



Programa de la Asignatura: PROGRAMACIÓN ORIENTADA A OBJETOS		Codigo: <b>IF008</b>
Departamento: Informática	Sede: Esquel	

Profesor Responsable: AGUADO, MATIAS ANTONIO

Regimen de Cursado: Segundo Cuatrimestre

Carga Horaria:

Total	Sem. Teóricos	Total Teóricos	Sem. Prácticos	Total Prácticos	Sem. Teo/Prácticos	Total Teo/Prácticos
120	4	60	4	60	0	0

Carga Horaria Teóricos/Teóricos-Prácticas:  
 Dias Lunes de 18:00 a 22:00  
 Dias Miercoles de 18:00 a 22:00

Carga Horaria Prácticas:  
 Dias de a

**Asignaturas Correlativas:**

Código	Nombre	Para la/s Carrera/s
IF006	ALGORÍTMICA Y PROGRAMACIÓN II	APU y Lics en Informática

**I. Objetivos de la Asignatura:**

Comprender y aprender el Paradigma Orientado a Objetos. Entender el contexto histórico en el que surge el Paradigma Orientado a Objetos y con qué objetivo. Aplicar los conceptos pensando en objetos desde la modelización, el diseño hasta la programación. Comprender y aplicar los conceptos de Abstracción y Modelización, Encapsulamiento/Ocultamiento, Herencia, Polimorfismo, Instanciación, Sobrecarga (Overloading) y Sobreescritura (Overriding), Clases: Reusabilidad, Eficiencia y Portabilidad, Comportamiento y Estado de los Objetos. Lenguajes Orientados a Objetos: Principales características y particularidades. Aprender a programar en JAVA respetando los fundamentos de la POO. Criterios de selección y uso de Entornos Integrados de Desarrollo (IDEs), configuración y principales funcionalidades. Comprender la organización, documentación y uso de las librerías de clases existentes más utilizadas para el desarrollo de aplicaciones OO con Java. Construir aplicaciones con Interfaz Gráfica de Usuario, manejo de eventos y acceso a Base de Datos, mapear entidades de BD existentes con clases java y desarrollar lógica de negocio para la resolución de consignas sobre aplicaciones de uso real.

**II.1 Contenidos Mínimos:**

- \* Paradigma de programación orientada a objetos.
- \* Abstracción y Encapsulamiento en Programación orientada a objetos.
- \* Objetos. Clases e instancias.
- \* Herencia. Jerarquía de clases.
- \* Polimorfismo.
- \* Lenguajes y entornos de desarrollo.
- \* Interfaces gráficas. Eventos.
- \* Nociones de concurrencia y paralelismo.



Programa de la Asignatura: PROGRAMACIÓN ORIENTADA A OBJETOS	Codigo: <b>IF008</b>
Departamento: Informática	Sede: Esquel

## II. 2 Programa Analítico:

### 1) Introducción al Paradigma Orientado a Objetos

Introducción al Desarrollo de Software, breve reseña histórica, desde la programación estructurada a la programación orientada a objetos. Del Spaguetti code a los principios de Modularización, Reutilización, Alta Cohesión y Bajo Acoplamiento. Manejo del costo y la complejidad del software. Crisis del software. Calidad y Gestión del cambio. Lineamientos de la Ingeniería del Software sobre el Paradigma Orientado a Objetos

### 2) Abstracción y Modelado de Objetos

Abstracción: Identificar los aspectos relevantes del dominio de un problema, Identificar Clases, Objetos y sus relaciones, identificar atributos y comportamiento de los objetos. Modelización: Una forma de manejar la complejidad y simplificar la realidad (Dominio de la aplicación/problema). Creación de maquetas y prototipos para representar un modelo de la realidad. Experimentación práctica con problemas sencillos. Percepción e identificación de Objetos. (Juego de Ajedrez, Texto Literario)

### 3) Programación Orientada a Objetos

Introducción al Lenguaje de Programación Java. Metodología de Programación, creación y desarrollo de programas en Java. Biblioteca de Clases Java. Creación de Proyectos y paquetes. Creación de clases, concepto de visibilidad (public, protected, private), definición y tipos de atributos(variables), tipos predefinidos de datos, creación de métodos getters y setter, creación de métodos definidos por el usuario, creación de constructores (sobrecarga), Instanciación de objetos: Identidad, Estado y Comportamiento. Uso de concepto de this y static. Encapsulamiento/Ocultamiento: modificadores de acceso para métodos y atributos. Instanciación de objetos. Implementación de Juego de Ajedrez. Buenas prácticas de la programación.-

