 

**SÍLABO 2023-2**

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS

| ASIGNATURA | **ESTRUCTURAS DE DATOS Y ALGORITMOS** |
| --- | --- |
| ÁREA | INGENIERÍA DE SOFTWARE |
| CÓDIGO | 1492 |
| NIVEL | QUINTO |
| CARÁCTER | OBLIGATORIO |
| REQUISITO | PROGRAMACIÓN ORIENTADA A OBJETOS |
| CRÉDITOS | TRES (3) |
| HORAS DE TEORÍA SEMANAL | UNA (1) |
| HORAS DE PRÁCTICA SEMANAL | CUATRO (4) |
| IDIOMA | ESPAÑOL |
| PROFESOR(ES) | NINA HANCO, HERNÁN (Coordinador)  GUTIERREZ CARDENAS JUAN MANUEL,  MAYHUA QUISPE ANGELA GABRIELA. |

1. **SUMILLA**

La asignatura presenta las estructuras de datos y los algoritmos requeridos en el desarrollo de software. Los temas tratados son los siguientes: algoritmos de búsqueda y de ordenamiento, estructuras de datos fundamentales: arreglos, listas, pilas, colas, árboles y grafos y los algoritmos que operan con estas estructuras, recursividad, así como los fundamentos del análisis de algoritmos.

1. **OBJETIVOS GENERALES**

* Mejorar la capacidad de diseñar algoritmos y programas de calidad, seleccionando y adaptando las estructuras de datos más adecuados al problema que se desea resolver.
* Implementar estructuras de datos y algoritmos usando programación orientada a objetos.

1. **OBJETIVOS ESPECÍFICOS**
2. Conocer las estructuras de datos fundamentales estudiándolas como tipos de datos abstractos.
3. Conocer los algoritmos más importantes que se utilizan para la manipulación de las estructuras de datos fundamentales.
4. Conocer y aplicar las operaciones fundamentales en estructuras lineales, tales como los arreglos y las matrices.
5. Conocer los conceptos relacionados con el análisis de algoritmos.
6. Conocer y desarrollar algoritmos recursivos.
7. Conocer los algoritmos de ordenamiento más importantes.
8. Conocer algunos principios generales y técnicas de diseño de algoritmos.
9. **PROGRAMA ANALÍTICO**

**UNIDAD 1: INTRODUCCIÓN A LOS ALGORITMOS Y MANEJO DE ARREGLOS 20 horas**

**PRIMERA SEMANA**

Concepto de algoritmo. Notación algorítmica. Clases, objetos y métodos. Estructuras de datos.

**SEGUNDA SEMANA**

Análisis de algoritmos de forma teórica yexperimental. Complejidad del peor caso y del mejor caso. Complejidad en tiempo y en espacio. Notación O grande.

**TERCERA SEMANA**

Algoritmos elementales para arreglos unidimensionales: inserción, búsqueda, eliminación.

Algoritmos de búsqueda: búsqueda secuencial y binaria: Análisis de complejidad

**CUARTA SEMANA**

Algoritmos de ordenamiento y su análisis: Método de la Burbuja, Inserción y Selección. Prueba de corrección. Arreglos bidimensionales.

**UNIDAD 2: ESTRUCTURAS DE DATOS DINÁMICAS LINEALES**

**20 horas**

**QUINTA SEMANA**

Listas Enlazadas: operaciones de recorrido, búsqueda, inserción y eliminación**.**

**SEXTA SEMANA**

Listas Enlazadas. Implementación y análisis**.**

**SÉPTIMA SEMANA**

**Cola**: operaciones de encolar y desencolar. Representación. Colas prioridad.

**OCTAVA SEMANA**

**Pila**: operaciones de apilar y desapilar. Representación. Aplicaciones de pilas.

**UNIDAD 3: RECURSIVIDAD 10 horas**

**NOVENA SEMANA**

**Recursividad**. Algoritmos recursivos. Caso base y caso recurrente. Llamadas recursivas.

**DÉCIMA SEMANA**

Algoritmos recursivos de ordenamiento: Quicksort y MergeSort. Aplicaciones de recursividad.

**UNIDAD 4: GRAFOS Y ÁRBOLES 25 horas**

**DÉCIMO PRIMERA SEMANA**

Grafos, definición.Representación. Recorrido en profundidad y en anchura.

**DÉCIMO SEGUNDA SEMANA**

Problema de árboles de expansión de costo mínimo. Algoritmo de Kruskal.

**DÉCIMO TERCERA SEMANA**

Algoritmo de Prim. Algoritmo de Dijkstra

**DÉCIMO CUARTA SEMANA**

Árbol Binario. operaciones de búsqueda, inserción y eliminación. Recorridos de árboles.

**DÉCIMO QUINTA SEMANA**

Árbol Binario de Búsqueda: operaciones de búsqueda, inserción y eliminación. Representación.

