



UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS
(Universidad del Perú, DECANA DE AMÉRICA)

FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SOFTWARE
“Año del Bicentenario del Perú: 200 años de independencia”

1. INFORMACIÓN GENERAL

- 1.1 Nombre y código de la asignatura: **20W0401 - ALGORÍTMICA II**
1.2 Número de créditos: 04
1.3 Número de horas semanales: Teoría: 3 horas, Laboratorio: 02 horas
1.4 Ciclo de estudio: IV
1.5 Periodo Académico: 2021 - 2
1.6 Pre-requisitos: Algorítmica I / Programación y Fundamentos de Algorítmica
1.7 Profesores: Mg. Cabrera Díaz Javier Elmer. jcabrerad@unmsm.edu.pe
Mg. Augusto Cortez Vasquez. acortezv@unmsm.edu.pe
Mg. Gilberto A. Salina Azaña. gsalinasa@unmsm.edu.pe

2. SUMILLA

La presente asignatura pertenece al área de formación profesional, de naturaleza teórica, práctica en sesiones de laboratorio, con el propósito de conocer, comprender y aplicar, los conceptos y principios del enfoque orientado a objetos en la programación.

Comprende: tipos abstractos de datos, objetos, clases, métodos, mensajes, arreglo de objetos, relaciones entre clases, herencia, polimorfismo, clases abstractas, interfaces, interfaces gráficas de usuario, gestión de errores y excepciones, clases genéricas, persistencia de objetos, programación concurrente, hilos, conectividad con base de datos.

3. COMPETENCIA GENERAL

Al finalizar la asignatura, el estudiante tendrá las siguientes competencias generales:

- CG01 Gestiona la información y la difusión de conocimientos con adecuada comunicación oral y escrita de la propia profesión, ejerciendo el derecho de libertad de pensamiento con responsabilidad
- CG02 Capacidad de análisis y síntesis en la toma de decisiones con responsabilidad, sentido crítico y autocrítico
- CG03 Desempeña su profesión con liderazgo, adecuándose a los cambios y a las nuevas tendencias, comprometido con la paz, medio ambiente, equidad de género, defensa de los derechos humanos y valores democráticos

- CG04 Trabaja en equipo con una perspectiva transdisciplinar para comprender y transformar la realidad compleja
- CG05 Genera nuevos conocimientos que aportan al desarrollo de la sociedad mediante la investigación, con sentido ético
- CG06 Aplica conocimientos a la práctica para resolver problemas con compromiso ético

COMPETENCIAS ESPECÍFICAS

Al finalizar la asignatura, el estudiante tendrá las siguientes competencias específicas:

- CE2 Capacidad de Análisis
- CE3 Pensamiento Crítico
- CE4 Comunicación oral y escrita
- CE14 Aplica metodologías, métodos Técnicas

4. PROGRAMACIÓN:

UNIDAD I: Introducción a la programación orientada a objetos.				
Capacidades		<ul style="list-style-type: none"> ● Identifica, define y aplica los conceptos de la programación orientada a objetos, mediante TAD. ● Diseña y modela a través de diagramas de clase de diseño, análisis o dominio según el caso basados en UML. ● Implementa clases y objetos cumpliendo las de la POO ● Sobrecarga de métodos para optimizar el código de una clase ● Analiza, diseña, implementa y utiliza objetos de programación que permitan resolver problemas reales y de ingeniería. 		
Contenidos		Actividades	Recursos	Estrategias
Semana 01	Presentación del sílabo. Normas de participación en el aula virtual. Teoría: Introducción TAD POO nuevo paradigma. Diferencia con programación estructurada y modular. Nociones de Clase: representación de la información por medio de objetos; Atributos o estado. Métodos o comportamiento. Abstracción de objetos en clases. Constructores. Destructores Métodos de acceso: modificación del estado de un objeto: setters y getters Métodos de instancia y de clase.	SÍNCRONAS Video conferencia Establecemos acuerdos de convivencia en el ambiente virtual. Dialogamos sobre el contenido del sílabo. Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del sílabo Lectura del material teórico Resolución de problemas sobre los temas tratados de manera personal y colaborativa Formación de equipos de trabajo Participación en el foro de discusión	Aula Virtual Presentación del material: Sílabo, vídeos, lecturas, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking

