

CIBERTEC

VISIÓN: Ser la institución líder de educación superior técnica en el Perú con alcance a nivel nacional.

MISIÓN: Formar profesionales íntegros y competentes brindando una educación superior de alta calidad que contribuya al desarrollo económico y ambiental del país.

I. INFORMACIÓN GENERAL DEL CURSO

Curso : Algoritmos y Estructura de Datos (SP4683)
Ciclo : Segundo
Período : 2023
Horas : 2 Teoría + 3 Laboratorio
Carrera(s) : Administración y Sistemas, Análisis de Datos Empresariales, Arquitectura de Datos Empresariales

II. INTRODUCCIÓN

Algoritmos y Estructura de Datos es un curso que pertenece a la línea de programación y desarrollo de aplicaciones y se dicta en las carreras de Computación e Informática, Redes y Comunicaciones, y Administración y Sistemas. Brinda un conjunto de técnicas de programación que permite a los alumnos diseñar algoritmos apropiados y adquirir buenos hábitos de programación.

El curso es teórico-práctico y se basa en el paradigma de la programación orientada a objetos. En primer lugar, se inicia con la creación de paquetes, clases y objetos desarrollándose el concepto de encapsulamiento. Continúa con el manejo de arreglos. Se utiliza la clase ArrayList así como el manejo de archivos de texto.

III. METODOLOGÍA

El proceso de enseñanza-aprendizaje se basa en el aprendizaje a partir de la experiencia. Busca motivar al estudiante a través de situaciones cercanas a la realidad y propiciar la reflexión para la resolución de problemas en los que se aplican de forma práctica los conocimientos adquiridos. El aprendizaje del curso se consolida con el desarrollo de un proyecto de investigación aplicada asesorado por el docente.

Esta metodología contribuye a que el alumno sea protagonista de su aprendizaje individual y colaborativo mientras que el docente asume un rol de planificador, facilitador y guía, creando escenarios que permiten a los alumnos la adquisición de competencias profesionales. El curso aplica la metodología de taller. En ese sentido, las actividades que se realizan en clases exigen la activa participación del alumno y la constante práctica tanto en las sesiones de laboratorio como fuera de él. Inmediatamente después de la presentación de cada tema, el alumno debe transferir lo aprendido mediante ejercicios propuestos que se entregan cada semana.

IV. LOGRO DEL CURSO

Al término del curso, el alumno implementa los conceptos de encapsulamiento en la programación orientada a objetos durante la construcción de aplicaciones básicas (Java). Realiza mantenimiento de datos almacenados en arreglos de objetos y archivos planos.

V. RESULTADOS DE APRENDIZAJE DE LA CARRERA

Nro	Resultado de Aprendizaje de la Carrera	Aporte
RAC 1	Desarrollo de soluciones de software multiplataforma utilizando herramientas tecnológicas adecuadas	Indirecto
RAC 2	Contribución en el aseguramiento de la calidad de las soluciones informáticas	
RAC 3	Participación en la definición y diseño de las soluciones informáticas	
RAC 4	Contribución en la administración de los servicios y proyectos de TI	
RAC 5	Resolución de situaciones y orientación a resultados	Indirecto
RAC 6	Innovación y desarrollo de emprendimientos	
RAC 7	Compromiso con la actualización profesional y la mejora continua	
RAC 8	Capacidad de liderazgo y trabajo en equipo	Indirecto
RAC 9	Responsabilidad ética y profesional	
RAC 10	Comunicación asertiva	

