

SÍLABO

Taller de programación (100000I50N)

2024 - Ciclo 1 Marzo

1. DATOS GENERALES

1.1. Carrera:	Ingeniería de Sistemas e Informática Ingeniería Empresarial Ingeniería de Software Ingeniería de Redes y Comunicaciones Ingeniería de Seguridad y Auditoría Informática Ingeniería de Diseño Gráfico
1.2. Créditos:	3
1.3. Enseñanza de curso:	Presencial
1.4. Horas semanales:	4

2. FUNDAMENTACIÓN

Este curso proporcionará al estudiante los conocimientos necesarios para desarrollar e implementar aplicaciones con base en el lenguaje de programación orientado a objetos – JAVA. Ello le permitirá desarrollar las habilidades de programación bajo el enfoque de objetos, que sirve de apoyo en el desarrollo de programas para los cursos de ciclos superiores.

3. SUMILLA

Este curso es de naturaleza teórico-práctica que describe los fundamentos de programación, las sentencias de programación y la programación orientada a objetos.

4. LOGRO GENERAL DE APRENDIZAJE

Al finalizar el curso el alumno aplica las técnicas de programación a través del uso del paradigma orientado a objetos a fin de implementar aplicaciones en un lenguaje de programación.

5. UNIDADES Y LOGROS ESPECÍFICOS DE APRENDIZAJE

Unidad de aprendizaje 1: Programación con Java: Fundamentos de programación, estructuras secuenciales, estructuras condicionales.	Semana 1,2,3,4 y 5
Logro específico de aprendizaje: Al finalizar la unidad el participante identifica los fundamentos de los lenguajes de programación y de las estructuras secuenciales y condicionales.	
Temario: <ul style="list-style-type: none"> Introducción a Java. JDK: Entorno de ejecución y bibliotecas de Java. Compilación y ejecución de Java. (JVM) El Lenguaje Java: Tipos de datos primitivos. Identificadores, variables, constantes. Asignación e inicialización de variables. Salida por pantalla: System.out.println(). Entrada mediante teclado: la clase Scanner Ejecución secuencial en Java. Operadores aritméticos, lógicos, relacionales. Salida por pantalla con formato: System.out.printf() Clase Math. Comentarios. (en línea/bloque) Estructura condicional: Operadores condicionales y de relación. Sentencia: IF ELSE. Estructura condicional anidada. Estructura condicional multiple. Sentencia: SWITCH, CASE Resolución de problemas con sentencias IF..ELSE, SWITCH, CASE 	
Unidad de aprendizaje 2: Programación con Java: Estructuras repetitivas y arreglos.	Semana 6,7,8,9,10 y 11

Logro específico de aprendizaje:

Al finalizar la unidad el participante aplica las estructuras de control repetitivas y arreglos para la solución de problemas de software.

Temario:

- Introducción a Estructuras Repetitivas en Java. Sentencia Repetitiva WHILE. Contadores, acumuladores.
- Sentencia Repetitiva DO .. WHILE.
- Sentencia Repetitiva FOR. Bucles anidados.
- Teoría de Arreglos. Conceptos, tipos de arreglos. Arreglos unidimensionales. Declaración, creación e inicialización. Recorrido y cálculos con elementos.
- Resolución de problemas con sentencias repetitivas utilizando arreglos como parámetros
- Arreglos bidimensionales. Declaración, creación e inicialización. Recorrido y cálculos con elementos. Utilización de arrays como parámetros.

Unidad de aprendizaje 3:

Introducción a la Programación Orientada a Objetos, manejo de excepciones, manejo de cadenas de texto en Java.

Semana 12,13,14,15,16,17 y 18

Logro específico de aprendizaje:

Al finalizar la unidad, el alumno aplica las técnicas de programación orientada a objetos, gestión de excepciones y manejo de cadenas de texto para la solución de problemas de software.

Temario:

- Introducción a la programación orientada a objetos (POO) Definición, estructura de la clase. Atributos, modificadores de acceso. Métodos getter & setter. Instanciación de Objetos. Constructores. Sobrecarga. Puntero this.
- Métodos, parámetros, sobre escritura. Métodos de instancia y static.
- Modularidad: Creación de Procedimientos usando métodos de la clase Main. Paso de parámetros.
- Resolución de problemas con programación orientada a objetos y métodos con paso de parámetros.
- La clase String y sus principales métodos. Encadenamiento de metodos de la clase String. La clase StringBuilder/StringBuffer y sus principales métodos.
- Manejo de excepciones: Sentencias Try, Catch y Finally. Errores y excepciones comunes. Propagación de excepciones. Debuging.

