

SILABO

Silabo adaptado en el marco de la emergencia sanitaria por el COVID-19

ALGORITMO Y ESTRUCTURA DE DATOS I

Asignatura no presencial

ÁREA CURRICULAR: CIENCIAS DE LA COMPUTACIÓN

I. DATOS GENERALES

1.1 Departamento Académico	: Ingeniería y Arquitectura
1.2 Semestre Académico	: 2022–II
1.3 Código de la asignatura	: 09005303050
1.4 Ciclo	: III
1.5 Créditos	: 05
1.6 Horas semanales totales	: 11
1.6.1 Horas lectivas (Teoría, Práctica, Laboratorio)	: 7 (T=3, P=1, L=3)
1.6.2 Horas no lectivas	: 4
1.7 Condición de la asignatura	: Obligatoria
1.8 Requisitos	: 09111402050 Introducción a la Programación

II. SUMILLA

Es de naturaleza teórico-práctico, dirigido a que el alumno desarrolle programas para computadora haciendo uso de las características básicas de la programación orientada a objetos y almacenando datos en arreglos. Los principales temas a tratar son: Clases y objetos, atributos y métodos, encapsulamiento, herencia, polimorfismo, sobrecarga de métodos, algoritmos con vectores.

Unidades: I: Introducción a la Teoría Orientada a Objetos, II: Algoritmos para la manipulación de datos en Vectores. III: Introducción a la Teoría Orientada a Objetos – Métodos y Atributos de Instancia y de Clase. IV: Propiedades de la Teoría Orientada a Objetos: Herencia, Polimorfismo y Encapsulamiento. Vectores de Objetos.

III. COMPETENCIAS Y SUS COMPONENTES COMPRENDIDOS EN LA ASIGNATURA

3.1 Competencias

- Aplica conocimientos de computación y matemáticas apropiadas para los resultados del estudiante y las disciplinas enseñadas.
- Analiza un problema e identifica y define los requerimientos apropiados para su solución.
- Diseña, implementa y evalúa un sistema basado en computadoras, procesos, componentes o programa que satisfagan las necesidades requeridas.
- Usa técnicas, destrezas, y herramientas modernas necesarias para la práctica de la computación.

3.2 Componentes

Capacidades

- Determina el esquema de solución de un problema, seleccionando los componentes de la Teoría Orientada a Objetos.
- Aplica la lógica de programación desarrollada.

**Contenidos
actitudinales -**

Respeto a la
persona.

- Respeto de las normas establecidas por la universidad.
- Llega puntual al aula y tiene una constante asistencia a clases que demuestra un mayor interés en el curso.
- Compromiso para desarrollar los ejercicios propuestos.



IV. PROGRAMACIÓN DE CONTENIDOS

UNIDAD I: Introducción a la Teoría Orientada a Objetos.					
CAPACIDAD: <ul style="list-style-type: none"> Describe, explica y aplica los conceptos de la Teoría Orientada a Objetos. Implementa métodos de acuerdo a sus necesidades. 					
SEMANA	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	ACTIVIDAD DE APRENDIZAJE	HORAS	
				L	T. I.
1	Primera sesión Concepto de la Teoría Orientada a Objetos (TOO). Definición de Clase y Objeto. Características de las Clases. Diagrama de Clase. Segunda sesión Componentes de una clase, atributos y métodos. Correlación con el lenguaje de programación. Laboratorio Desarrollo de soluciones creando clases, declarando atributos e identificando métodos.	<ul style="list-style-type: none"> Entendimiento de los conceptos de la TOO mediante ejemplos abstraídos del entorno. Explicación de la correlación de los conceptos de la TOO con el lenguaje de programación Visual C# y Java. 	Lectivas (L): <ul style="list-style-type: none"> Desarrollo del tema – 3 h Desarrollo de ejercicios - 1 h Ejercicios guiados – 3h Trabajo Independiente (T.I) <ul style="list-style-type: none"> Desarrollo de ejercicios – 4h 	7	4
2	Primera sesión Métodos: Definición y clasificación. Llamada de métodos. Métodos definidos por usuario. Métodos que no retornan valor y no reciben parámetros. Segunda sesión Métodos que no retornan valor y reciben parámetros. Laboratorio Desarrollo de soluciones usando métodos que no retornan valor pero que reciben y no reciben parámetros.	<ul style="list-style-type: none"> Explicación del concepto de métodos. Desarrollo de ejemplos de métodos que no retornan valor y no reciben parámetros, y métodos que no retornan valor y reciben parámetros. Ejercicios de refuerzo. 	Lectivas (L): <ul style="list-style-type: none"> Desarrollo del tema – 3 h Desarrollo de ejercicios - 1 h Ejercicios guiados – 3h Trabajo Independiente (T.I) - Desarrollo de ejercicios – 4h	7	4

