



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

**FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

SÍLABO

“Año del Bicentenario del Perú: 200 años de independencia”

1. INFORMACIÓN GENERAL

- 1.1. Nombre y código de la asignatura : ALGORÍTMICA Y PROGRAMACIÓN ORIENTADA A OBJETOS
1.2. Código de la asignatura : 20118041
1.3. Tipo de asignatura : Obligatoria
1.4. Horas semanales : Teoría: 03 hora, Laboratorio: 02 horas
1.5. Semestre o año académico : 2021 – II
1.6. Ciclo : III
1.7. Créditos : 4
1.8. Modalidad : No presencial (virtual)
1.9. Prerrequisitos : 2010301 Programación y Fundamentos de Algorítmica.
1.10. Profesores: Mg. Cabrera Díaz Javier Elmer. jcabrerad@unmsm.edu.pe
Mg. Luis Angel Guerra Grados. lguerrag1@unmsm.edu.pe
Mg. Gilberto A. Salina Azaña. gsalinasa@unmsm.edu.pe
Mg. Jorge L. Zavaleta Campos. jzavaletac@unmsm.edu.pe

2. SUMILLA

Esta asignatura pertenece al área de formación básica, es de naturaleza teórica y práctica, tiene el propósito de proveer a los estudiantes los conceptos y técnicas de orientación a objetos que son fundamentos teóricos y prácticos para la resolución de problemas y construcción de programas basados y orientados a objetos para lograr la competencia: “Desarrolla y mantiene los sistemas de información y tecnologías de información basadas en metodologías, estándares y métricas internacionales de calidad, cubriendo todas las etapas del ciclo de la vida del software con compromiso ético y solidario”. Los contenidos principales son: el paradigma de la programación orientada a objetos. Conceptos básicos (objeto, clase, encapsulación, polimorfismo, etc). Relaciona semánticas y jerárquicas, clases abstractas. La especificación y construcción de Patrones o plantillas, excepciones y persistencia. Casos prácticos y su implementación en lenguajes como C++ y Java.

3. COMPETENCIAS GENERALES

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 transdisciplinaria 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:

- CE01 Capacidad de Análisis
- CE02 Pensamiento Crítico
- CE03 Comunicación oral y escrita
- CE04 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 12.10.2021 Presentación del sílabo. Normas de participación en el aula virtual. Teoría: <ul style="list-style-type: none"> • TAD. Introducción • POO nuevo paradigma. Diferencia con programación estructurada y modular. <ul style="list-style-type: none"> ○ Nociones de Clase: representación de la información por medio de objetos; Atributos o estado. Métodos o comportamiento. Abstracción de objetos en clases. ○ Métodos de acceso: modificación del estado de un objeto: setters y getters ○ Métodos de instancia y de clase. Laboratorio: Lenguaje de programación Java. Declaración y construcción de clases	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 invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking	
Semana 02 19.10.2021 Teoría: <ul style="list-style-type: none"> • Declaración de clases. <ul style="list-style-type: none"> ○ 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 invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking	
Semana 03 26.10.2021 Teoría: <ul style="list-style-type: none"> • 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:	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 invertida Aprendizaje Basado en Problemas (ABP) Trabajo colaborativo Design thinking	

	Implementación de programas en lenguaje Java. Proyecto Primer avance del proyecto de la asignatura.			
--	--	--	--	--

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 02.11.2021 Teoría <ul style="list-style-type: none"> Relaciones entre objetos: Uso, asociación. <ul style="list-style-type: none"> 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 09.11.2021 Teoría <ul style="list-style-type: none"> Relaciones entre objetos: Uso, asociación, agregación y composición. <ul style="list-style-type: none"> 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 16.11.2021 Teoría: <ul style="list-style-type: none"> Relaciones entre clases: la herencia y la representación en diagrama de clase. <ul style="list-style-type: none"> Clase base, clase derivada Clasificación Herencia simple y múltiple. Constructores, destructores en clases derivadas. Redefinición de métodos de clases derivadas. Laboratorio: Diseño e implementación de una aplicación y sus relaciones	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 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 23.1a.2021	Teoría: <ul style="list-style-type: none"> • Definición de clases abstractas, métodos abstractos. <ul style="list-style-type: none"> ○ Modelado de clases abstractas. • Implementación de clases abstractas. • Definición Interfaces, métodos abstractos. <ul style="list-style-type: none"> ○ Modelado de interfaces. ○ Implementación interfaces, herencia de interfaces. Laboratorio: Diseño e implementación de una aplicación y sus relaciones usando clases abstractas en Java Proyecto Tercer 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		
Semana 8 30.11.2021	Examen Parcial	SÍNCRONAS Examen Final	Aula Virtual Examen virtual preparado por el docente	Formulación de preguntas
Semana 9 07.12.2021	Teoría: <ul style="list-style-type: none"> • Excepciones. Def. Tipos de excepciones <ul style="list-style-type: none"> ○ Gestión de excepciones, manejo de excepciones, lanzamiento de excepciones. ○ Creación y manejo de excepciones definidas por el usuario. Laboratorio: Diseño e implementación de una aplicación y sus relaciones usando control de errores 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		

UNIDAD IV: 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 14.12.2021	Teoría: <ul style="list-style-type: none"> • Clases Genéricas. Def. Modelado de clases genéricas. <ul style="list-style-type: none"> ○ Implementación de clases genéricas. ○ Definición de colecciones, clasificación. Operaciones básicas. 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 21.12.2021	Teoría: <ul style="list-style-type: none"> • Polimorfismo def. Tipos de polimorfismo. <ul style="list-style-type: none"> ○ Diseño e implementación. Reutilización de código. Laboratorio: Diseño e implementación de una aplicación y sus relaciones usando polimorfismo 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 12 28.12.2021	Teoría: <ul style="list-style-type: none"> • Interfaces graficas de usuario. <ul style="list-style-type: none"> ○ 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 04.01.2022	Teoría: <ul style="list-style-type: none"> • Persistencia. Def. de flujos y archivos. <ul style="list-style-type: none"> ○ 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 11.01.2022	Teoría: <ul style="list-style-type: none"> Patrones de diseño. def. <ul style="list-style-type: none"> Clasificación: creacionales, estructurales y de comportamiento Laboratorio: Implementación usando patrones de software	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 18.01.2022	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 25.01.2022	Examen Final	SÍNCRONAS Examen Final	Aula Virtual Examen virtual preparado por el docente	Formulación de preguntas

5. ESTRATEGIA DIDÁCTICA

El profesor promueve la participación de los estudiantes en su propio aprendizaje utilizando estrategias de aprendizaje activo como el método de casos prácticos para la parte práctica y el trabajo colaborativo para el desarrollo del proyecto, que conlleven al “Saber a Aprender”, “Saber a Hacer” y “Saber Ser” de forma cooperativa o colaborativa. Para tal efecto, publicara en el Aula Virtual los materiales didácticos de la asignatura, los problemas a desarrollarse y ejemplos desarrollados y la guía básica del proyecto en equipo.

Para cada sesión de aprendizaje, el estudiante habrá revisado el material que el profesor ha colocado en el aula virtual. Seguidamente, el profesor desarrolla los tópicos programados en el silabo, haciendo uso de la video conferencia para realizar las aclaraciones a los conceptos teóricos, prácticos y/o coordinaciones de la práctica o proyecto a desarrollar, la cual se grabará y estará disponible permanentemente en el aula virtual. Durante la video conferencia, el estudiante participará a través del chat o foros de discusión, los cuales pueden realizarse hasta varios días después para mejorar su aporte al tema tratado en la sesión de aprendizaje. Finalmente, el estudiante implementará el ejemplo desarrollado y el problema planteado en para la sesión de aprendizaje.

Design thinking

Por lo cual, concluimos que el estudiante desarrollará un aprendizaje autónomo y el docente desempeñará el papel de un facilitador y guía del aprendizaje del estudiante.

