

# SÍLABO

## Estructura de Datos y Algoritmos

<b>Código</b>	24UC00430	<b>Carácter</b>	Obligatorio	
<b>Requisito</b>	Lenguaje de Programación 2			
<b>Créditos</b>	3			
<b>Horas</b>	<b>Teóricas</b>	2	<b>Prácticas</b>	2
<b>Año académico</b>	2025			

### I. Introducción

Estructura de Datos y Algoritmos es una asignatura de especialidad y de carácter obligatorio para la Escuela Académico Profesional de Ciencia de la Computación, que se ubica en el cuarto ciclo de estudios. Esta asignatura contribuye a desarrollar las competencias Diseño y Desarrollo de Soluciones, y Conocimientos de Computación, en el nivel 2. Tiene como requisito Lenguajes de Programación 2. Por su naturaleza, incluye componentes teóricos y prácticos que permiten realizar representaciones y manipulaciones eficientes de los datos para proponer soluciones computacionales a problemas prácticos. Por otro lado, debido a la naturaleza de los contenidos que desarrolla, la asignatura puede tener un formato virtual en la modalidad A Distancia.

Los contenidos generales que la asignatura desarrolla son los siguientes: estructuras de datos básicas y estáticas, estructura de datos dinámica, estructura de datos avanzada, uso de grafos y tablas.

### II. Resultado de aprendizaje de la asignatura

Al finalizar la asignatura, el estudiante será capaz de evaluar técnicas eficientes de implementación de estructuras de datos para la solución de problemas computacionales con ciertas restricciones de espacio y tiempo.

**III. Organización de los aprendizajes**

<b>Unidad 1</b> <b>Estructuras de datos básicas y estáticas</b>		<b>Duración en horas</b>	16
<b>Resultado de aprendizaje de la unidad:</b>	Al finalizar la unidad, cada estudiante será capaz de evaluar la eficiencia de las estructuras de datos básicas, como arreglos y listas, para la resolución de problemas computacionales simples, considerando restricciones de tiempo y espacio.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Fundamentos de estructuras de datos: definición y clasificación (básicas vs. avanzadas)</li> <li>2. Arreglos unidimensionales y multidimensionales: implementación, acceso y manipulación</li> <li>3. Listas estáticas: definición, operaciones (inserción, eliminación, búsqueda) y ejemplos prácticos</li> <li>4. Análisis de la eficiencia de arreglos y listas estáticas en la resolución de problemas simples (restricciones de tiempo y espacio)</li> </ol>		

<b>Unidad 2</b> <b>Estructuras de datos dinámicas</b>		<b>Duración en horas</b>	16
<b>Resultado de aprendizaje de la unidad:</b>	Al finalizar la unidad, cada estudiante será capaz de comparar la estructura de datos dinámica más adecuada, como listas enlazadas, pilas, colas y árboles binarios, para la optimización de la manipulación de grandes volúmenes de datos en diferentes contextos de aplicación.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Concepto de estructuras de datos dinámicas y su relación con la gestión de memoria</li> <li>2. Listas enlazadas simples y dobles: implementación, manipulación, recorrido y aplicaciones</li> <li>3. Pilas y colas: implementación, manipulación y aplicaciones</li> <li>4. Árboles binarios de búsqueda (BST): inserción, eliminación, recorrido, búsqueda y aplicaciones</li> <li>5. Comparación de eficiencia entre estructuras de datos dinámicas y estáticas, considerando restricciones de espacio y tiempo</li> </ol>		

<b>Unidad 3</b> <b>Estructuras de datos avanzadas</b>		<b>Duración en horas</b>	16
<b>Resultado de aprendizaje de la unidad:</b>	Al finalizar la unidad, cada estudiante será capaz de evaluar el uso de estructuras de datos avanzadas, como montículos y tablas <i>hash</i> , para el mejoramiento del rendimiento de algoritmos complejos en situaciones que demandan un uso eficiente de recursos computacionales de espacio y tiempo.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Montículos (<i>heaps</i>): implementación, operaciones (inserción, eliminación) y aplicaciones en algoritmos de prioridad</li> <li>2. Tablas <i>hash</i>: fundamentos, técnicas de <i>hashing</i> y métodos para resolución de colisiones</li> <li>3. Eficiencia y aplicabilidad de montículos y tablas <i>hash</i> en la optimización de algoritmos</li> <li>4. Análisis comparativo de estructuras de datos avanzadas en la mejora de rendimiento algorítmico</li> </ol>		