### 4) Herencia y Relaciones entre Clases

Concepto y características de la Herencia en la POO. Ejemplos y representación de Herencia en el mundo real. Reutilización basada en la herencia. Contextos y Utilidad de la Herencia en la POO. Generalización y Especialización de clases, Super clases y subclases. Constructores en contexto de Herencia, concepto super(). Clases abstractas y métodos abstractos, Interfaces, uso y aplicación. Implementación de clase abstracta y método abstracto. Juego de Ajedrez. Relaciones de asociación, composición y dependencia entre clases. Ejemplos prácticos.

### 5) Polimorfismo

Entender el concepto de Polimorfismo. Contexto, Utilidad y precondiciones para implementar Polimorfismo. Uso de sobrecarga y Sobreescritura (Overloading y Overriding) en contexto de Polimorfismo. Uso de clases y métodos abstractos en contexto de Polimorfismo. Concepto y Ejemplos de enlace estático y enlace dinámico Aplicación práctica: Implementar el movimiento de las piezas de ajedrez.

### 6) Tratamiento de Errores y Excepciones

Concepto del Manejo de errores. Introducción y lineamientos de la Ingeniería de Software. Definición y condiciones de error. Definición de Excepción. Jerarquía de Excepciones en Java. Principales métodos de la clase Exception (getMessage(), getStackTrace(), printStackTrace()). Inspección de código para detección de errores. Validación mediante pruebas. Uso de Debugging. Manejo estructurado de excepciones, bloques try, catch y finally.

### 7) Acceso a Datos con JDBC

Introducción a Java Database Connectivity (JDBC). Revisión de la API y sus principales clases. Concepto de Driver JDBC. Explicación y uso de la clase DriverManager, Connection, Statement, ResultSet. Explicación y uso de Class.forName() para cargar Driver JDBC. Criterios y escenarios para selección de tipo de Sentencias (Statement, PreparedStatement, CallableStatement). Implementar programa Java con acceso a datos e impresión de registros de BD por consola.

### 8) Interfaz Gráfica de Usuario (GUI) y Eventos

Bibliotecas de clases para componentes gráficos y eventos Swing y AWT. Componentes gráficos Swing: Utilización de JFrame, Layouts, JPanel, JTable, JButton, JTextField, JLabel, JComboBox, JDialog. AWT y Manejo de Eventos: Implementación de Interfaces ActionListener, MouseListener, ItemListener. Implementación de los métodos MouseClicked(), ActionPerformed(), ItemStateChanged(). Implementación de aplicación grafica con acceso a base de datos y filtrado de registros con visualización re resultados listados en pantalla.

## III. Descripción de Actividades Teóricas y Prácticas:

### Semana Descripción



Programa de la Asignatura: PROGRAMACIÓN ORIENTADA A OBJETOS		Codigo: <b>IF008</b>
Departamento: Informática	Sede: Esquel	

