • Resolución de ejercicios de TDA lista y su implementación con arreglos.
7	• Monitoreo y Retroalimentación mediante la resolución de ejercicio de repaso de todos los temas visto hasta la actualidad.
8	EVALUACIÓN PARCIAL DEL CURSO
9	El TDA Lista: • Implementación de listas mediante nodos enlazados con un solo enlace. • Resolución de ejercicios de TDA lista y su implementación con Listas Enlazadas.
10	Listas específicas PILAS y COLAS: • Pilas y colas: Definición y estudio de sus principales funciones. • Implementación a partir de la reutilización de listas enlazadas • Monitoreo y Retroalimentación. Evaluación del Logro.
UNIDAD III: EL TD TABLA DE DISPERSIÓN. EL TDA CONJUNTO.	
LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante implementa y aplica estructuras de datos basadas en tablas de dispersión y conjuntos, comprendiendo su funcionamiento interno, uso de pares clave-valor y su reutilización en la implementación del TDA Conjunto, resolviendo problemas representativos que requieren operaciones eficientes de almacenamiento y búsqueda. Criterios: Correcta construcción de tablas hash de tipo abierta, manipulación eficiente de elementos clave-valor, reutilización estructurada de código, uso correcto de operaciones de conjuntos (unión, intersección, diferencia). Resultado: Implementaciones funcionales de tablas de dispersión y conjuntos, ejercicios resueltos con análisis de funcionamiento, y evidencia de evaluación y retroalimentación del logro.	



UNIVERSIDAD RICARDO PALMA
Oficina de Desarrollo Académico, Calidad y Acreditación
FACULTAD DE INGENIERÍA
Escuela Profesional de Ingeniería Informática

Semana	Contenido
11	Tablas de dispersión: - Definición de la estructura de datos Tabla de Dispersión (TD) y sus principales funciones. - Tabla de dispersión de tipo abierta. - Implementación de tablas de dispersión de tipo abierta con el uso de pares clave-valor. - Resolución de ejercicios de TDA Tabla de dispersión y su implementación.

12	El TDA Conjunto: - Definición del TDA Conjunto y sus principales funciones. - Implementación de conjuntos a través de la reutilización de tablas de dispersión. - Resolución de ejercicios de TDA Conjunto y su implementación. - Evaluación y retroalimentación del logro.
----	--

UNIDAD IV: ÁRBOLES. PROYECTO DE CURSO.

LOGRO DE APRENDIZAJE: Al finalizar la unidad, el estudiante diseña e implementa árboles binarios, árboles binarios especiales y árboles n-arios, aplicándolos en la solución de problemas estructurados y jerárquicos, y desarrolla un proyecto integrador que articula los conocimientos adquiridos durante el curso, demostrando dominio técnico y capacidad analítica.

Criterios: Correcta construcción y recorrido de árboles, resolución eficiente de problemas jerárquicos, integración coherente de estructuras estudiadas, claridad en la presentación oral y escrita del proyecto.

Resultado: Programas funcionales con estructuras de árbol aplicadas a problemas reales, solución documentada de ejercicios diversos de árboles, presentación y defensa del proyecto final.

Semana	Contenido
13	Árboles binarios: - Definición y Tipos - Creación y recorrido. - Árbol binario de búsqueda. - Resolución de problemas en los cuales se usa árboles binarios.
14	Árboles binarios y n-arios: - Árboles binarios especiales. - Definición, creación y recorrido de árboles n-arios. - Resolución de problemas que usan árboles especiales y n-arios. - Resolución de ejercicios diversos de TDA árbol. - Evaluación y retroalimentación del logro.
15	- Monitoreo y Retroalimentación mediante la resolución de ejercicio de repaso de todos los temas visto en la segunda parte del curso. - Presentación y exposición del proyecto final del curso.



16	EVALUACIÓN FINAL DEL CURSO
17	EVALUACIÓN SUSTITUTORIA DEL CURSO

IX. ESTRATEGIAS DIDÁCTICAS

Disertación, Aprendizaje Basado en Proyectos, Problemas; Aprendizaje Colaborativo, Aprendizaje Basado en Investigación, Estudio de Casos, Talleres, etc. Se podrán desarrollar actividades sincrónicas (que los estudiantes realizarán al mismo tiempo con el docente) y asincrónicas (que los estudiantes realizan independientemente fortaleciendo su aprendizaje autónomo. La planificación y ejecución de las sesiones de aprendizaje deberán considerar actividades que se organizan de la siguiente manera:

Exploración: preguntas de reflexión vinculada con el contexto, entre otros.