	Laboratorio: Lenguaje de programación Java. Declaración y construcción de clases			
Semana 02	Teoría: Declaración de clases. Constructores. Destructores. Uso y aplicaciones Instanciación de clases Modificadores de acceso: públicos, protegidos, privados Ocultamiento de la información. Laboratorio: Implementación en Java Proyecto Asignación de temas de proyecto grupal.	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 03	Teoría: Referencia al objeto actual. Métodos de acceso: declaración, mensajes, paso de parámetros, retorno de valores. Otras operaciones Modificadores de acceso: públicos, protegidos, privados. Diferentes formas de representar una clase manteniendo su comportamiento. Diagramas de clases: diseño, análisis y dominio Laboratorio: Implementación de programas en lenguaje Java. Proyecto Primer avance del proyecto de la asignatura.	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking

UNIDAD II: Relaciones entre clases.				
Capacidades		<ul style="list-style-type: none"> ● Comprende, explica y aplica las relaciones entre clases y objetos a problemas informáticos. ● Analiza, diseña (modela) e implementa soluciones a problemas de la realidad aplicando relaciones entre objetos utilizando arreglo de objetos aplicando mecanismos de reutilización. ● Analiza, diseña (modela) e implementa soluciones a problemas de la realidad teniendo en cuenta la herencia simple y la herencia múltiple ● Valora la metodología de la programación orientada a objetos. 		
Contenidos		Actividades	Recursos	Estrategias
Semana 04	Teoría Relaciones entre objetos: Uso, asociación, Representación en diagrama de clases. Implementación de un vector de objetos y sus operaciones básicas. Ejemplo de diseño e implementación Laboratorio: Analiza implementaciones Diseña e implementa problemas re conjunto de objetos en Java	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 05	Teoría Relaciones entre objetos: Uso, asociación, agregación y composición. Representación en diagrama de clases. Laboratorio: Analiza implementaciones Diseña e implementa problemas re conjunto de objetos en Java Proyecto Segundo avance del proyecto de la asignatura.	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 06	Teoría: Relaciones entre clases: la herencia y la representación en	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis.

	<p>diagrama de clase.</p> <p>Clase base, clase derivada</p> <p>Clasificación Herencia simple y múltiple.</p> <p>Constructores, destructores en clases derivadas.</p> <p>Redefinición de métodos de clases derivadas.</p> <p>Laboratorio:</p> <p>Diseño e implementación de una aplicación y sus relaciones</p>	<p>ASÍNCRONAS</p> <p>Revisión del material de clase.</p> <p>Resolución de los problemas planteados en clase de manera personal y colaborativa</p> <p>Resolución de la guía de laboratorio</p> <p>Participación en el foro de discusión</p>	<p>Aula Virtual</p> <p>Presentación del material:</p> <p>vídeos, lecturas, guía de laboratorio, chat, foros de discusión</p>	<p>Formulación de preguntas</p> <p>Aula invertida</p> <p>Aprendizaje Basado en Problemas (ABP)</p> <p>Trabajo colaborativo</p> <p>Design thinking</p>
--	---	---	--	---