1	Breve reseña historica desde la programacion estructurada al paradigma orientado a objetos. Pensamiento orientado a objetos, material audiovisual de referencia.
2	Introduccion del concepto de Abstraccion. Debate y conceptualizacion de problemas cotidianos modelizando en objetos e identificando solo aspectos relevantes para un proposito determinado.
3	Identificar clases, objetos y relaciones. Ejercitacion practica con texto literario y juego de Ajedrez.
4	Puesta en comun en aula y refinamiento de las abstracciones realizadas. Ejercitacion con diagrama de clases identificando metodos y atributos principales.
5	Presentacion de resumen y aspectos principales de un programa en Java. Criterios de seleccion de Entorno de Desarrollo, Instalacion y puesta a punto de los entornos.
6	Creacion de Programa Java sobre modelizacion basada en Texto Literario. Constructores, Getters y Setters: Instanciacion, visualizacion y modificacion de estado de los objetos durante su ciclo de vida.
7	Presentacion del concepto y características. Ejercitacion con ejemplos de herencia en el mundo real y representacion en el Paradigma Orientado a Objetos.
8	Parcial : Examen multiple choice sobre los conceptos de cada unidad vista en clase.
9	Refinamiento, retrabajo del codigo existente y nuevos desarrollos para la aplicacion de herencia sobre el practico del programa Java de Juego de Ajedrez.
10	Presentacion del concepto y características del Polimorfismo. Ejercitacion con ejemplos en el mundo real.
11	Resolucion practica de Polimorfismo. Ejercitacion practica con figuras geometricas. Aplicacion de polimorfismo al programa Java del Juego de Ajedrez.
12	Presentacion de manejo de excepciones en Java. Presentacion y ejercitacion practica de Acceso a Datos usando JDBC con BD MySQL. Incorporación al programa de Juego de Ajedrez para consultar datos a BD. Creacion de nuevo programa Java sobre BD existente.
13	Presentacion de librerias de clases GUI (Swing y AWT). Ejercitacion practica
14	Desarrollo de modulo de aplicativo comercial con acceso a DB y presentacion de informacion en GUI. Aplicando todo lo anteriormente visto.
15	Verificacion de avance, inspeccion, depuracion y debugging sobre el avance realizado de cada integrante.
16	Entrega Final: Presentacion del programa y defensa oral conceptual sobre el trabajo realizado

#### IV. Bibliografía:

<b>Título:</b> Ingeniería de Software Orientada a Objetos	<b>Año:</b> 2002
<b>Autor:</b> Bernd Bruegge / Allen H. Dutoit	<b>Editorial</b> Prentice Hall
<b>Básica para:</b> 1,2	<b>Complementaria para:</b> 3
<b>Título:</b> Object-Oriented and Classical Software Engineering (8th Edition)	<b>Año:</b> 2011
<b>Autor:</b> Stephen R. Schach	<b>Editorial</b> Mc Graw Hill
<b>Básica para:</b> 1,2,4	<b>Complementaria para:</b> 2,4
<b>Título:</b> Programación en Java 6	<b>Año:</b> 2011
<b>Autor:</b> Luis Joyanes Aguilar / Ignacio Zahonero Martinez	<b>Editorial</b> Mc Graw Hill
<b>Básica para:</b> 1,2,3,4,5,6,8	<b>Complementaria para:</b>
<b>Título:</b> Introducción a la Programación Orientada a Objetos con Java	<b>Año:</b> 2009
<b>Autor:</b> Varios Autores	<b>Editorial</b> Universidad Politécnica de
<b>Básica para:</b> 3,4,5,6,8	<b>Complementaria para:</b>
<b>Título:</b> Java Database Programming with JDBC	<b>Año:</b> 1996
<b>Autor:</b> Pratik Patel	<b>Editorial</b> O Reilly
<b>Básica para:</b> 7	<b>Complementaria para:</b>

#### IV 1. Sitios Web Recomendados:



Programa de la Asignatura: PROGRAMACIÓN ORIENTADA A OBJETOS	Codigo: <b>IF008</b>
Departamento: Informática	Sede: Esquel

**URL (Link):** <http://www.javahispano.org/>  
**Descripción:** Sitio con amplio contenido en español y publicaciones de la comunidad de desarrolladores Java.  
**Unidades para la que es requerida:** 1,3

**URL (Link):** <https://stackoverflow.com/>  
**Descripción:** Sitio de preguntas y resolución a problemas técnicos de programación con alto nivel de respuesta.  
**Unidades para la que es requerida:** 3

**URL (Link):** <https://docs.oracle.com/javase/7/docs/api/>  
**Descripción:** Sitio con la documentación oficial de las librerías del lenguaje de programación Java, referencia para el uso del lenguaje.  
**Unidades para la que es requerida:** 3, 5

**URL (Link):** <https://www.eclipse.org/>  
**Descripción:** Sitio Oficial del Entorno de Desarrollo Integrado Eclipse, abierto y gratuito amplia aceptación y soporte por parte de la comunidad de desarrolladores.-  
**Unidades para la que es requerida:** 3, 5