6. METODOLOGÍA

La metodología del curso está basada en exposiciones del profesor, resolución de ejercicios y desarrollo de programas, que se complementan con experiencias individuales de ejercicios prácticos. En la plataforma virtual de aprendizaje, el alumno encuentra material didáctico relacionado con cada uno de los temas que indica el sílabo. El alumno tiene una evaluación permanente mediante ejercicios y casos que el profesor indica, lo que ayuda al alumno a ejercitarse en el desarrollo de programas y estar preparado para las evaluaciones obligatorias. Cabe mencionar que la participación del estudiante en clase es muy importante para entender y relacionar los contenidos del curso con las realidades comunes en las empresas y organizaciones. Los principios de aprendizaje que este curso promueve son aprendizaje autónomo, aprendizaje basado en evidencias y aprendizaje colaborativo.

7. SISTEMA DE EVALUACIÓN

El cálculo del promedio final se hará de la siguiente manera:

$$(20\%)PC1 + (20\%)PC2 + (20\%)PC3 + (10\%)PA + (30\%)PROY$$

Donde:

Tipo	Descripción	Semana	Observación
PC1	PRÁCTICA CALIFICADA 1	5	Práctica calificada individual.
PC2	PRÁCTICA CALIFICADA 2	10	Práctica calificada individual.
PC3	PRÁCTICA CALIFICADA 3	15	Práctica calificada individual.
PA	PARTICIPACIÓN EN CLASE	17	Participación en clase individual.
PROY	PROYECTO FINAL	18	Proyecto final grupal.

Indicaciones sobre Fórmulas de Evaluación:

1. La nota mínima aprobatoria final es de 12.
2. En este curso, no aplica examen rezagado.

8. FUENTES DE INFORMACIÓN

Bibliografía Base:

- CEBALLOS SIERRA, F. J. *Java 2: lenguaje y aplicaciones*. IC Editorial. <https://tubiblioteca.utp.edu.pe/cgi-bin/koha/opac-detail.pl?biblionumber=30940>
- Prieto, Natividad. *Empezar a programar usando JAVA (2a. ed.)*. Editorial de la Universidad Politécnica de Valencia. <https://tubiblioteca.utp.edu.pe/cgi-bin/koha/opac-detail.pl?biblionumber=36578>

Bibliografía Complementaria:

No hay bibliografía

9. COMPETENCIAS

Carrera	Competencias específicas
Ingeniería de Sistemas e Informática	<ul style="list-style-type: none"> • Soluciones Informáticas • Análisis de Sistemas
Ingeniería de Software	<ul style="list-style-type: none"> • Soluciones Informáticas • Desarrollo de Software • Análisis de Sistemas
Ingeniería de Redes y Comunicaciones	<ul style="list-style-type: none"> • Soluciones Informáticas • Análisis de Sistemas
Ingeniería de Seguridad y Auditoría Informática	
Ingeniería de Diseño Gráfico	<ul style="list-style-type: none"> • Proyectos de la Industria Gráfica • Gráfica, Web y Multimedia
Ingeniería Empresarial	<ul style="list-style-type: none"> • Gestión de las Tecnologías de la Información
Ingeniería de Sistemas e Informática	<ul style="list-style-type: none"> • Soluciones Informáticas • Análisis de Sistemas
Ingeniería Empresarial	<ul style="list-style-type: none"> • Gestión de las Tecnologías de la Información
Ingeniería de Software	<ul style="list-style-type: none"> • Soluciones Informáticas • Desarrollo de Software • Análisis de Sistemas

10. CRONOGRAMA DE ACTIVIDADES

Unidad de aprendizaje	Semana	Sesión	Tema	Actividades y evaluaciones
	1	1	Introducción a Java. JDK: Entorno de ejecución y bibliotecas de Java. Compilación y ejecución de Java. (JVM) El Lenguaje Java: Tipos de datos primitivos. Identificadores, variables, constantes. Asignación e inicialización de variables. Salida por pantalla: System.out.println(). Entrada mediante teclado: la clase Scanner	<ul style="list-style-type: none"> • Presentación del curso y Sistema de evaluación. Prueba de Entrada (15 minutos). Ejercicios básicos con salida, estructuras secuenciales en el entorno JAVA.
	2	2	Ejecución secuencial en Java. Operadores aritméticos, lógicos, relacionales. Salida por pantalla con formato: System.out.printf() Clase Math. Comentarios. (en línea/bloque)	<ul style="list-style-type: none"> • Exposición del docente. Ejercicios básicos con salida, estructuras secuenciales en el entorno JAVA. Ejercicios con operaciones