3	<p>Primera sesión Métodos que retornan valor y no reciben parámetros.</p> <p>Segunda sesión Métodos que retornan valor y reciben parámetros.</p> <p>Laboratorio Desarrollo de soluciones usando métodos que retornan valor pero que reciben y no reciben parámetros.</p>	<ul style="list-style-type: none"> - Desarrollo de ejemplos de métodos que retornan valor y no reciben parámetros, y métodos que retornan valor y reciben parámetros. - Ejercicios de refuerzo. 	<p><u>Lectivas (L):</u></p> <ul style="list-style-type: none"> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <p><u>Trabajo Independiente (T.I)</u></p> <ul style="list-style-type: none"> - Desarrollo de ejercicios – 4h 	7	4
---	---	---	--	---	---



UNIDAD II. Algoritmos para la manipulación de datos en Vectores

CAPACIDAD:

- Aplica el algoritmo de búsqueda de datos en un arreglo unidimensional, en combinación con la TOO.
- Aplica el algoritmo de modificación de datos en un arreglo unidimensional, en combinación con la TOO.
- Aplica el algoritmo de ordenamiento de datos en un arreglo unidimensional, en combinación con la TOO.
- Aplica el algoritmo de eliminación de datos en un arreglo unidimensional, en combinación con la TOO.

4	Primera sesión Algoritmo de búsqueda y modificación de datos en un vector. Segunda sesión Algoritmo de eliminación de datos en vector. Laboratorio Desarrollo de soluciones con operaciones sobre los elementos de un vector: búsqueda, modificación y eliminación. Evaluación de Laboratorio N° 01.	<ul style="list-style-type: none"> - Explicación de los algoritmos de búsqueda, modificación y eliminación de datos en vectores. - Desarrollo de ejemplos de los algoritmos de búsqueda, modificación y eliminación. - Ejercicios de refuerzo. 	<u>Lectivas (L):</u> - Desarrollo del tema – 3h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h	7	4
5	Primera sesión Algoritmo de ordenamiento de datos en un vector, método de Transposición. Ordenamiento ascendente y descendente. Segunda sesión Práctica Calificada 1 Desarrollo de soluciones con métodos, vectores y algoritmos. Laboratorio Desarrollo de soluciones de ordenamiento de datos en un vector.	<ul style="list-style-type: none"> - Explicación del algoritmo de ordenamiento de datos en vectores. - Desarrollo de ejemplos del algoritmo de ordenamiento. - Ejercicios de refuerzo. - Evaluación: práctica calificada 1. 	<u>Lectivas (L):</u> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h	7	4

UNIDAD III. Introducción a la Teoría Orientada a Objetos – Métodos y Atributos de Instancia y de Clase

CAPACIDAD: <ul style="list-style-type: none"> Determina, cuando lo requiere, la creación de métodos y atributos de instancia. Determina, cuando lo requiere, la creación de métodos y atributos de clase. 					
6	Primera sesión Desarrollo de soluciones con métodos, vectores y algoritmos. Segunda sesión Método Constructor. Método main. Llamada al método constructor.	<ul style="list-style-type: none"> Ejercicios de refuerzo. Explicación de los métodos especiales: el método constructor y el método main. 	Lectivas (L): <ul style="list-style-type: none"> Desarrollo del tema – 3 h Desarrollo de ejercicios - 1 h Ejercicios guiados – 3h 	7	4



	Laboratorio Diseño y programación de soluciones utilizando el método constructor y método main.	<ul style="list-style-type: none"> Desarrollo de ejemplos con los métodos especiales. 	Trabajo Independiente (T.I) <ul style="list-style-type: none"> Desarrollo de ejercicios – 4h 		
7	Primera sesión Atributos de clase y atributos de instancia. Forma de acceder a ambos tipos de atributos. Segunda sesión Métodos de clase y métodos de instancia. Forma de invocar a ambos tipos de métodos. Laboratorio Desarrollo de soluciones implementando clases que contengan miembros de clase y miembros de instancia.	<ul style="list-style-type: none"> Explicación de los atributos de clase y atributos de instancia. Explicación de los métodos de clase y métodos de instancia. Desarrollo de ejemplos con atributos y métodos de clase y de instancia. 	Lectivas (L): <ul style="list-style-type: none"> Desarrollo del tema – 3 h Desarrollo de ejercicios - 1 h Ejercicios guiados – 3h Trabajo Independiente (T.I) <ul style="list-style-type: none"> Desarrollo de ejercicios – 4h 	7	4
8	Examen parcial				

