



DATOS DE IDENTIFICACIÓN DEL CURSO

DEPARTAMENTO:	Departamento de Ciencias Computacionales				
ACADEMIA A LA QUE PERTENECE:	Técnicas Modernas de Programación				
NOMBRE DE LA MATERIA:	Lenguajes de Programación Comparados				
CLAVE DE LA MATERIA:	CC208				
CARÁCTER DEL CURSO:	Básica Particular				
TIPO DE CURSO:	Curso				
No. DE CRÉDITOS:	11				
No. DE HORAS TOTALES:	80	Presencial	68	No presencial	12
ANTECEDENTES:	CC200 – Programación Orientada a Objetos				
CONSECUENTES:	CC307 – Programación Lógica y Funcional				
CARRERAS EN QUE SE IMPARTE:	Licenciatura en Informática Licenciatura en Ingeniería en Computación				
FECHA DE ULTIMA REVISIÓN:	Agosto 2009				

PROPÓSITO GENERAL

Este curso aborda los lenguajes de programación estudiándolos a partir de los principios teóricos de lenguajes formales, reglas gramaticales y reconocimiento de construcciones del lenguaje. Los lenguajes son clasificados de acuerdo a las características de sus reglas gramaticales así como de su paradigma de programación

OBJETIVO TERMINAL

El alumno conocerá las formas, y criterios de clasificación de lenguajes de programación, los elementos básicos que los constituyen y las herramientas necesarias para la creación y reconocimiento de lenguajes de programación.

CONOCIMIENTOS PREVIOS

Conocer las metodologías de programación estructurada y programación orientada a objetos.

HABILIDADES Y DESTREZAS A DESARROLLAR

Analizar, investigar, identificar

ACTITUDES Y VALORES A FOMENTAR

Puntualidad, responsabilidad

METODOLOGÍA DE ENSEÑANZA APRENDIZAJE



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERÍAS
DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN



Método	Método tradicional de exposición	Método Audiovisual	Aula Interactiva	Multimedia	Desarrollo de proyecto	Dinámicas	Estudio de casos	Otros (Especificar)
%	80	10	0	0	0	10	0	0



CONTENIDO TEMÁTICO

MODULO 1. Introduccción		22 HRS
El alumno conocerá los conceptos colaterales básicos para comprender, reconocer y construir lenguajes de programación analizando e identificando los conceptos vistos en cada tema		
1.1	Clasificación de los lenguajes de programación	4 HRS
El alumno conocerá las formas básicas de clasificación de los lenguajes de programación identificando los criterios y características de cada uno de los métodos utiliza para realizar la clasificación		
1.1.1	Estudio de los lenguajes de programación desde el punto de vista de las áreas de aplicación <i>OBJETIVO DEL SUBTEMA</i>	
1.1.2	Estudio de los lenguajes de programación desde el punto de vista de su nivel <i>OBJETIVO DEL SUBTEMA</i>	
1.1.3	Estudio de los lenguajes de programación desde el punto de vista de los paradigmas de programación <i>OBJETIVO DEL SUBTEMA</i>	
1.2	Lenguajes formales	1 HRS
<i>El alumno conocerá los elementos gramaticales básicos que componen a un lenguaje de programación desde el punto de vista de lenguajes formales</i>		
1.3	Características de los lenguajes de programación	1 HRS
<i>El alumno aprenderá los conceptos de las características utilizadas para evaluar a un lenguaje de programación identificando cada uno de ellos.</i>		
1.4	Diseño de un lenguaje de programación	16 HRS
<i>OBJETIVO DEL TEMA:</i>		
1.4.1	Sintaxis <i>OBJETIVO DEL SUBTEMA</i>	
1.4.1.1	Jerarquía de Chomsky El alumno aprenderá la clasificación general de	



