



VIGILADA MINEDUCACIÓN

SYLLABUS DE LA ASIGNATURA

1. IDENTIFICACIÓN

Nombre asignatura	Fundamentos de Programación Orientada a Objetos
Código	103008
Departamento (área)	Ciencias Computacionales
Programa (s) en los que se ofrece	Ingeniería de sistemas
Modalidad de la asignatura	Presencial
Número de créditos	3
Horas de trabajo presencial/sincrónico	4
Horas de trabajo dirigido/tutoría sincrónica	2
Horas de trabajo independiente/asincrónico	3
Prerrequisitos	Lógica de programación
Año- Periodo académico	2024-2
Docente (s)	Jorge Iván Meza Martínez
E-mail docente (s)	jimezam@autonoma.edu.co

2. JUSTIFICACIÓN

En el campo de la programación existen diferentes paradigmas o formas de implementar soluciones computacionales, entre ellas Programación Estructurada, Orientada a Objetos (POO), Funcional, Lógica, Orientada a Eventos y Orientada a Aspectos.

La POO trata de representar la solución del problema de una forma cercana a la realidad, modelando los sistemas como conjuntos de objetos que se relacionan entre sí. Este paradigma de programación tiene importantes ventajas, entre las cuales se puede resaltar la modularidad y la mayor facilidad para reutilizar componentes.

En este curso se verán los conceptos fundamentales de la programación orientada a objetos y como éstos permiten modelar e implementar la solución computacional a un problema.

3. DESEMPEÑO ESPERADO

RESULTADOS DE APRENDIZAJE

- Aplica el paradigma de programación orientada a objetos para estructurar soluciones de software en piezas más simples y reutilizables, llevando a cabo procesos de análisis, diseño y codificación.
- Aplica los principios de encapsulamiento, abstracción, herencia y polimorfismo en el



VIGILADA MINEDUCACIÓN

SYLLABUS DE LA ASIGNATURA

desarrollo de soluciones de software basadas en el paradigma de programación orientada a objetos.

4. CONTENIDOS

1. Introducción al lenguaje Java (caso práctico)

- Motivación principal de su creación
- Fortalezas y debilidades del lenguaje
- Concepto de máquina virtual
- Herramientas de software
 - JDK / JRE
 - IDE (Netbeans)
- Palabras reservadas del lenguaje
- Compilar y ejecutar: "hola mundo"
- Consultar la documentación oficial

2. Introducción al paradigma OO

- Motivación principal
- Conceptos de clase y objeto
- Elementos de una clase
 - Atributos
 - Métodos
- Reglas de nombrado para los identificadores
- Modificador "final"
- Representación de las clases
 - Diagrama de clases
 - Lenguaje Java

3. Uso de clases y objetos en Java

- Identificación de clases y responsabilidades
- Implementación de las clases
- Instanciación de objetos
 - Declaración
 - Creación (operador "new")
- Valor "null"
- Invocación (operador punto)
 - De atributos
 - De métodos
- Referencias vs objetos

4. Paquetes

- Motivación principal
- Creación (package)



VIGILADA MINEDUCACIÓN

SYLLABUS DE LA ASIGNATURA

- Importación (`import`)
- Revisión al paquete `java.lang`
- Otros paquetes interesantes
 - `Random`
 - `Math`
 - `Date` y `Time`

5. Clasificación elementos de una clase

- Tipos de miembro de acuerdo a su alcance
 - De instancia
 - De clase (`static`)
- Tipos de métodos de acuerdo a su uso
 - Principal (`main`)
 - Constructores
 - Modificadores (`setters`)
 - Analizadores (`getters`)
 - Definidos por el usuario
- Pseudobjeto `"this"`
- Sobrecarga de métodos (*overloading*)

6. Ocultamiento de información

- Motivación principal
- Modificadores de acceso
 - `public`
 - `"package"`
 - `protected`
 - `private`
- Necesidad de *setters/getters*

7 Manejo de cadenas de texto

- Conceptos básicos
- Principales tareas
 - Creación
 - Recorrido
 - Acceso a los elementos
- Uso de la documentación oficial
- Principales métodos de la clase