<b>Unidad 4 Grafos y algoritmos de grafos</b>		<b>Duración en horas</b>	16
<b>Resultado de aprendizaje de la unidad:</b>	Al finalizar la unidad, cada estudiante será capaz de evaluar la eficiencia de algoritmos basados en grafos, como los de búsqueda, recorridos y caminos mínimos, para la resolución de problemas de conectividad y optimización en redes y sistemas complejos.		
<b>Ejes temáticos:</b>	<ol style="list-style-type: none"> <li>1. Representación de grafos: matrices de adyacencia y listas de adyacencia</li> <li>2. Árboles generadores mínimos: Prim y Kruskal</li> <li>3. Algoritmos de recorrido: BFS (recorrido en anchura) y DFS (recorrido en profundidad)</li> <li>4. Algoritmos de caminos mínimos: Dijkstra y Floyd-Warshall</li> <li>5. Aplicaciones prácticas de grafos en problemas de conectividad y optimización, considerando eficiencia de algoritmos en términos de tiempo y espacio</li> </ol>		

#### IV. Metodología

##### Modalidad A Distancia - formato virtual

**1. Aprendizaje Basado en Proyectos (ABP):** Esta metodología es ideal para el desarrollo de habilidades prácticas y teóricas en la implementación de estructuras de datos. Los estudiantes pueden trabajar en proyectos que les permitan diseñar, implementar y optimizar estructuras de datos en contextos reales. Dado que la mayor parte de la asignatura es asíncrona, los estudiantes pueden avanzar en sus proyectos a su propio ritmo, recibiendo retroalimentación en las sesiones síncronas.

**2. Aprendizaje Invertido:** En esta metodología, los estudiantes estudian el contenido teórico de manera asíncrona (mediante videos, lecturas o recursos interactivos), lo que les permite aprovechar las sesiones síncronas para resolver dudas, realizar análisis y discutir aplicaciones avanzadas de las estructuras de datos. Esta estrategia maximiza el tiempo síncrono para la colaboración y el análisis crítico, mientras que el contenido básico se aborda de forma individual.

**3. Aprendizaje Basado en Problemas (ABP):** Esta metodología fomenta la resolución de problemas complejos y es perfectamente aplicable a la computación. Los estudiantes pueden enfrentarse a desafíos específicos de estructuras de datos (por ejemplo, cómo optimizar una base de datos grande usando una tabla *hash*) y trabajar en la búsqueda de soluciones. Esto es ideal para sesiones asíncronas, donde los estudiantes tienen tiempo para investigar y aplicar los conceptos, y luego discutir soluciones durante las sesiones síncronas.

**4. Aprendizaje Colaborativo:** Dado que la asignatura incluye componentes de diseño y evaluación de algoritmos, el aprendizaje colaborativo es clave para que los estudiantes trabajen en grupos, compartan enfoques y comparen soluciones a problemas complejos. Esta metodología fomenta la interacción en foros y plataformas en línea para resolver

problemas juntos de manera asíncrona, y en las sesiones síncronas se consolidan y evalúan las soluciones propuestas.

## V. Evaluación

### Sobre la probidad académica

Las faltas contra la probidad académica se consideran infracciones muy graves en la Universidad Continental. Por ello, todo docente está en la obligación de reportar cualquier incidente a la autoridad correspondiente; sin perjuicio de ello, para la calificación de cualquier trabajo o evaluación, en caso de plagio o falta contra la probidad académica, la calificación será siempre cero (00). En función de ello, todo estudiante está en la obligación de cumplir el [Reglamento Académico](#)<sup>1</sup> y conducirse con probidad académica en todas las asignaturas y actividades académicas a lo largo de su formación; de no hacerlo, deberá someterse a los procedimientos disciplinarios establecidos en el mencionado documento.

### Modalidad A Distancia - formato virtual

Rubros	Unidad por evaluar	Semana	Entregable	Instrumento	Peso parcial (%)	Peso total (%)
Evaluación de entrada	Requisito	Primera sesión	Evaluación individual teórica	Prueba objetiva	<b>0</b>	
Consolidado 1 <b>C1</b>	Unidad 1	1-3	Actividades virtuales		15	<b>20</b>
			Trabajo teórico-práctico grupal Implementación y análisis de eficiencia de arreglos y listas aplicados en un problema simple con variaciones en el tamaño de la entrada de datos	Rúbrica de evaluación	85	
<b>Evaluación parcial EP</b>	Unidad 1 y 2	<b>4</b>	Evaluación individual teórico-práctica sobre estructuras de datos básicas y dinámicas con ejercicios prácticos que requieran implementar y comparar arreglos, listas y árboles binarios en la resolución de problemas computacionales	Prueba de desarrollo	<b>25</b>	