VI. UNIDADES DE APRENDIZAJE

UNIDAD 1. Clases y objetos		Duración: 20 horas
Logro de la Unidad de Aprendizaje Al término de la unidad, el alumno crea clases y objetos aplicando el concepto de encapsulamiento y haciendo uso de los especificadores de acceso: <i>public</i> y <i>private</i> así como la referencia <i>this</i> del lenguaje Java. Aplica además los métodos básicos de la clase String.		
Capacidades	Conocimientos	
1. Aplica el concepto de encapsulamiento. 2. Usa los especificadores de acceso: <i>public</i> y <i>private</i> . 3. Emplea la referencia <i>this</i> , modificadores <i>static</i> y <i>final</i> del lenguaje Java; aplica sobrecarga. 4. Aplica los métodos básicos de la clase String.	Temario 1.1. Tema 1: Conceptos básicos de POO 1.1.1. Introducción 1.1.2. Clase 1.1.3. Objeto 1.1.4. Declaración y creación de objetos (operador new) 1.1.5. Acceso a los miembros de la clase (operador punto) 1.1.6. Paquete 1.1.7. Especificador de acceso public 1.1.8. Sentencia import 1.2. Tema 2: Control de acceso y encapsulamiento 1.2.1. Dirección de memoria de un objeto 1.2.2. Asignación entre referencias 1.2.3. Constructor 1.2.4. Creación e inicialización de objetos 1.2.5. Especificador de acceso private 1.2.6. Encapsulamiento 1.2.7. Métodos de acceso público: set/get 1.3. Tema 3: Miembros de clase, constantes y sobrecarga 1.3.1. Referencia this 1.3.2. Modificador static (elementos únicos) 1.3.3. Bloque de inicialización static 1.3.4. Sobrecarga 1.3.5. Uso del this en sobrecarga 1.3.6. Modificador final (constantes) 1.3.7. Librería 1.4. Tema 4: Clase String 1.4.1. Descripción 1.4.2. Métodos básicos de la clase String 1.4.3. Concatenación 1.4.4. Recorrido	

UNIDAD 2. Arreglo lineal		Duración: 20 horas
Logro de la Unidad de Aprendizaje Al término de la unidad, los alumnos manipulan arreglos lineales con tipos de datos primitivos.		
Capacidades	Conocimientos	
1. Manipula arreglos lineales con tipos de datos primitivos.	Temario 2.1. Tema 5: Conceptos y operaciones simples 2.1.1. Descripción 2.1.2. Declaración e inicialización 2.1.3. Declaración privada e inicialización 2.1.4. Recorrido 2.1.5. Remplazo 2.1.6. Operaciones públicas básicas 2.1.7. Operaciones públicas complementarias 2.2. Tema 6: Artificios y operaciones variadas 2.2.1. Descripción 2.2.2. Declaración, creación y reserva 2.2.3. Declaración privada, creación y reserva 2.2.4. Ingreso personalizado 2.2.5. Recorrido 2.2.6. Redimensionamiento 2.2.7. Método privado ampliarArreglo 2.2.8. Operaciones públicas básicas 2.2.9. Operaciones públicas complementarias 2.3. Tema 7: Artificios y operaciones especiales 2.3.1. Método privado Buscar 2.3.2. Método privado Intercambiar 2.3.3. Operaciones públicas complementarias 2.4. Tema 8: Artificios y operaciones complementarias 2.4.1. Método privado Eliminar 2.4.2. Método privado Insertar 2.4.3. Operaciones públicas complementarias	

UNIDAD 3. Clase ArrayList		Duración: 15 horas
Logro de la Unidad de Aprendizaje Al término de la unidad, los alumnos utilizan los métodos de la clase ArrayList para efectuar operaciones con objetos (ingresar, consultar, eliminar, modificar, listar entre otras).		
Capacidades	Conocimientos	
1. Utiliza los métodos de la clase ArrayList para efectuar operaciones con objetos (ingresar, consultar, eliminar, modificar, listar, entre otras).	Temario 3.1. Tema 9: Conceptos y operaciones simples 3.1.1. Descripción 3.1.2. Coleccionista de objetos distintos 3.1.3. Coleccionista de objetos iguales 3.1.4. Declaración privada y creación 3.1.5. Métodos básicos de la clase <i>ArrayList</i> 3.1.6. Operaciones públicas básicas 3.1.7. Operaciones públicas complementarias 3.2. Tema 10: Operaciones variadas 3.2.1. Métodos adicionales de la clase <i>ArrayList</i> 3.2.2. Operaciones públicas básicas 3.3. Tema 11: Mantenimiento 3.3.1. Diseño básico de un proyecto	

UNIDAD 4. Técnicas Avanzadas de POO	Duración: 6 horas
--	--------------------------

Logro de la Unidad de Aprendizaje

Al término de la unidad, los alumnos entienden el concepto de herencia y polimorfismo, la relación de generalización/especialización entre clases e interfaces, se emplean las técnicas de casting y clases abstractas en casos prácticos de herencia simple y múltiple.