**V. Metodologías de Enseñanza:**

Tomando como punto de partida la información y conocimiento previo de programación que tienen los alumnos, e incorporando gradualmente los nuevos conceptos del Paradigma Orientado a Objetos con énfasis en lo teórico conceptual con el objetivo de desarrollar el pensamiento orientado a objetos, fomentando las capacidades de análisis de los alumnos y enfoque de pensamiento crítico previo al abordaje técnico de la programación. Los profesores abordarán los contenidos de manera vivencial y con soporte en la experiencia práctica en proyectos de desarrollo de software, asimismo se presentarán escenarios de problemas reales de desarrollo de software tanto estructurado como orientado a objetos. Se proporcionará material de lectura específica para cada unidad del programa, además de la bibliografía recomendada a fin de que sirva como base los desafíos que propondrá la cátedra. La ejercitación práctica se realizará de forma gradual, incorporando los conocimientos adquiridos a lo largo de cada unidad de la materia y fomentando la visión crítica y el enfoque en el desarrollo de software de calidad sobre la base de la auto-documentación, reutilización y la escalabilidad del código generado. Se realizarán puestas en común de los trabajos de cada alumno utilizando material de proyección, y fomentado el análisis de lo realizado, el debate grupal y el brainstorming ante un error/problema a resolver para lograr un objetivo propuesto. En cada punto de revisión/análisis de los trabajos individuales se detectarán los problemas técnicos y errores conceptuales con el objetivo de repasar los aspectos que resulte necesario a fin de promover el entendimiento y comprensión acabada de cada tema. Se propondrá el desarrollo del módulo de una aplicación real con propósito comercial que reúna los requisitos de un escenario de desarrollo orientado a objetos con definición precisa del contexto y el usuario tipo que la utilizará y los requerimientos que deberá satisfacer. Este trabajo será el entregable final de la materia que formará parte de los requisitos de aprobación de la cursada.

**VI.1 Condiciones para la aprobación del cursado de la asignatura:**

Para aprobar el cursado de la materia el alumno deberá aprobar dos (2) exámenes parciales, el primero escrito, el segundo oral y la entrega de un trabajo práctico final de un programa Java con Acceso a Datos e Interfaz Gráfica de Usuario. Se considerará aprobado el parcial si está correcto en un 60%. Si no aprueba cada parcial tendrá su respectivo examen recuperatorio con las mismas condiciones de aprobación. Para aprobar el trabajo práctico el alumno debe haber alcanzado los objetivos mínimos propuestos, con autonomía de trabajo y tiempos acordes al cronograma propuesto. Como calificación final del cursado se calcula el promedio entre las notas obtenidas en los dos exámenes parciales y la nota del trabajo práctico final. Si el alumno, sin justificación, no asistiera a algún parcial y su correspondiente recuperatorio se considerará ausente. Los alumnos que no registren condición de ausente tendrán oportunidad de recuperar los parciales desaprobados en el cuatrimestre en el que no se dicta la materia, en fecha y hora fijada por la Facultad a tal efecto.



Programa de la Asignatura: PROGRAMACIÓN ORIENTADA A OBJETOS	Codigo: <b>IF008</b>
Departamento: Informática	Sede: Esquel

**VI.2 Condiciones para la aprobación de la asignatura:**

Aprobación regular de la materia: Para aprobar la materia el alumno regular debe cumplir con las condiciones de correlatividad y aprobación de cursado y rendir un examen final regular, en las fechas y horas que fije la Facultad a tal efecto. Aprobación libre de la materia: El alumno libre debe presentar una propuesta previa a la mesa de examen donde desarrollara un programa Java con aplicación de los contenidos de la materia. Deberá entregar el práctico propuesto y rendir, en primera instancia, un examen práctico, el cual incluye contenidos de práctica y teoría. Es escrito y puede incluir análisis de código fuente y desarrollo conceptual de los contenidos de la materia Este primer examen puede tomarse en dos partes si se considera necesario. Si el resultado del mismo es satisfactorio se toma un siguiente examen oral, en el cual se incluyen contenidos integrados de toda la asignatura.

**Vigencia de este Programa**

Año	Firma	Profesor Responsable
2018		AGUADO, MATIAS ANTONIO

**Visado**

Decano	Sec. Académico Facultad	Jefe de Departamento
Dr. Ing. Francisco Andrés CARABELLI DECANO Facultad de Ingeniería	Ing. Sandra Beatriz KUHN SECRETARIO ACADEMICO Facultad de Ingeniería	
Fecha:	Fecha:	Fecha: