|  | **PROGRAMA EDUCATIVO:**  **LICENCIATURA EN INGENIERÍA EN TECNOLOGÍAS DE LA INFORMACIÓN E INNOVACIÓN DIGITAL**  **EN COMPETENCIAS PROFESIONALES** |  |
| --- | --- | --- |

**PROGRAMA DE ASIGNATURA: ESTRUCTURA DE DATOS** \_\_\_\_\_\_\_\_\_\_\_\_ **CLAVE:**\_\_\_\_**ESD**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| Propósito de aprendizaje de la Asignatura | | El estudiante implementará algoritmos que utilicen estructuras de datos para mejorar el rendimiento, calidad y mantenibilidad de los sistemas informáticos, a través de la modularidad y escalabilidad que satisfagan las necesidades de la organización. | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Competencia a la que contribuye la asignatura | | Desarrollar soluciones tecnológicas multiplataforma de software web y móvil utilizando programación orientada a objetos, frameworks, bases de datos, estándares de calidad y diseño para resolver problemas del sector productivo, con un enfoque de inclusión, compromiso con la responsabilidad social, equidad social y de género, excelencia, vanguardia, innovación social e interculturalidad. | | | | |
| Tipo de competencia | Cuatrimestre | | Créditos | Modalidad | Horas por semana | Horas Totales |
|
| **Específica** | **4** | | **4.6875** | **Escolarizada** | **5** | **75** |

| **Unidades de Aprendizaje** | **Horas del Saber** | **Horas del Saber Hacer** | **Horas Totales** |
| --- | --- | --- | --- |
|  |  |  |
| 1. Conceptos básicos de estructuras de datos orientadas a objetos | 10 | 10 | 20 |
| 1. Estructuras de datos básicas | 10 | 25 | 35 |
| 1. Estructuras de datos avanzadas | 10 | 10 | 20 |
| **Totales** | **30** | **45** | **75** |

| **Funciones** | **Capacidades** | **Criterios de Desempeño** |
| --- | --- | --- |
| Diseñar soluciones tecnológicas web y móviles con base en un análisis, utilizando algoritmos y estructuras de datos eficientes, patrones de diseño, base de datos, experiencia de usuario, metodologías y estándares con el fin de satisfacer los requerimientos considerados por la empresa. | Realizar el análisis de los requerimientos del software utilizando estándares y técnicas de recopilación de información para el levantamiento, la validación y la especificación de los requerimientos que representen las necesidades del cliente. | Integrar un reporte técnico sobre el análisis de los requerimientos de software que documente:  - La identificación de flujos de datos del problema.  - Identificación de eventos(casos de uso).  - Identificar y determinar los requerimientos, tanto funcionales como no funcionales.  - Identificación del alcance.  - Validación de requisitos para asegurar su precisión, completitud y consistencia.  - Verificar que los requisitos sean factibles de implementar con las tecnologías disponibles.  - Identificar y gestionar los riesgos asociados a los requisitos. |
| Realizar el diseño funcional y arquitectónico de la aplicación utilizando herramientas, estructuras de datos y patrones de diseño para definir la representación física y lógica de la solución. | Integrar un reporte técnico sobre el diseño del software utilizando diagramas UML que documente:  - La definición de las funcionalidades de la aplicación y su organización en módulos independientes.  - La descripción de los pasos que los usuarios seguirán para realizar las diferentes tareas dentro de la aplicación.  - La especificación de las reglas y lógica que gobiernan el comportamiento de la aplicación(Reglas del Negocio).  - Arquitectura y Patrones de diseño.  - La identificación de los componentes y servicios que conforman la aplicación y cómo interactúan entre sí.  - La selección de las tecnologías y frameworks que se utilizarán para desarrollar la aplicación.  - La selección de la estructura de datos adecuada para representar y organizar la información eficientemente en la aplicación. |