Capacidades	Conocimientos
1. Implementa herencia simple utilizando superclases concretas y abstractas. 2. Implementa herencia múltiple a través de interfaces.	Temario 4.1. Tema 12: Herencia y Polimorfismo 4.1.1. Generalización/Especialización 4.1.2. Herencia 4.1.3. Relación “Es un” o “Es una” 4.1.4. Uso de “super” 4.1.5. Sobreescritura de métodos 4.1.6. Clases abstractas y métodos abstractos 4.1.7. Técnicas de casting 4.1.8. Polimorfismo y uso “instanceof” 4.2. Tema 13: Interfaces 4.2.1. Definición 4.2.2. Herencia Múltiple

UNIDAD 5. Archivos	Duración: 6 horas
---------------------------	--------------------------

Logro de la Unidad de Aprendizaje

Al término de la unidad, los alumnos utilizan las clases `BufferedReader`, `FileReader`, `PrintWriter` y `FileWriter` para almacenar la data en archivos de texto.

Capacidades	Conocimientos
1. Utiliza las clases <code>BufferedReader</code> , <code>FileReader</code> , <code>PrintWriter</code> y <code>FileWriter</code> para almacenar la data en archivos de texto.	Temario 5.1. Tema 14: Manejo de archivos de texto 5.1.1. Descripción 5.1.2. Clases <code>PrintWriter</code> y <code>FileWriter</code> 5.1.3. Escritura en un archivo de texto 5.1.4. Clases <code>BufferedReader</code> y <code>FileReader</code> 5.1.5. Método <code>split</code> de la clase <code>String</code> 5.1.6. Lectura de un archivo de texto 5.2. Tema 15: Mantenimiento 5.2.1. Diseño básico de un Proyecto con archivos de texto

VII. EVALUACIÓN

Fórmula del Curso:

$$\text{Promedio Final} = 15\% \text{ PROM(CT,2,0)} + 10\%(\text{CP1}) + 20\% \text{ PROM(CL,2,0)} + 25\%(\text{CL3}) + 15\%(\text{CF1}) + 15\%(\text{SP1})$$

Dónde:

- CT = Caso Teórico
- CP = Caso Teórico Parcial
- CL = Caso Práctico de Laboratorio
- CF = Caso Teórico Final
- SP = Sustentación de Proyecto

Cronograma:

TIPO DE EVALUACIÓN	SEMANA
CT1	06
CL1	06
CP1	08
CL2	10
AP1	11
CT2	12
CL3	14
SP1	15
CF1	16

Consideraciones:

- La nota mínima aprobatoria es 13.
- El curso considera el desarrollo de un Proyecto aplicado que se evalúa a través de un Avance de Proyecto (AP) y la correspondiente Sustentación de Proyecto (SP).
- Ninguna evaluación es susceptible de eliminación.
- El curso Sí considera la posibilidad de rendir un Examen Sustitutorio que reemplace una de las evaluaciones del curso, a excepción del Avance de Proyecto (AP) y la Sustentación de Proyecto (SP).
- La rendición del Examen Sustitutorio se realiza en la semana 17 y requiere una inscripción previa según el procedimiento que indique Secretaría Académica oportunamente.

VIII. BIBLIOGRAFÍA**Bibliografía Básica**

- JOYANES AGUILAR, Luís
2008 Fundamentos de programación: algoritmos, estructuras de datos y objetos.
Madrid, España: McGraw-Hill 2003
(005.1 JOYA/A 2008)
- LEWIS, JOHN
2006 Estructuras de datos con Java: diseño de estructuras y algoritmos.
Madrid: Pearson Educación, 2006.
(005.73 LEWI/E)

Bibliografía Complementaria

- DEITEL, Harvey
2008 Cómo programar en Java
México, D.F.: Pearson Educación.
(005.133J DEIT 2008)