8. Asociaciones simples

- Conceptos básicos
 - Utilidad
 - Multiplicidad
 - Dirección
- Notación en el diagrama de clases



VIGILADA MINEDUCACIÓN

SYLLABUS DE LA ASIGNATURA

- Implementación en Java

9. Agregación y composición

- Conceptos básicos
 - Diferencias
 - Verificación: "*todo x esta conformado por y*"
- Notación en el diagrama de clases
- Implementación en Java

10. Creación de una interfaz de usuario basada en texto

- Introducción a la separación de capas o lógicas
- Mostrar mensajes en pantalla (*salida estándar*)
 - Interpolación de variables
 - Carácteres "escapados"
- Leer mensajes del teclado (*entrada estándar*)

11. Arreglos dinámicos

- Diferencias con los arreglos estáticos
 - Clases de recubrimiento (*wrapper*)
- Creación
- Acceso y recorrido
- Eliminación de elementos

12. Herencia

- Conceptos básicos
 - Verificación: "*todo x es un y*"
- Notación en el diagrama de clases
- Implementación en Java
- Pseudobjeto "super"
- Requerimientos para su implementación
- Sobreescritura de métodos (*overriding*)

5. ESTRATEGIAS DE ENSEÑANZA

- El curso consiste semanalmente en 4 horas de clase presencial con el profesor, 2 horas de trabajo supervisado y al menos 3 horas de trabajo individual.
- La asistencia a clase y la realización de sus actividades, tanto evaluativas como de práctica, son obligatorias.
- Antes de cada clase los estudiantes deben realizar las lecturas y actividades especificadas en el aula del curso, se harán actividades evaluativas de dichas lecturas.



VIGILADA MINEDUCACIÓN

SYLLABUS DE LA ASIGNATURA

- Las sesiones presenciales del curso se concentrarán en la solución de dudas de las lecturas asignadas, la presentación magistral de los temas complementarios por parte del docente, el seguimiento de ejemplos y la realización de ejercicios prácticos que ayuden al estudiante a formalizar los temas tratados en cada módulo.
- La evaluación de los temas vistos se realizará a partir de los talleres y exámenes dispuestos en cada uno de los tres cortes del curso. Cada uno de ellos se encuentra planeado para experimentar los temas teóricos y prácticos que se espera que el estudiante domine al finalizar el curso.

Generalmente este curso cuenta con el apoyo de un estudiante monitor para reforzar los conceptos teóricos y prácticos de los estudiantes del curso. La asistencia a las monitorías no es obligatorio pero totalmente necesaria.



VIGILADA MINEDUCACIÓN

SYLLABUS DE LA ASIGNATURA

6. ESTRATEGIA DE EVALUACIÓN DEL APRENDIZAJE

Corte	Tipo de evaluación	% parcial	% total
1	Talleres	50%	30%
	Parcial	50%	
2	Talleres	50%	35%
	Parcial	50%	
3	Talleres	40%	35%
	Taller final	60%	

7. BIBLIOGRAFÍA

Requerida

- [1] BARNES, David y KOLLING, Michael. Programación orientada a objetos con Java: una introducción práctica usando BlueJ, 3 ed. Pearson Educación, 2007 [005.133 B175]
- [2] DEAN, Johnn y DEAN Raymond. Introducción a la Programación con Java. McGraw-Hill, 2009 [005.133 D315]
- [3] DEITEL, H.M. Cómo programar en Java, 7 ed. Pearson Educación, 2008. [005.13 D347]
- [4] GOSLING, James y KEN, Arnold. El lenguaje de Programación Java. Addison-Wesley. 1997. [005.262 A756]

Sugerida

- [5] JIMENEZ, Luz Elena. Conceptos Básicos de Programación con Java. Universidad ICESI, 2002
- [6] NAIR, Premchand. Java Programming Fundamentals: Problem Solving Through Object Oriented Analysis and Design. CRC Press, 2008. [005.133 N147]