**DECIMO SEXTA SEMANA**

Retroalimentación del aprendizaje.

**DECIMOSÉPTIMA SEMANA**

Entrega final de notas

**V. METODOLOGÍA DE ENSEÑANZA - APRENDIZAJE**

La Universidad de Lima ha diseñado la **Metodología IATC** para orientar el proceso de enseñanza-aprendizaje y asegurar el logro de los objetivos de la siguiente manera:

* Impacto: motivar y generar curiosidad. Presentar objetivos y agenda.
* Adquisición del aprendizaje: transmitir el conocimiento con estrategias innovadoras. Promover la interacción.
* Transferencia de lo aprendido: desarrollar actividades significativas. Utilizar estrategias y técnicas didácticas.
* Cierre del aprendizaje: concluir sobre el aprendizaje. Reflexionar sobre el logro del objetivo.

Además, la asignatura se desarrollará tomando en cuenta las siguientes metodologías:

1. Revisión previa de los temas de clase, por parte de los alumnos, bajo el concepto de aula invertida.
2. Desarrollo participativo de las sesiones y temas del curso, con enfoque teórico-práctico, propiciándose que los alumnos desarrollen ejercicios individuales y grupales en clase.
3. Autoevaluación de los alumnos, mediante la asignación de ejercicios selectos para resolución fuera de aula y posterior validación contra la solución del grupo.
4. Consolidación de los conceptos, mediante la presentación de casos reales de aplicación, generando discusión sobre otros escenarios donde podrían aplicarse.

Adicionalmente, los profesores brindarán asesorías a los alumnos según el rol establecido, para atender consultas y resolver dudas sobre la asignatura.

1. **SISTEMA DE EVALUACIÓN**

Para el sistema de evaluación, esta asignatura es de tipo:

| **Teórico práctico** |
| --- |

La nota final de la asignatura (NF) es el promedio ponderado de las notas obtenidas en el proceso de evaluación continua (EC):

La nota de EC comprende:

| **Semana** | **Tipo de evaluación** | **Peso (%)** | **Objetivo que se evalúa** |
| --- | --- | --- | --- |
| 5 | Ejercicio individual (EI) | 20 | Objetivos: 1 y 4 |
| 9 | Ejercicio individual (EI) | 25 | Objetivos: 2 ,3 y 6 |
| 13 | Proyecto (PR) | 20 | Objetivos: 3 y 5 |
| 15 | Ejercicio individual (EI) | 35 | Objetivos: 7 |
|  |  |  |  |

**VII. REFERENCIAS**

**OBLIGATORIA**

1. Cormen, T. H. (2009). *Introduction to algorithms*. Cambridge, Mass: MIT Press. Código Biblioteca U. Lima: 005.1 C775
2. Mehlhorn, K., y Sanders, P. (2010). Algorithms and Data Structures: The Basic Toolbox. Springer.

**COMPLEMENTARIA**

1. Kleinberg, J., y Tardos, E. (2013). *Algorithm design*. Boston: Pearson/Addison-Wesley.
2. Hubbard, J. R. (2007). *Data Structures with Java*. Schaum´s Edition. (2ª ed.)
3. Cairo Battistutti, O. (2003). *Metodología de la Programación; algoritmos, diagramas de flujo y programas*. (2da ed.). Código Biblioteca U. Lima: 005.1/c16.
4. Camacho Fernández, D. (2003). *Programación, algoritmos y ejercicios resueltos en java*. Código Biblioteca U. Lima: 005.73/c22
5. Drozdek, A. (2004). *Data structures and algorithms in Java*. Cambridge, Mass: Thomson Course Technology. Código Biblioteca U. Lima: 005.133J D96.
6. Franch Gutierrez, X. (2002). *Estructura de Datos; diseño e implementación*. (4ta ed.). Código Biblioteca U. Lima: 005.73/f81
7. Fernández,J., Oliver, F.J., Sánchez, J.L. (1992). *Algoritmos; problemas resueltos y comentados*. Código Biblioteca U. Lima 004.0151 F385.
8. Joyanes Aguilar, L. (2008). *Fundamentos de programación: algoritmos, estructura de datos y objetos* (4ta ed.). Madrid, McGraw-Hill. Código Biblioteca U. Lima 005.1 J79F 2008
9. Joyanes Aguilar, L. (2005). *Programación en C; Metodología, algoritmo y estructura de datos*. Madrid, McGraw-Hill. Código Biblioteca U. Lima: 005.133c/j79p
10. Villalobos, S. J. A. (2008). *Introducción a las estructuras de datos: Aprendizaje activo basado en casos : un enfoque moderno usando Java, UML, objetos y eclipse*. Bogotá: Pearson Educación. Código Biblioteca U. Lima 005.73 V66 2008
11. Weiss, M. A. (1995). *Estructuras de Datos y algoritmos*. Delaware, Editorial Addison -Wesley Iberoamericana. Código Biblioteca U. Lima: 005.73/w42