**UNIDADES DE APRENDIZAJE**

| Unidad de Aprendizaje | 1. Conceptos básicos de estructuras de datos orientadas a objetos | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Propósito esperado | El estudiante desarrollará habilidades para implementar estructuras de datos utilizando tipos de datos abstractos, recursividad, arreglos, clases parametrizadas y tipos genéricos con el fin de optimizar la gestión y manipulación de información en entornos de programación. | | | | | |
| **Tiempo Asignado** | **Horas del Saber** | 10 | **Horas del Saber Hacer** | 10 | **Horas Totales** | 20 |

| **Temas** | **Saber**  **Dimensión Conceptual** | **Saber Hacer**  **Dimensión Actuacional** | **Ser y Convivir**  **Dimensión Socioafectiva** |
| --- | --- | --- | --- |
| Tipo de datos abstractos | Describir los tipos de datos abstractos. | Codificar tipos de datos abstractos. | Desarrollar el razonamiento crítico, lógico y matemático a través de la resolución de problemas de programación de manera individual y/o colaborativa, utilizando arreglos y recursividad desde la perspectiva de la programación orientada a objetos. |
| Recursividad | Definir recursividad. |  |
| Identificar los tipos de recursividad. | Implementar recursividad para la solución de problemas. |
| Arreglos | Distinguir el concepto y características de arreglos. |  |
| Identificar los elementos para la declaración, creación y manipulación de arreglos. | Programar ejemplos de aplicación  de arreglos para la solución de problemas. |
| Clases parametrizadas y tipos genéricos | Describir el funcionamiento de los parámetros de tipo y su relación con la creación de instancias. | Instanciar estructuras genéricas de distintos tipos. |

| **Proceso Enseñanza-Aprendizaje** | | | |
| --- | --- | --- | --- |
| Métodos y técnicas de enseñanza | Medios y materiales didácticos | Espacio Formativo | |
| **Aula** |  |
| Prácticas en laboratorio  Estudio de casos  Equipo colaborativo | Laboratorio de cómputo  Uso de Internet  Proyector o pantalla  Pizarrón  Bibliografía  Herramientas de programación | **Laboratorio / Taller** | X |
|  |  | **Empresa** |  |

| **Proceso de Evaluación** | | |
| --- | --- | --- |
| Resultado de Aprendizaje | Evidencia de Aprendizaje | Instrumentos de evaluación |
| Los estudiantes utilizan diversas técnicas y herramientas de programación, para implementar arreglos, tipos de datos abstractos y clases parametrizadas. | A partir de un portafolio de evidencias implementar estructuras de datos y lo documenta en un reporte técnico que incluya:  - Introducción: con la descripción general del problema resuelto.  - Propuesta de solución: Descripción de los tipos de datos abstractos, arreglos, clases parametrizadas, tipos genéricos utilizados, recursividad y la justificación de su uso.  - Pruebas realizadas.  - Conclusiones  - Referencias bibliográficas. | Ejercicios prácticos  Guía de observación |

| Unidad de Aprendizaje | II. Estructuras de datos básicas | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Propósito esperado | El estudiante implementará estructuras de datos básicas, algoritmos de ordenamiento y búsqueda, así como analizar la complejidad de los algoritmos, para diseñar soluciones óptimas en entornos de programación. | | | | | |
| **Tiempo Asignado** | **Horas del Saber** | 10 | **Horas del Saber Hacer** | 25 | **Horas Totales** | 35 |