		lenguajes según la jerarquía de Chomsky, así como las máquinas teóricas que se utilizan para reconocerlos, analizando la diferencia entre los niveles de la jerarquía y distinguiendo qué máquina debe aplicarse para reconocer cada tipo de lenguaje		
	1.4.1.2	Backus Naur Form		
		El alumno conocerá las formas básicas más usuales para la definición de lenguajes realizando construcciones de lenguaje utilizando estas notaciones		
	1.4.1.3	Árboles de reconocimiento		
		<i>OBJETIVO DEL SUBTEMA:</i>		
	1.4.1.4	Diagramas de sintaxis		
		<i>OBJETIVO DEL SUBTEMA:</i>		
	1.4.1.5	Autómatas		
		El alumno conocerá las máquinas teóricas utilizadas para el reconocimiento de lenguajes, entenderá su funcionamiento aplicándolas al reconocimiento de lenguajes previamente definidos		
	1.4.1.5.1	Autómata finito		
		<i>OBJETIVO DEL SUBTEMA:</i>		
	1.4.1.5.2	Autómata descendente		
		<i>OBJETIVO DEL SUBTEMA:</i>		
	1.4.1.5.3	Autómata lineal limitado		
		<i>OBJETIVO DEL SUBTEMA:</i>		
	1.4.1.5.4	Máquina de Turing		
		<i>OBJETIVO DEL SUBTEMA:</i>		
	1.4.2	Semántica y sus documentos		
		El alumno conocerá los elementos semánticos que se encuentran en los lenguajes así como los documentos que los definen identificándolos conceptualmente		
	1.4.3	Proceso de compilación		
		El alumno comprenderá los pasos que comprenden el proceso de compilación de un programa identificando cada uno de ellos, el orden en que se llevan a cabo y en qué consisten		
	1.4.4	Proceso de carga y enlace		
		El alumno comprenderá los pasos que comprenden el proceso de carga y enlace de un programa identificando cada uno de ellos, el orden en que se llevan a cabo y en qué consisten		



MODULO 2. Elementos de un lenguaje de programación		14 HRS
El alumno conocerá los elementos básicos que constituyen a los lenguajes de programación, analizando e identificando los conceptos vistos en cada tema		
2.1	Abstracción	2 HRS
	<i>OBJETIVO DEL TEMA</i>	
2.2	Tipos de datos y declaraciones	8 HRS
	<i>OBJETIVO DEL TEMA</i>	
2.2.1	Tipos de datos y su clasificación	
	<i>OBJETIVO DEL SUBTEMA</i>	
2.2.1.1	Tipos de dato primitivo	
	El alumno conocerá los tipos de dato primitivos, sus conceptos, su representación en memoria identificando conceptualmente y clasificando cada tipo de acuerdo a sus características	
2.2.1.2	Tipo de dato estructurado	
	El alumno conocerá los tipos de dato estructurados, sus conceptos, su representación en memoria identificando conceptualmente y clasificando cada tipo de acuerdo a sus características	
2.2.2	Variables y parámetros	
	El alumno aprenderá los conceptos y tipos de variable y parámetros, identificando las similitudes y diferencias entre estos	
2.2.3	Ligaduras	
	El alumno aprenderá el concepto de ligadura y conocerá los distintos tipos de ligadura que existen realizando la clasificación de éstas según su tipo y tiempo	
2.2.4	Tiempo de vida y alcance	
	El alumno conocerá el concepto de tiempo de vida de una variable así como su alcance, determinando qué tiempo de vida y qué alcance corresponde a cada variable según su tipo	
2.3	Expresiones	2 HRS
	El alumno conocerá el concepto de evaluación de expresiones y las notaciones que se utilizan para llevar a cabo su evaluación	