UNIDAD III: Clases abstractas, interfaces y excepciones

Capacidades		<ul style="list-style-type: none"> ● Comprende, explica y aplica las clases abstractas, interfaces y excepciones. ● Analiza, diseña modela e implementa soluciones a problemas de la realidad aplicando excepciones permitiendo el control de errores y recuperar trazas de errores de grandes proyectos. ● Analiza, diseña (modela) e implementa soluciones a problemas de la realidad aplicando mecanismos de reutilización de código. ● Valora la metodología de la programación orientada a objetos. 		
Contenidos		Actividades	Recursos	Estrategias
Semana 07	<p>Teoría:</p> <p>Definición de clases abstractas, métodos abstractos.</p> <p>Modelado de clases abstractas.</p> <p>Implementación de clases abstractas.</p> <p>Definición Interfaces, métodos abstractos.</p> <p>Modelado de interfaces.</p> <p>Implementación interfaces, herencia de interfaces.</p> <p>Laboratorio:</p> <p>Diseño e implementación de una aplicación y sus relaciones usando</p>	<p>SÍNCRONAS</p> <p>Video conferencia</p> <p>Desarrollo de la clase</p>	<p>Aula Virtual</p> <p>Google Meet</p> <p>Material preparado por el docente.</p> <p>Aula Virtual</p> <p>Presentación del material:</p> <p>vídeos, lecturas, guía de laboratorio, chat, foros de discusión</p>	<p>Recuperación de saberes previos.</p> <p>Exposición.</p> <p>Síntesis.</p> <p>Formulación de preguntas</p>
		<p>ASÍNCRONAS</p> <p>Revisión del material de clase.</p> <p>Resolución de los problemas planteados en clase de manera personal y colaborativa</p> <p>Resolución de la guía de laboratorio</p> <p>Participación en el foro de discusión</p>		<p>Aula invertida</p> <p>Aprendizaje Basado en Problemas (ABP)</p> <p>Trabajo colaborativo</p> <p>Design thinking</p>

	<p>clases abstractas en Java</p> <p>Proyecto</p> <p>Tercer avance del proyecto de la asignatura.</p>			
Semana 8	Examen Parcial	<p>SÍNCRONAS</p> <p>Examen Final</p>	Aula Virtual Examen virtual preparado por el docente	Formulación de preguntas
Semana 9	<p>Teoría:</p> <p>Definición excepciones</p> <p>Tipos de excepciones</p> <p>Gestión de excepciones, manejo de excepciones, lanzamiento de excepciones.</p> <p>Creación y manejo de excepciones definidas por el usuario.</p> <p>Laboratorio:</p> <p>Diseño e implementación de una aplicación y sus relaciones usando control de errores en Java</p>	<p>SÍNCRONAS</p> <p>Video conferencia</p> <p>Desarrollo de la clase</p>	Aula Virtual Google Meet Material preparado por el docente.	<p>Recuperación de saberes previos.</p> <p>Exposición. Síntesis. Formulación de preguntas</p>
		<p>ASÍNCRONAS</p> <p>Revisión del material de clase.</p> <p>Resolución de los problemas planteados en clase de manera personal y colaborativa</p> <p>Resolución de la guía de laboratorio</p> <p>Participación en el foro de discusión</p>	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	<p>Aula invertida</p> <p>Aprendizaje Basado en Problemas (ABP)</p> <p>Trabajo colaborativo</p> <p>Design thinking</p>

UNIDAD III: Clases genéricas, persistencia, polimorfismo e interfaz gráfica.

Capacidades		<ul style="list-style-type: none"> ● Diseña y aplica clases y métodos que actúen sobre diferentes tipos de datos. ● Analiza, diseña modela e implementa soluciones a problemas de la realidad aplicando la persistencia de objetos ● Manipula objetos de diferentes clases por medio de un mismo método y realizar la misma operación basado en el concepto de polimorfismo. ● Identifica, define y aplica las interfaces gráficas de usuario (GUI). 		
Contenidos		Actividades	Recursos	Estrategias
Semana 10	Teoría: Definición de clases genéricas, métodos genéricos. Modelado de clases genéricas. Implementación de clases genéricas. Definición de colecciones, clasificación Operaciones básicas Implementación. Reutilización. Laboratorio: Diseño e implementación de una aplicación implementando clases genéricas en Java Proyecto Cuarto avance del proyecto de la asignatura.	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 11	Teoría: Definición polimorfismo Tipos de polimorfismo Diseño e implementación Reutilización de código. Laboratorio: Diseño e implementación de una aplicación y sus relaciones usando	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat,	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo

	polimorfismo en Java	personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	foros de discusión	Design thinking
Semana 12	Teoría: Definición de interfaces graficas de usuario conceptos de Disposición de componentes Gestión Eventos Diseño e implementación Práctica: Construcción de una aplicación bajo el paradigma de la POO usando interfaces graficas de usuario Proyecto Quinto avance del proyecto de la asignatura.	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 13	Teoría: Definición de flujos y archivos. Clasificación. operaciones básicas y tipos de acceso Manejo de objetos persistentes. Laboratorio: Construcción de una aplicación bajo el paradigma de la POO usando interfaces graficas de usuario y aplicando conceptos de persistencia	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas
		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 14	Teoría: Definición patrones. Clasificación: creacionales, estructurales y de comportamiento Laboratorio: Construcción de una aplicación bajo el paradigma de la POO	SÍNCRONAS Video conferencia Desarrollo de la clase	Aula Virtual Google Meet Material preparado por el docente.	Recuperación de saberes previos. Exposición. Síntesis. Formulación de preguntas

		ASÍNCRONAS Revisión del material de clase. Resolución de los problemas planteados en clase de manera personal y colaborativa Resolución de la guía de laboratorio Participación en el foro de discusión	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 15	Teoría: Exposición del proyecto Presentación del informe final del proyecto.	SÍNCRONAS Video conferencia Expone el grupo su proyecto de software desarrollado.	Aula Virtual Google Meet Material preparado por el docente.	Trabajo en grupo. Uso de videos. Uso de Presentación.
		ASÍNCRONAS Elaboran el informe del proyecto de la asignatura	Aula Virtual Presentación del material: vídeos, lecturas, guía de laboratorio, chat, foros de discusión	Aula invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking
Semana 16	Examen Final	SÍNCRONAS Examen Final	Aula Virtual Examen virtual preparado por el docente	Formulación de preguntas

5. ESTRATEGIA DIDACTICA

Por parte del docente se desarrollará la asignatura siguiendo los criterios inductivos, flexibles, participativos y constructivos con la participación del alumno; en este sentido se utilizará las técnicas participativas, laboratorios guiados, desarrollo de proyectos.

Por parte del estudiante, participará activamente a través de elaboración de pseudocódigos y programas, ejercicios calificados, exámenes calificados, elaboración de proyecto en grupo.

6. EVALUACIÓN DEL APRENDIZAJE

Unidades	Criterios		Evaluación SUM	
			Denominación	Porcentaje (%)
1,2	EXAMEN PARCIAL		N1	20
	EVALUACIONES CONTINUAS (EC=prom(PP+proy))		N2	60
1,2,3,4	Practicas calificadas y laboratorio (Promedio de todas PCs y PLs)	PP =prom(PCs+PLs)		

1,2,3,4	Proyecto de fin de curso. (Promedios entregables y defensa final)	Proy=prom(PE+DF)		
1,2,3,4	EXAMEN FINAL		N3	20
	PROMEDIO FINAL		N1+N2+N3	100

7. REFERENCIAS BIBLIOGRÁFICAS

A. BASICA

- Liang, Daniel. (2012). Introduccion to Java Programming.. Pearson Educación. 1316p. ISBN: 9780133761313.
- Deitel, Paul & Deitel Harvey. (2012). Java como programar. México. Edit. Pearson Educación. 616p. ISBN: 978-607-32-1150-5.
- Bell, Douglas & Parr, Mike. (2011). Java para estudiantes. México. Edit Pearson Educación. 552p. ISBN: 978-607-32-0557-3.
- Wu, Thomas. (2008) Programación en Java. Introducción a la Programación Orientada a Objetos. México D. F. McGraw-Hill. 1186p.
- Joyanes, Luis & Zahonero, Ignacio. (2010). Programacion C, C++, Java y UML. Mexico D. F. MacGraw-Hill, 1000p.

B. COMPLEMENTARIA

- Ceballos Sierra, Francisco Javier (2000) Java 2. Curso de Programación. México D. F. Alfaomega Grupo Editor S. A. 779p

C. ENLACES Y SITIOS WEB

Interfaz gráfica de usuario:

- o <http://www.proyectoautodidacta.com/comics/la-interfaz-grfica-de-usuario-gui/>
- o <http://doutdex.wordpress.com/2007/06/09/gui-graphical-user-interface-interface-grafic-a-de-usuario-en-java/>