Unidad 1 Programación con Java: Fundamentos de programación, estructuras secuenciales, estructuras condicionales				aritméticas. Ejercicios con String: equals, contains. Casos prácticos.
	3	3	Estructura condicional: Operadores condicionales y de relación. Sentencia: IF ELSE. Estructura condicional anidada.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con sentencias condicionales. Ejercicios con condicionales anidadas. Casos prácticos.
	4	4	Estructura condicional multiple. Sentencia: SWITCH, CASE	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con sentencias condicionales múltiples. Ejercicios con String: toUpperCase, toLowerCase, replace. Casos prácticos.
	5	5	Resolución de problemas con sentencias IF..ELSE, SWITCH, CASE	<ul style="list-style-type: none"> Exposición del docente. Elabora programas con sentencias condicionales
		6	Evaluación	<ul style="list-style-type: none"> PRÁCTICA CALIFICADA 1
	6	7	Introducción a Estructuras Repetitivas en Java. Sentencia Repetitiva WHILE. Contadores, acumuladores.	<ul style="list-style-type: none"> Desarrollo de programas con sentencias while. Ejercicios con manejo de Regex (Pattern & Matcher) como complemento a las validaciones.
	7	8	Sentencia Repetitiva DO .. WHILE.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con la sentencia Do .. While. Uso de las clases para manejo de Regex (Pattern & Matcher) como complemento a las validaciones. Casos prácticos.
	8	9	Sentencia Repetitiva FOR. Bucles anidados.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con la sentencia For y bucles anidados.

Unidad 2 Programación con Java: Estructuras repetitivas y arreglos				Ejercicios con String: format. Casos prácticos.
	9	10	Teoría de Arreglos. Conceptos, tipos de arreglos. Arreglos unidimensionales. Declaración, creación e inicialización. Recorrido y cálculos con elementos.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con arreglos unidimensionales. Ejercicios con StringBuilder: append. Casos prácticos.
	10	11	Resolución de problemas con sentencias repetitivas utilizando arreglos como parámetros	<ul style="list-style-type: none"> Exposición del docente. Elabora programas con arreglos unidimensionales.
		12	Evaluación	<ul style="list-style-type: none"> PRÁCTICA CALIFICADA 2
Unidad 3 Introducción a la Programación Orientada a Objetos, manejo de excepciones, manejo de cadenas de texto en Java	11	13	Arreglos bidimensionales. Declaración, creación e inicialización. Recorrido y cálculos con elementos. Utilización de arrays como parámetros.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con arreglos bidimensionales. Casos prácticos.
	12	14	Introducción a la programación orientada a objetos (POO) Definición, estructura de la clase. Atributos, modificadores de acceso. Métodos getter & setter. Instanciación de Objetos. Constructores. Sobrecarga. Puntero this.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con clases y objetos. Casos prácticos.
	13	15	Métodos, parámetros, sobre escritura. Métodos de instancia y static.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con clases y sobre escritura de métodos. Casos prácticos
	14	16	Modularidad: Creación de Procedimientos usando métodos de la clase Main. Paso de parámetros.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con metodos de la clase main y paso de parámetros. Casos prácticos.
	15	17	Resolución de problemas con programación orientada a objetos y métodos con paso de parámetros.	<ul style="list-style-type: none"> Exposición del docente. Elabora programas con clases y objetos. Métodos con paso de parámetros.
		18	Evaluación	<ul style="list-style-type: none"> PRÁCTICA CALIFICADA 3

	16	19	La clase String y sus principales métodos. Encadenamiento de metodos de la clase String. La clase StringBuilder/StringBuffer y sus principales métodos.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con manejo de cadenas de texto. Casos prácticos.
	17	20	Manejo de excepciones: Sentencias Try, Catch y Finally. Errores y excepciones comunes. Propagación de excepciones. Debuging.	<ul style="list-style-type: none"> Exposición del docente. Ejercicios básicos con manejo de excepciones. Casos prácticos.
			Evaluación	<ul style="list-style-type: none"> PARTICIPACIÓN EN CLASE
	18	21	Evaluación	<ul style="list-style-type: none"> PROYECTO FINAL