	identificando los tipos de notación que existen y en que consisten		
2.3.1	Evaluación de expresiones		
	<i>OBJETIVO DEL SUBTEMA</i>		
2.3.2	Notaciones de expresiones		
	<i>OBJETIVO DEL SUBTEMA</i>		
2.4	Funciones		2 HRS
	<i>OBJETIVO DEL TEMA:</i>		
2.4.1	Evaluación de funciones		
	El alumno conocerá el concepto de evaluación de funciones y los diferentes tipos de evaluación de acuerdo al orden en que se realizan identificando conceptualmente cada tipo de función según su lugar en la evaluación		
2.4.2	Recursividad		
	El alumno aprenderá el concepto de recursividad los tipos de recursividad mas communes identificando cada tipo de recursividad		
MODULO 3. Paradigma de programación imperativa			12 HRS
El alumno comprenderá las similitudes y diferencias existentes entre los lenguajes pertenecientes al estilo de programación del paradigma imperativo identificando las características que determinan la pertenencia de un lenguaje a un determinado paradigma			
3.1	Paradigma estructurado en bloques		4 HRS
	El alumno aprenderá los antecedentes que definieron el estilo de programación estructurada en bloques, así como los conceptos generales de este estilo, identificando las características y conceptos que permiten clasificar a un lenguaje como parte de este paradigma		
3.1.1	Antecedentes generales de la programación modular		
	<i>OBJETIVO DEL SUBTEMA:</i>		
3.1.2	Conceptos generales de la programación modular		
	<i>OBJETIVO DEL SUBTEMA:</i>		
3.2	Paradigma estructurado a objetos		4 HRS
	<i>El alumno aprenderá los antecedentes que definieron el estilo de programación orientada a objetos, así como los conceptos generales de este estilo, identificando las características y</i>		



	<i>conceptos que permiten clasificar a un lenguaje como parte de este paradigma</i>		
	3.2.1	Antecedentes generales e la programación orientada a objetos	
		<i>OBJETIVO DEL SUBTEMA</i>	
	3.2.2	Conceptos generales de la programación orientada a objetos	
		<i>OBJETIVO DEL SUBTEMA</i>	
MODULO 4. Paradigma de programación declarativa			12 HRS
<i>El alumno comprenderá las similitudes y diferencias existentes entre los lenguajes pertenecientes al estilo de programación del paradigma declarativo identificando las características que determinan la pertenencia de un lenguaje a un determinado paradigma</i>			
4.1	Paradigma lógico		4 HRS
	<i>El alumno aprenderá los antecedentes que definieron el estilo de programación lógica, así como los conceptos generales de este estilo, identificando las características y conceptos que permiten clasificar a un lenguaje como parte de este paradigma</i>		
	4.1.1	Antecedentes generales de la programación lógica	
		<i>OBJETIVO DEL SUBTEMA:</i>	
	4.1.2	Conceptos generales de la programación lógica	
		<i>OBJETIVO DEL SUBTEMA:</i>	
4.2	Paradigma funcional		4 HRS
	<i>El alumno aprenderá los antecedentes que definieron el estilo de programación funcional, así como los conceptos generales de este estilo, identificando las características y conceptos que permiten clasificar a un lenguaje como parte de este paradigma</i>		
	4.2.1	Antecedentes generales de la programación funcional	
		<i>OBJETIVO DEL SUBTEMA</i>	
	4.2.2	Conceptos generales de la programación funcional	
		<i>OBJETIVO DEL SUBTEMA</i>	
4.3	Paradigma de bases de datos		4 HRS
	<i>El alumno aprenderá los antecedentes que definieron el estilo de programación para bases de datos, así como los conceptos</i>		



	<i>generales de este estilo, identificando las características y conceptos que permiten clasificar a un lenguaje como parte de este paradigma</i>		
4.3.1	Antecedentes generales de los lenguajes de programación para bases de datos OBJETIVO DEL SUBTEMA		
4.3.2	Conceptos generales de los lenguajes de programación para bases de datos OBJETIVO DEL SUBTEMA		
MODULO 5. Tendencias actuales y futuras			8 HRS
<i>El alumno analizará las similitudes y diferencias de algunos lenguajes contemporáneos pertenecientes a otros paradigmas contra los lenguajes pertenecientes a los paradigmas imperativos y declarativos</i>			
5.1	Lenguajes visuales		4 HRS
	<i>El alumno aprenderá los antecedentes que definieron el estilo de programación estructurada en bloques, así como los conceptos generales de este estilo, identificando las características y conceptos que permiten clasificar a un lenguaje como parte de este paradigma</i>		
5.1.1	Antecedentes de los lenguajes visuales OBJETIVO DEL SUBTEMA:		
5.1.2	Conceptos generales de los lenguajes visuales OBJETIVO DEL SUBTEMA:		
5.2	Lenguajes basados en scripts		4 HRS
	<i>El alumno aprenderá los antecedentes que definieron el estilo de programación visual, así como los conceptos generales de este estilo, identificando las características y conceptos que permiten clasificar a un lenguaje como parte de este paradigma</i>		
5.2.1	Antecedentes de los lenguajes OBJETIVO DEL SUBTEMA		
5.2.2	Conceptos generales de los lenguajes OBJETIVO DEL SUBTEMA		
MODULO 6. Comparación de lenguajes			6 HRS
<i>El alumno aplicará los conocimientos adquiridos para comparar algunos lenguajes entre sí determinando las diferencias y similitudes en razón de la sintaxis utilizada por</i>			