<sup>1</sup> Descarga el documento en el siguiente enlace <https://shorturl.at/fhosu>

Consolidado 2 <b>C2</b>	Unidad 3	5-7	Actividades virtuales		15	<b>20</b>
			Evaluación individual teórico-práctica de un problema que optimice un algoritmo utilizando montículos o tablas <i>hash</i>	Prueba de desarrollo	85	
<b>Evaluación final EF</b>	Todas las unidades	<b>8</b>	Trabajo teórico-práctico Proyecto final que incluya la implementación y evaluación de una solución computacional utilizando estructuras de datos avanzadas, optimizando tanto en términos de tiempo como de espacio, para resolver un problema complejo con restricciones definidas	Rúbrica de evaluación	<b>35</b>	
Evaluación sustitutoria*	Todas las unidades <b>Fecha posterior a la evaluación final</b>		Evaluación individual teórico-práctica sobre la selección e implementación de estructuras de datos avanzadas y ejercicios prácticos donde el estudiante diseñe una solución utilizando estructuras como grafos, montículos o tablas <i>hash</i> , optimizando tiempo y espacio	Prueba de desarrollo		

\*Reemplaza la nota más baja obtenida en los rubros anteriores.

**Fórmula para obtener el promedio:**

$$PF = C1 (20 \%) + EP (25 \%) + C2 (20 \%) + EF (35 \%)$$

## VI. Atención a la diversidad

En la Universidad Continental generamos espacios de aprendizaje seguros para todas y todos nuestros estudiantes, en los cuales puedan desarrollar su potencial al máximo. En función de ello, si un(a) estudiante tiene alguna necesidad, debe comunicarla al o la docente. Si el estudiante es una persona con discapacidad y requiere de algún ajuste razonable en la forma en que se imparten las clases o en las evaluaciones, puede comunicar ello a la Unidad de Inclusión de Estudiantes con Discapacidad. Por otro lado,

si el nombre legal del estudiante no corresponde con su identidad de género, puede comunicarse directamente con el o la docente de la asignatura para que utilice su nombre social. En caso hubiera algún inconveniente en el cumplimiento de estos lineamientos, se puede acudir a su director(a) o coordinador(a) de carrera o a la Defensoría Universitaria, lo que está sujeto a la normativa interna de la Universidad.

## VII. Bibliografía

### Básica

Cormen, T. (2013). *Algorithms unlocked*. MIT Press.  
[https://catalogo.continental.edu.pe/permalink/51UCCI\\_INST/1ud8d5s/alma99514398707836](https://catalogo.continental.edu.pe/permalink/51UCCI_INST/1ud8d5s/alma99514398707836)

Cormen, T., Leiserson, C., Rivest, R. y Stein, C. (2022). *Introduction to algorithms* (4.ª ed.). MIT Press.  
[https://catalogo.continental.edu.pe/permalink/51UCCI\\_INST/pbd5ol/cdi\\_proquest\\_ebookcentral\\_EBC6925615](https://catalogo.continental.edu.pe/permalink/51UCCI_INST/pbd5ol/cdi_proquest_ebookcentral_EBC6925615)

Guardati, S. (2016). *Estructuras de datos básicas. Programación orientada a objetos con Java*. Alfaomega Grupo Editor.  
[https://catalogo.continental.edu.pe/permalink/51UCCI\\_INST/1ud8d5s/alma99504698607836](https://catalogo.continental.edu.pe/permalink/51UCCI_INST/1ud8d5s/alma99504698607836)

Tymoschuk, K., Guzman, A. y Fritelli, V. (2020). *Algoritmos y Estructuras de Datos* (2.ª ed.). Jorge Sarmiento Editor y Universitas Libros.  
<https://ebookcentral.continental.elogim.com/lib/unicont/detail.action?docID=6780950>

### Complementaria

Fager, J., Pantoja, W., Villacrés, M., Páez, L., Ochoa, D. y Cuadros-Vargas, E. (2014). *Estructura de datos*. Iniciativa Latinoamericana de Libros de Textos Abiertos (LATIn).

## VIII. Recursos digitales

Estructuras de datos y algoritmos. (s.f.). *Curso de estructuras de datos y algoritmos*. GitHub.  
<https://jmlon.github.io/Estructuras-Algoritmos/>

Los Algoritmos . (s.f). *Los-algoritmos.com*. <https://the-algorithms.com/es>

Visualgo. (s.f.). *Visualgo: visualising data structures and algorithms through animation*.  
<https://visualgo.net/en>