| **Temas** | **Saber**  **Dimensión Conceptual** | **Saber Hacer**  **Dimensión Actuacional** | **Ser y Convivir**  **Dimensión Socioafectiva** |
| --- | --- | --- | --- |
| Listas | Definir el concepto de estructura de datos dinámica. |  | Desarrollar el razonamiento crítico y lógico a través de la resolución de problemas de programación de manera individual y/o colaborativa, a través de la implementación de estructuras de datos básicas desde la perspectiva de la programación orientada a objetos. |
| Describir las características de las estructuras de datos dinámicas. |  |
| Describir la metodología de codificación de estructuras de datos dinámicas. |  |
| Definir el concepto de listas. |  |
| Diferenciar los tipos de listas (simples, dobles y circulares), sus componentes y su función práctica en la resolución de problemas. | Programar ejemplos de aplicación utilizando distintos tipos de listas (simples, dobles y circulares) para la resolución de problemas. |
| Pilas | Definir las operaciones que se realizan con listas. |  |
| Definir el concepto de pilas. |  |
| Comprender los componentes de las pilas y su función práctica en la resolución de problemas. |  |
| Definir las operaciones que se realizan con pilas. | Programar ejemplos de aplicación utilizando pilas para la resolución de problemas. |
| Colas | Definir el concepto de colas. |  |
|  | Comprender los componentes de las colas y su función práctica en la resolución de problemas. |  |
| Definir las operaciones que se realizan con colas. | Programar ejemplos de aplicación utilizando colas para la resolución de problemas. |
|  | Describir los algoritmos básicos de ordenamiento. |  |
| Describir los algoritmos básicos de búsqueda. |  |
| Describir los algoritmos básicos de recorrido. |  |
| Comprender la complejidad algorítmica de búsqueda y ordenamiento. | Codificar algoritmos de ordenamiento, búsqueda y recorrido con la estructura de datos adecuada referente al caso de estudio. |

| **Proceso Enseñanza-Aprendizaje** | | | |
| --- | --- | --- | --- |
| Métodos y técnicas de enseñanza | Medios y materiales didácticos | Espacio Formativo | |
| **Aula** |  |
| - Prácticas en laboratorio  - Estudio de casos  - Simulación | Laboratorio de cómputo  Uso de Internet  Proyector o pantalla  Pizarrón  Bibliografía  Simuladores interactivos  Herramientas de programación | **Laboratorio / Taller** | X |
|  |  | **Empresa** |  |

| **Proceso de Evaluación** | | |
| --- | --- | --- |
| Resultado de Aprendizaje | Evidencia de Aprendizaje | Instrumentos de evaluación |
| Los estudiantes elaboran programas que contengan estructuras de datos básicas, métodos de búsqueda y ordenamiento para la resolución de casos prácticos. | A partir de un caso práctico implementar estructuras de datos básicas y lo documenta en un reporte técnico que incluya:  - Introducción: con la descripción general del problema resuelto.  - Diseño de la propuesta de solución: Descripción de las estructuras de datos básicas, algoritmos de ordenamiento y búsqueda utilizados y análisis de la complejidad de los algoritmos utilizados.  - Pruebas realizadas.  - Conclusiones  - Referencias bibliográficas. | Estudio de casos  Rúbrica |

| Unidad de Aprendizaje | III. Estructuras de datos avanzadas | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Propósito esperado | El estudiante implementará estructuras de datos avanzadas para diseñar soluciones óptimas en entornos de programación. | | | | | |
| **Tiempo Asignado** | **Horas del Saber** | 10 | **Horas del Saber Hacer** | 10 | **Horas Totales** | 20 |

| **Temas** | **Saber**  **Dimensión Conceptual** | **Saber Hacer**  **Dimensión Actuacional** | **Ser y Convivir**  **Dimensión Socioafectiva** |
| --- | --- | --- | --- |
| Árboles | Definir el concepto de árboles. |  | Desarrollar el razonamiento crítico y lógico a través de la resolución de problemas de programación de manera individual y/o colaborativa, a través de la implementación de estructuras de datos avanzadas desde la perspectiva de la programación orientada a objetos. |
| Comprender los componentes de árboles y su función práctica en la resolución de problemas. |  |
| Definir las operaciones que se realizan con árboles. |  |
| Diferenciar los algoritmos de recorrido en árboles. | Programar ejemplos de aplicación utilizando árboles para la resolución de problemas. |
| Diccionarios | Definir el concepto de diccionario como estructura de datos. |  |
| Comprender los componentes de diccionarios y su función práctica en la resolución de problemas. |  |
| Definir las operaciones que se realizan con diccionarios. | Programar ejemplos de aplicación utilizando diccionarios para la resolución de problemas. |
| Conjuntos | Definir el concepto de conjuntos como estructura de datos |  |
| Comprender los componentes de conjuntos y su función práctica en la resolución de problemas. |  |
| Definir las operaciones que se realizan con conjuntos. | Programar ejemplos de aplicación utilizando conjuntos para la resolución de problemas. |

