



UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS  
FACULTAD DE INGENIERÍA  
PROYECTO CURRICULAR DE INGENIERÍA DE SISTEMAS



### I. IDENTIFICACIÓN DE LA ASIGNATURA

PROGRAMACIÓN AVANZADA	
CÓDIGO DE LA ASIGNATURA	200303
ÁREA	PROGRAMACIÓN
SEMESTRE	SEGUNDO
PLAN DE ESTUDIOS	1996 - Ajuste 2002
HORAS TOTALES POR SEMESTRE	96
HORAS TEÓRICAS	32
HORAS PRÁCTICAS	64
SEMANAS POR SEMESTRE	16
PRE REQUISITO	NINGUNO
CO REQUISITO	NINGUNO
CRÉDITOS	4

### II. CARACTERÍSTICAS DE LA ASIGNATURA

#### 1. CONCEPTOS PREVIOS REQUERIDOS:

- Principios básicos de orientación a objetos
- Desarrollo de problemas con abstracciones orientadas a objetos

#### 2. PUNTOS DE APOYO PARA OTRAS ASIGNATURAS:

- Estructura lógica conceptual basada en paradigmas de programación
- Herramienta fundamental para ingeniería de software.
- Herramienta fundamental para bases de datos
- Herramienta fundamental para Redes

#### 3. ASIGNATURAS RELACIONADAS

- Ingeniería de Software
- Bases de Datos
- Lógica
- Matemáticas Discretas

#### 4. CONTENIDO FUNDAMENTAL:

Unidad 1. Recomendaciones Y Estándares Para Codificación.  
Unidad 2. Diagrama De Clases.  
Unidad 3. Implementación De Clases Con Diseño.  
Unidad 4. Gestión Del Almacenamiento De Objetos  
Unidad 5. Relaciones Entre Clases, Paquetes Y Espacios De Nombres Como Subsistemas.  
Unidad 6. Gestión de Código y Pautas Para obtener calidad en los programas.  
Unidad 7. Programación Con Diseño Orientado A Objetos.  
Unidad 8. Otros Tópicos En Programación Orientada A Objetos.



**UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS**  
**FACULTAD DE INGENIERÍA**  
**PROYECTO CURRICULAR DE INGENIERÍA DE SISTEMAS**



**5. JUSTIFICACIÓN:**

La informática es una disciplina en la que generalmente se ven involucradas dos características importantes: la creatividad y el formalismo. No basta con conocer un lenguaje en el cual se codifiquen algunas o muchas líneas que puedan solucionar un problema, es más importante aún, plantear una solución informática con un programa bien diseñado que se convierta fácilmente en un software robusto. Existen otras áreas en las que se establecerá la forma en cómo se debe afrontar un problema, pero es la programación y el diseño de código la que finalmente plasmará la respuesta. Es necesario entonces establecer el rigor suficiente que garantice que un programa de software esté bien formado, y es la programación avanzada la que propone tal fin.

Esta materia estará enfocada al seguimiento riguroso necesario para el establecimiento de un programa de software bien formado, basado en los principios fundamentales para la constitución de una buena solución computacional. El lenguaje utilizado para el desarrollo de programación avanzada será aquel en el cual se pueda desarrollar el contenido conceptual del curso. Preferiblemente C++, java o C#.

**6. OBJETIVO GENERAL:**

Desarrollar los elementos conceptuales necesarios para hacer una buena elección tecnológica y filosófica, y con ella poder construir programas bien diseñados.

**7. OBJETIVOS ESPECÍFICOS:**

- Realizar programas computacionales basados en estándares, normas y principios de diseño y reutilización que permitan ofrecer soluciones eficaces.
- Establecer los criterios de selección de un lenguaje de programación que garanticen el desarrollo adecuado de un proyecto informático.
- Identificar propuestas de desarrollo importantes y robustas que garanticen una buena solución informática.
- Identificar algunas de las tendencias del modelo OO importantes para la solución de problemas más complejos y puntuales.
- Manejar elementos conceptuales de mayor profundidad sobre el modelo OO.

**8. ESTRATEGIAS METODOLÓGICAS:**

- La materia se desarrollará teórico práctica. Se hará primero conceptualización y luego prácticas de laboratorio sobre los conceptos previamente discutidos.