6. EVALUACIÓN DEL APRENDIZAJE

Unidades	Criterios	Evaluación SUM	
		Denominación	Porcentaje (%)
1,2	EXAMEN PARCIAL	N1	20
	EVALUACIONES CONTINUAS (EC=prom(PP+proy))		
1,2,3,4	Practicas calificadas y laboratorio (Promedio de todas PCs y PLs)	N2	60
1,2,3,4	Proyecto de fin de curso. (Promedios entregables y defensa final)		
1,2,3,4	EXAMEN FINAL	N3	20
	PROMEDIO FINAL	N1+N2+N3	100

Matriz de evaluación por competencias para una asignatura

Unidades de aprendizaje	Criterio	Desempeño	Productos	Instrumentos de evaluación	Pesos
Introducción a la programación orientada a objetos	Analiza, diseña (modela), implementa y utiliza objetos de programación que permitan resolver problemas reales.	Construye e implementa diagramas de clases usando el enfoque orientado a objetos.	Diagramas de clases con la implementación de clases en el lenguaje Java	Tarea de diagramas de clases y construcción de clases en Java. Participación en clase o foro de discusión. Examen Parcial	
Relaciones entre clases	Analiza, diseña (modela) e implementa soluciones a problemas reales aplicando relaciones entre clases (asociación, composición y reutilización)	Construye e implementa diagramas de clases identificando las relaciones entre ellas (asociación composición y reutilización).	Diagramas de clases usando relaciones de asociación composición y agregación con su la implementación de clases en el lenguaje Java	Tarea de diagramas de clases usando relaciones y construcción de clases en Java. Participación en clase o foro de discusión. Examen Parcial.	
Clase abstractas, interfaces y excepciones	Analiza, diseña (modela) e implementa soluciones a problemas reales aplicando clases abstractas e interfaces. Implementa excepciones para el control de errores y recuperar trazas de errores.	Construye e implementa diagramas de clases identificando las relaciones entre ellas, aplicando los conceptos de clases abstractas e interfaces.	Software de bajo el paradigma de la POO aplicando conceptos de clases abstractas e interfaces con la gestión de errores con el uso de excepciones, en base al diagrama de clases.	Tarea de aplicaciones de software bajo el enfoque de la POO, clases abstractas, interfaces y uso de excepciones. Participación en clase o foro de discusión. Informes del proyecto. Examen Parcial.	
Clases genéricas, persistencia, polimorfismo e interfaz gráfica	Analiza, diseña (modela) e implementa soluciones a problemas reales aplicando los conceptos de polimorfismo, persistencia e interfaces graficas de usuario.	Construye e implementa soluciones a la problemas reales utilizando conceptos de polimorfismo, persistencia e interfaces graficas de usuario.	Software de bajo el paradigma de la POO agregando los conceptos de polimorfismo, persistencia e interfaces graficas de usuario, en base al diagrama de clases.	Tarea de aplicaciones de software bajo el enfoque de la POO usando polimorfismo, persistencia e interfaces graficas de usuario. Participación en clase o foro de discusión. Informes del proyecto. Examen Final.	
Persistencia y patrones de diseño.	Analiza, diseña (modela) e implementa soluciones a problemas reales aplicando los conceptos de persistencia y patrones de diseño.	Construye e implementa soluciones a la problemas reales utilizando conceptos de persistencia y patrones de diseño.	Software de bajo el paradigma de la POO agregando los conceptos de persistencia y patrones de diseño, en base al diagrama de clases.	Tarea de aplicaciones de software Participación en clase o foro de discusión. Examen Final. Informes del proyecto. Exposición del proyecto. Software del proyecto.	

7. REFERENCIAS BIBLIOGRÁFICAS

- 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.
- Eckel, Bruce (2003) Piensa en Java. 2da Edicion. Madrid, España: Prentice-Hall.
- 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.
- Joyanes, Luis & Zahonero, Ignacio (2010) Programación C, C++, Java y UML. México D.F.: MacGraw-Hill, 1000p.
- Deitel & Deitel. (2008) Java como programar. México. : Edit. Pearson Prentice Hall.
- Ceballos Sierra, Francisco Javier. (2000) Java 2. Curso de Programación. México D. F.: Alfaomega Grupo Editor S. A. 779p.
- Nolasco Valenzuela, Jorge, Jara Barrientos, José. (2015) Java 8 y Android Studio. Lima. Perú. : Editora INKADROID S.A.C. 238p. ISBN: 978-612-46976-0-9.
- Patrones de diseño de software:
<http://www.proactiva-calidad.com/java/patrones/index.html>
<http://di002.edv.uniovi.es/~cueva/asignaturas/PFCOviedo/PFCpatronesJava.pdf>