| **Proceso Enseñanza-Aprendizaje** | | | |
| --- | --- | --- | --- |
| Métodos y técnicas de enseñanza | Medios y materiales didácticos | Espacio Formativo | |
| **Aula** |  |
| Prácticas en laboratorio  Estudio de casos  Equipo colaborativo | Laboratorio de cómputo  Uso de Internet  Proyector o pantalla  Pizarrón  Bibliografía  Herramientas de programación | **Laboratorio / Taller** | X |
|  |  | **Empresa** |  |

| **Proceso de Evaluación** | | |
| --- | --- | --- |
| Resultado de Aprendizaje | Evidencia de Aprendizaje | Instrumentos de evaluación |
| Los estudiantes elaboran programas que contengan estructuras de datos avanzadas y métodos de búsqueda para la resolución de casos prácticos. | A partir de un caso práctico implementar estructuras de datos avanzadas y lo documenta en un reporte técnico que incluya:  - Introducción: con la descripción general del problema resuelto.  - Diseño de la propuesta de solución: Descripción de las estructuras de datos utilizadas (árboles, diccionarios y/o conjuntos) y métodos de búsqueda. Incluyendo la justificación del diseño propuesto.  - Pruebas realizadas.  - Conclusiones  - Referencias bibliográficas. | Proyecto grupal  Rúbrica |

| **Perfil idóneo del docente** | | |
| --- | --- | --- |
| **Formación académica** | **Formación Pedagógica** | **Experiencia Profesional** |
| Ingeniero en  informática, Ingeniería en ciencias de la  computación o  un campo  relacionado. | Manejo de herramientas didácticas para  enseñanza-aprendizaje, de evaluación,  habilidades tecnológicas para la educación, uso de simuladores adecuados a la signatura | Experiencia práctica en el desarrollo de software y la implementación de estructuras de datos en entornos reales. |

| **Referencias bibliográficas** | | | | | |
| --- | --- | --- | --- | --- | --- |
| Autor | Año | Título del documento | Lugar de publicación | Editorial | ISBN |
| Miguel Hernández Bejarano, Luis Eduardo Baquero Rey | 2021 | Estructuras de datos: Fundamentación práctica | Bogotá Colombia | Ediciones de la U | 9587922719, 9789587922714 |
| Enrique Gómez Jiménez | 2023 | Estructura de datos: Un enfoque con Python, java y C++ | Bogotá Colombia | Alpha Editorial | 9587789016, 9789587789010 |
| V. M. de la Cueva Hernández | 2020 | Estructuras de datos y algoritmos fundamentales | México | Editorial Digital del Tecnológico de Monterrey | 6075016228, 9786075016221 |
| Silvia Guardati | 2015 | Estructuras de Datos Básicas Programación orientada a objetos con Java | México | Alfaomega | 9786076225004, 6076225009 |
| Manning Publications | 2024 | Grokking Algorithms, Second Edition | USA | Manning Publications | 1633438538, 978-1633438538 |
|  |  |  |  |  |  |

| **Referencias digitales** | | | |
| --- | --- | --- | --- |
| Autor | Fecha de recuperación | Título del documento | Vínculo |
| Oracle | 15-mayo-2024 | JNI Types and Data Structures | https://docs.oracle.com/javase/8/docs/technotes/ |
|  |  |  |  |