Problematicación: conflicto cognitivo de la unidad, otros.

Motivación: bienvenida y presentación del curso, otros.

Presentación: PPT, otros.

Práctica: resolución individual de un problema, resolución colectiva de un problema, entre otros.

Evaluación de la unidad: presentación del resultado o producto.

Extensión / Transferencia: presentación de la resolución individual de un problema.

X. EVALUACIÓN

Las evaluaciones se realizan a lo largo del semestre con el propósito de determinar en qué medida el estudiante va logrando las competencias de la asignatura. Las actividades de enseñanza se complementarán con actividades de evaluación continua (AEC) tales como:

Los exámenes parcial y final se realizarán en las semanas 8 y 16.

El promedio final de la asignatura se obtendrá de la manera siguiente:

$$\frac{0.4 * ((PAR1 + FIN1) / 2) + 0.2 * ((LAB1 + LAB2 + LAB3 + LAB4 + LAB5) / 5) + 0.2 * ((PRO1 + PRO2 + EXP1) / 3) + 0.2 * (PRT1)}{1}$$

Dónde: PAR1 es examen parcial, FIN1 es examen final, LAB es nota de laboratorio, PRO es avance del proyecto del curso, EXP1 es exposición del proyecto del curso y PRT1 es participación en el curso.

Para la evaluación constante del curso se usan los siguientes instrumentos de evaluación:

- Pruebas Escritas: Para evaluar conocimientos teóricos, análisis de algoritmos, notación Big-O y estructuras de datos.
- Exposiciones y defensas orales: Para evaluar la comprensión, argumentación técnica, y justificación de decisiones de implementación.
- Autoevaluación y coevaluación: Para fomentar el pensamiento crítico y la valoración de los propios aprendizajes o los del equipo.

XI. RECURSOS

- Equipos: computadora, laptop.
- Materiales: apuntes de clase del docente, separatas de problemas, lecturas, videos.
- Plataformas: Entorno de programación, diagramadores, pizarra.

XII. REFERENCIAS

Bibliografía Básica

- Drozdek, A. (2007). *Estructura de datos y algoritmos en Java*. Thomson. (Disponible en la Biblioteca Virtual, código 25L-INF-BVI-05Luis Joyanes. Lucas Sanchez. Ignacio Zahonero. Estructura de Datos en C++. Editorial McGrawHill. 2007.



UNIVERSIDAD RICARDO PALMA
Oficina de Desarrollo Académico, Calidad y Acreditación
FACULTAD DE INGENIERÍA
Escuela Profesional de Ingeniería Informática

- Hernández Bejarano, M., & Baquero Rey, L. E. (2021). *Estructuras de datos: Fundamentación práctica* (1.ª ed.). Ediciones de la U LTDA.
- Deitel, P. J., & Deitel, H. M. (2016). *C++ how to program* (10th ed.). Pearson. ISBN 978-0134448237.

Bibliografía complementaria

- Bruno Lopez Takeyas. Notación O grande. (s.f.). *Apuntes de Matemáticas para Computación*. Instituto Tecnológico de Nuevo Laredo. Recuperado en junio del 2025, de: [\[https://nlaredo.tecnm.mx/takeyas/Apuntes/Matematicas%20para%20Computacion/Apuntes/Notacion%20O%20grande.pdf\]](https://nlaredo.tecnm.mx/takeyas/Apuntes/Matematicas%20para%20Computacion/Apuntes/Notacion%20O%20grande.pdf)
- Santiago Chio (s.f). *Chio Code: Diseño y Análisis de Algoritmos [Canal de YouTube]*. YouTube. [\[https://www.youtube.com/@ChioCode\]](https://www.youtube.com/@ChioCode)
- **Introducción a C++**. (s.f.). *Fundamentos de informática – Universidad de Valladolid*. Recuperado en junio de 2025, de: [\[https://www2.eii.uva.es/fund_inf/cpp/temas/1_introduccion.html\]](https://www2.eii.uva.es/fund_inf/cpp/temas/1_introduccion.html)