UNIDAD IV. Propiedades de la Teoría Orientada a Objetos: Herencia, Polimorfismo y Encapsulamiento. Vectores de Objetos

CAPACIDAD:

- Entiende y aplica la propiedad de Herencia, teniendo en cuenta que también es una forma de reutilización de código.
- Entiende y aplica la propiedad de Polimorfismo, entendiendo también los conceptos de métodos abstractos.
- Entiende y aplica la propiedad de Encapsulamiento.
- Usa el almacenamiento de objetos en vectores, para almacenar datos de diferente tipo de datos.

9	Primera sesión Encapsulamiento. Modificadores de acceso: público y privado. Métodos de acceso: set y get. Trabajar con 2 clases. Segunda sesión Ejercicios con encapsulamiento Laboratorio Desarrollo de soluciones implementando clases que contengan modificadores de acceso.	<ul style="list-style-type: none"> - Explicación del concepto de encapsulamiento y modificadores de acceso. - Desarrollo de ejemplos con encapsulamiento y modificadores de acceso. - Ejercicios de refuerzo. 	Lectivas (L): <ul style="list-style-type: none"> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h Trabajo Independiente (T.I) <ul style="list-style-type: none"> - Desarrollo de ejercicios – 4h 	7	4
---	---	--	--	---	---



10	Primera sesión Herencia, concepto, representación en diagrama UML, correlación con el lenguaje de programación. Segunda sesión Ejercicios combinados con Herencia Laboratorio Desarrollo de soluciones aplicando herencia entre dos o más clases.	<ul style="list-style-type: none"> - Explicación del concepto de herencia y su representación de clases. - Desarrollo de ejemplos con clases padre e hija. - Ejercicios de refuerzo. 	Lectivas (L): <ul style="list-style-type: none"> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h Trabajo Independiente (T.I)	7	4
----	--	---	---	---	---

			- Desarrollo de ejercicios – 4h		
11	Primera sesión Polimorfismo, concepto, representación en diagrama UML, correlación con el lenguaje de programación. Clase abstracta, método abstracto, clase interfaz. Segunda sesión Ejercicios combinados con Polimorfismo. Laboratorio Evaluación de Laboratorio N°02.	- Explicación del concepto de polimorfismo y su representación de clases. - Desarrollo de ejemplos con interfaces. - Ejercicios de refuerzo.	<u>Lectivas (L):</u> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h	7	4
12	Primera sesión Sobrecarga de métodos Segunda sesión Ejercicios combinados con sobrecarga de métodos. Laboratorio Desarrollo de soluciones aplicando polimorfismo y sobrecarga de métodos.	- Explicación de la sobrecarga de métodos. - Desarrollo de ejemplos con sobrecarga de métodos. - Ejercicios de refuerzo.	<u>Lectivas (L):</u> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h	7	4
13	Primera sesión Ejercicios combinados de todos los temas tratados. Segunda sesión Práctica Calificada 2 Laboratorio Desarrollo de soluciones de ejercicios combinados.	- Ejercicios de refuerzo. - Evaluación: práctica calificada 2.	<u>Lectivas (L):</u> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h	7	4

	Otras operaciones sobre un Vector de objetos: Modificación, eliminación y ordenamiento. Laboratorio Desarrollo de soluciones con operaciones de búsqueda, modificación, eliminación y ordenamiento de objetos almacenados en un Vector.	<ul style="list-style-type: none"> - modificación, eliminación y ordenamiento. - Desarrollo de ejemplos con vectores de objetos. - Ejercicios de refuerzo. 	<u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h		
15	Primera sesión Ejercicios de vectores de objetos y algoritmos para la gestión de datos. Segunda sesión Ejercicios combinados con vectores de objetos. Laboratorio Evaluación de Laboratorio N°03.	<ul style="list-style-type: none"> - Ejercicios de refuerzo. 	<u>Lectivas (L):</u> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h <u>Trabajo Independiente (T.I)</u> - Desarrollo de ejercicios – 4h	7	4
16	Examen Final.				
17	Entrega de promedios finales y acta del curso.				



