

LICENCIATURA EN
**CIENCIA
DE DATOS**

PROGRAMA

189 PROGRAMACION AVANZADA

LIC. FELIPE MORALES

1. Cantidad de horas semanales y totales

La materia tiene una carga semanal de 6 horas en total. 4 horas en formato presencial. Adicionalmente se puede complementar con encuentros virtuales en formato virtual.

2. Nombres de las/los integrantes del equipo docente

Profesor Lic. Felipe Morales.

3. Fundamentación

En esta asignatura vamos a hacer foco en un nuevo paradigma de programación: programación orientada a objetos. Este nuevo estilo de programación ha revolucionado el desarrollo de software en todos los órdenes, potenciando el mantenimiento, extensión y reutilización. Como futuros profesionales en el terreno de la informática es fundamental el manejo de este paradigma. Desde el punto de vista de las concepciones epistemológicas la propuesta se basa principalmente en el enfoque empírico-deductivo.

La finalidad de esta asignatura es continuar con el proceso de formación del estudiante en el terreno de la programación, propiciando un nuevo enfoque propio de este paradigma de programación.

4. Programa sintético

En términos generales los temas principales que constituyen el contenido central de la asignatura, estará centrado en dos ejes, en primer lugar en la **enseñanza de programación orientada a objetos**, para posteriormente abordar el **diseño de aplicaciones orientadas a objetos**.

5. Objetivos

Los objetivos propuestos son **enseñanza de programación orientada a objetos**, y **diseño de aplicaciones orientadas a objetos**.

6. Propósitos de la enseñanza

Como propósitos podemos mencionar la importancia de la adquisición de conocimientos en el terreno de la programación orientada a objetos, tanto desde un punto de vista práctico como teórico. Es importante mencionar que en la actualidad todos los desarrollos de software utilizan este tipo de paradigma de programación.

7. Contenidos

Programación avanzada:

Introducción al Paradigma Orientado a Objetos. El progreso de la abstracción. Metas del Paradigma Orientado a Objetos. Conceptos Básicos de la Programación Orientada a Objetos. Clase. Atributos. Operaciones. Interfaces. Objetos Metaclases. Diseño de Aplicaciones OO. Relaciones entre clases y objeto Herencia y Polimorfismo. Herencia Simple, Múltiple, de Interfaz y de Implementación Polimorfismo. Reuso. Sobrecarga. Sobreescritura. Variables Polimórficas. Ejemplos e implementación en lenguajes de programación Orientados a Objetos.

8. Bibliografía y recursos audiovisuales

Sobre este apartado incluimos:

Coad, P; Yourdon, E. (1991). Prentice-Hall International editions, ed. Object oriented Design

http://www.edu4java.com/es/conceptos/clases-instancias-prototipos-herencia_poo.html

https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Details_of_the_Object_Model

http://kuliahkaryawan.widyakartika.ac.id/IT/en/3077-2963/object-oriented-programming_3289_kuliahkaryawan-widyakartika.html

9. Metodología

Desde el punto de vista metodológico se procederá a:

- Presentación del Tema
- Resolución de problemas teóricos/prácticas en forma guiada
- Resolución de ejercicios a cargo del estudiante
- Revisión y seguimiento constante a cargo del docente

10. Uso del campus virtual e integración de TIC en la propuesta pedagógica

Todo el material, recursos, interacciones, foros, y demás se realizará sobre el **campus virtual**, constituyéndose de esta manera en el mecanismo de registro y seguimiento de avance académico del estudiante.

11. Evaluación

A. Requisitos de aprobación:

- a. Evaluación continua
- b. 75% de asistencia obligatoria (*)

B. Criterios de evaluación

- a. 2 (dos) evaluaciones parciales
- b. Aprueban con nota de 4 en adelante en casa evaluación. Promocionan aquellos que aprueban cada instancia de evaluación con 6 o más.

C. Formatos de la evaluación de las distintas instancias

- a. Primera evaluación en formato de examen parcial individual.
- b. Segunda evaluación en formato de presentación de trabajo grupal (la evaluación será individual). El trabajo deberá ser presentado, aprobado y defendido.

(*Atendiendo a situaciones y casos particulares.

12.Cronograma de actividades / Planificación de clases

A modo de cronograma preliminar se presenta la siguiente planificación de clases. La misma puede estar sujeta a cambios que serán debidamente difundidos.

Clase	Fecha	Unidad	Cronograma de Actividades	Cross
1	27/03/2023	I	Presentación de la materia. Introducción al Paradigma Orientado a Objetos. Pilares del Paradigma Orientado a Objetos. Encapsulamiento, abstracción, polimorfismo y herencia. Ejemplos en Python	Encuestas permanentes Uso de IDEs Cuenta en GitHub Uso de git
2	03/04/2023		Conceptos Básicos de la Programación Orientada a Objetos. Clases. Propiedades. Métodos. Objetos. IDE. Ejemplos concretos	
3	10/04/2023		Taller de Programación orientada a objetos (Python y otro lenguaje a elección por el estudiante)	
4	17/04/2023		Interfaces. Objetos Metaclasses. Decoradores	
5	24/04/2023		Workshop de Programación. Retrospectiva	
6	01/05/2023		Taller de Pair Programming. Lecciones aprendidas	
7	08/05/2023		1er Evaluación parcial. Muestra de Evaluación. Explicación de Evaluación.	
8	15/05/2023	II	Muestra de Evaluación. Explicación de Evaluación. Estadísticas de la Evaluación	Aprendizaje autoguiado Trabajo en equipo Prácticas en Python Ejemplos en otros lenguajes
9	22/05/2023		Diseño de Aplicaciones OO. (Análisis, Diseño, Programación). Notación UML. Diagramas.	
10	29/05/2023		Patrones de diseño. Patrones más usados.	
11	05/06/2023		Patrones de diseño. Definición de Trabajo Práctico Integrador	
12	12/06/2023		FERIADO NACIONAL	
13	19/06/2023		Muestra de Evaluación. Explicación de Evaluación. Clase de consulta y repaso.	
14	26/06/2023		Presentaciones y defensas de Trabajos Prácticos Integradores	
15	03/07/2023		Presentaciones y defensas de Trabajos Prácticos Integradores	
16	10/07/2023		Cierre de notas. Cierre de Cursada y de Actas	
	17/07/2023			