**9. CRITERIOS DE EVALUACIÓN:**

Proyecto Final		30%
Parcial	20%	
Laboratorios y exposiciones		30%
Apreciativa, asistencia, trabajos	20%	

**10. RECURSOS FÍSICOS REQUERIDOS:**

- Software
- Microsoft Visual studio . Net , Microsoft Visio.
  - JDK 1.4, BDK.
- Hardware y logísticos
- Laboratorios de cómputo

**11. PRÁCTICAS ESPECÍFICAS:**



**UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS**  
**FACULTAD DE INGENIERÍA**  
**PROYECTO CURRICULAR DE INGENIERÍA DE SISTEMAS**



**III. PARCELADO**

No	TEMA A DESARROLLAR	SEMANAS ACADÉMICAS															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1.	notaciones	X															
2.	Evolución de los esquemas de abstracción, El modelo orientado a objetos y sus divisiones, Evolución de los mecanismos de abstracción.	X	X														
3.	Modelamiento de clases, Modelamiento de interfaces, Modelamiento de relaciones, Modelamiento de sistemas.			X													
4.	Estudio de requerimientos, Heurística general para encontrar las clases, Otras fuentes de clases, Herramientas para el planteamiento y modelamiento de clases tarjetas CRC, Principio del Egoísmo, Clase-instancia-objeto, Principio de seguridad en una clase mediante el this, Control de acceso e interfaces de una clase, Acceso en clases e instancias, Atributos de una clase, Seguridad de los atributos, Ley de Remeter, Alcance de un atributo su valor inicial y su inicialización, Tipos de atributos según su cambiabilidad su valor inicial y su inicialización, Volatilidad en atributos y su inicialización, Contención en atributos, Tipos de almacenamiento en atributos, Multiplicidad en los atributos, Establecimiento de operaciones, Tipos de operaciones, Establecimiento de los argumentos para las operaciones, Alcance de las operaciones, Sobrecarga de operaciones, Tipos de clases, Interfaces.				X												
5.	Concepto e importancia de la gestión de objetos, Tipos de gestión, Recuperación del espacio de los objetos, Objetos alcanzables e inalcanzable, Problemática y necesidad de evitar pérdida de memoria, Recuperación de los objetos estáticos, Recuperación de los objetos en pila, el papel de las pilas en la recuperación de objetos en pila, Recuperación de los objetos libres o en montículo, Programación O.O con recolección, liberación o recuperación manual y automática.					X	X										
6.	Relación de asociación, Relación de generalización o herencia, Relación de dependencia, Relación de realización, Delegación con herencia, Delegación con asociación, Delegación con dependencias, Delegación con realización, Principio de la Burocracia Perfecta.							X	X								
7.	Establecimiento de sistemas y subsistemas vía espacio de nombres o paquetes,									X	X						



**UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS**  
**FACULTAD DE INGENIERÍA**  
**PROYECTO CURRICULAR DE INGENIERÍA DE SISTEMAS**



No	TEMA A DESARROLLAR	SEMANAS ACADÉMICAS															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	Anidación de subsistemas, Acceso en subsistemas.																
8.	Principios fundamentales para la gestión de código, Principios fundamentales para establece pautas de calidad en los programas.											X					
9.	El principio de abierto/cerrado, El principio de sustitución de Liskov, El principio de inversión de dependencia, El principio de separación de la interfaz, El principio de equivalencia reutilización/visión, El principio de cierre común, El principio de reutilización común, El principio de dependencia acíclica, El principio de las dependencias estables, El principio de las abstracciones estables.												X				
10.	Programación concurrente, Programación distribuida, Generalidades de arquitecturas distribuidas y servicios compartidos, Persistencia y el papel de la programación en las bases de datos, generalidades sobre programación en bases de datos y conectividad.													X	X	X	X

**IV. BIBLIOGRAFÍA**

AUTOR (ES)	TÍTULO	Editorial	Edición y/o año	Tipo*
Bertrand Meyer	Object-Oriented Software Constr	Prentice Hall		TR
Bjarne Stroustrup	El C++ lenguaje de programación segunda edición	Addison Wesley		TR
Ivor Horton	Beginning Java 2	Wrox		TC
Horstmann Cornell	Core Java 2 vol 1 y vol 2.	Prentice Hall.		TG
Ian F. Darw	Java Cookbook	O'Reilly		TR
Burton Harvey, Simon Robinson, Julian Templeman, Karli Watson	C# Programming	Wrox Press Ltda		TA
Paul Anderson and Gail Anderson	Navigating C++, and Object-Oriented Design.	Prentice Hall		TC
Kayshav Dattatri	C++ effective Object-Oriented Software Construction	Prentice Hall		TC
William H. Ford / William R Topp	Introduction to Computing Using C++ and Object Tecnology	Prentice hall		TC

TG: Texto Guía  
 TC: Texto Consulta  
 TR: Texto Referencia  
 TA: Texto Adicional