14	Primera sesión Almacenamiento de objetos en un Vector. Visualización del contenido de un Vector de objetos. Búsqueda de elementos. Segunda sesión	<ul style="list-style-type: none"> - Explicación de Vectores de objeto, modo de ingreso de datos. Uso de los algoritmos de búsqueda, 	<u>Lectivas (L):</u> - Desarrollo del tema – 3 h - Desarrollo de ejercicios - 1 h - Ejercicios guiados – 3h	7	4
----	--	---	---	---	---

V. ESTRATEGIAS METODOLÓGICAS

- **Método Expositivo – Interactivo.** Comprende la exposición del docente y la interacción con el estudiante, empleando las herramientas disponibles en el aula virtual de la asignatura
- **Método de Discusión Guiada.** Conducción del grupo para abordar situaciones y llegar a conclusiones y recomendaciones, empleando las herramientas disponibles en el aula virtual de la asignatura
- **Método de Demostración – Ejecución.** Se utiliza para ejecutar, demostrar, practicar y retroalimentar lo expuesto, empleando las herramientas disponibles en el aula virtual de la asignatura

VI. RECURSOS DIDÁCTICOS - Equipos: Computadora.

- **Materiales:** Material elaborado por los docentes, prácticas dirigidas de laboratorio y textos (ver fuentes de consultas).
- **Lenguaje de Programación:** Java (Ing. Computación y Sistemas) y C# (Ing. Industrial) - **Software:** NetBeans IDE 8.2 (Ing. Computación y Sistemas) y Visual Studio (Ing. Industrial) - **Clases en línea.**
- **Tutoría en línea.**
- **Foros**

VII. EVALUACIÓN DEL APRENDIZAJE

$$PF = (PE+EP+EF) / 3$$

Donde:

PE = Promedio de evaluaciones

EP = Examen parcial

EF = Examen final

$$PE = 0.6*PPC + 0.4*PL$$

$$PL = (Lb1+Lb2+Lb3) / 3$$

$$PPC = (P1+P2) / 2$$

Donde:

P1, P2 = Práctica calificada

PPC = Promedio de practica calificada

Donde:

PL = Promedio de laboratorio

Lb1, Lb2, Lb3 = Evaluación de laboratorio

VIII. FUENTES DE INFORMACIÓN

8.1 Bibliográficas

- Ceballos Sierra, Francisco Javier (2013). Enciclopedia de Microsoft Visual C#. 4ª edición. Ed. RA-MA. México D.F.
- Dorman, Scott (2013). C# 5.0 y Visual C# 2012. Ed. Anaya Multimedia. Madrid.
- Hugon, Jérôme (2014). C# 5.0: Desarrolle aplicaciones Windows con Visual Studio 2013. Ediciones ENI. Barcelona.
- Flores Cueto, Juan José (2014). Método de las 6'D: modelamiento-algoritmo-programación. Ed. Macro. Lima.
- Deitel, Paul; Deitel, Harvey (2012). Cómo programar en java. 9na edición. Ed. Pearson. México D.F.
- Schildt, Herbert (2012). Java 7. Ed. Anaya Multimedia. Madrid.



IX. APOORTE DEL CURSO AL LOGRO DE LOS RESULTADOS DEL ESTUDIANTE

El aporte del curso al logro de los Resultados del Estudiante (*Student Outcomes*) en la formación del graduado en Ingeniería de Computación y Sistemas e Ingeniería Industrial, se establece en la tabla siguiente:

K = clave **R** = relacionado **Recuadro vacío** = no aplica

K = clave **R** = relacionado **Recuadro vacío** = no aplica

(a)	Habilidad para aplicar conocimientos de matemática, ciencia e ingeniería	R
(b)	Habilidad para diseñar y conducir experimentos, así como analizar e interpretar los datos obtenidos	
(c)	Habilidad para diseñar sistemas, componentes o procesos que satisfagan las necesidades requeridas	K
(d)	Habilidad para trabajar adecuadamente en un equipo multidisciplinario	R
(e)	Habilidad para identificar, formular y resolver problemas de ingeniería	K
(f)	Comprensión de lo que es la responsabilidad ética y profesional	
(g)	Habilidad para comunicarse con efectividad	R
(h)	Una educación amplia necesaria para entender el impacto que tienen las soluciones de la ingeniería dentro de un contexto social y global	
(i)	Reconocer la necesidad y tener la habilidad de seguir aprendiendo y capacitándose a lo largo de su vida	R
(j)	Conocimiento de los principales temas contemporáneos	
(k)	Habilidad de usar técnicas, destrezas y herramientas modernas necesarias en la práctica de la ingeniería	K
(l)	Habilidad de gestionar proyectos y demostrar el conocimiento y la comprensión de los principios de gestión en ingeniería y la toma de decisiones económicas, y su respectiva aplicación.	