<i>éstos y el paradigma al que pertenecen</i>		
6.1	Comparar algunos lenguajes desde el punto de vista de elementos, áreas de aplicación y estilo de programación	4 HRS
	<i>El alumno comprenderá las diferencias existentes entre algunos lenguajes de programación de diferente tipo, identificando las diferencias de acuerdo a los criterios de clasificación vistos en clase</i>	
6.2	Investigar sobre otros lenguajes de programación	2 HRS
	<i>El alumno conocerá la existencia de otros lenguajes de programación diferentes a los analizados en clase analizándolos y clasificándolos conforme a los criterios vistos en el curso</i>	

CRITERIOS DE EVALUACIÓN

Evaluación en periodo ordinario

Primer examen departamental	25%
Segundo examen departamental	25%
Examen Global	30%
Trabajos de investigación:	
Tema 1.4.1.3	2.5%
Tema 1.4.2	2.5%
Tema 2.2.1.2	2.5%
Tema 2.2.3	2.5%
Tema 3.2.2	2.5%
Tema 4.3.2	2.5%
Tema 5.2.2	2.5%
Tema 6.2	2.5%
	100%

Para que el alumno tenga derecho al registro del resultado final de la evaluación en el periodo ordinario, se requiere tener un mínimo de asistencia del 80% a clases y actividades registradas durante el curso.

Evaluación en periodo extraordinario

La evaluación en periodo extraordinario deberá versar sobre el contenido global de la materia, y se calificará atendiendo a los siguientes criterios:

- La calificación obtenida en periodo extraordinario, tendrá una ponderación del 80% para la calificación final;
- La calificación obtenida por el alumno durante el periodo ordinario, tendrá una ponderación del 40% para la calificación en periodo extraordinario, y
- La calificación final para la evaluación en periodo extraordinario será la que resulte de la suma de los puntos obtenidos en las fracciones anteriores.



Para que el alumno tenga derecho al registro de la calificación en el periodo extraordinario, se requiere: tener un mínimo de asistencia del 65% a clases y actividades registradas durante el curso.

BIBLIOGRAFÍA

BÁSICA

TITULO	AUTOR	EDITORIAL	AÑO DE EDICIÓN	% DE COBERTURA DEL CURSO
Lenguajes de Programación, paradigma y práctica	Appleby, Doris.	McGrawHill, México,	1998	
Lenguajes de Programación	Pratt, Terrence W.	Prentice Hall, México	1997	
Lenguajes de Programación	Sethi, Ravi.	Addison Wesley	1992	
Lenguajes de Programación. 2da. Edición.	Tucker, Allen.	McGraw Hill.	1987	
Programming Language Essentials.	Bal, Henry.	Addison-Wesley Pub Co,	1999	
Programming Languages: Principles and Paradigms.	Tucker, Allen.	McGraw-Hill. Science/Engineering/Math,	2001	

COMPLEMENTARIA

TITULO	AUTOR	EDITORIAL	AÑO DE EDICIÓN	% DE COBERTURA DEL CURSO
Unix	Abrahams, Paul.	Addison Wesley, Estados Unidos,	1994	
Pascal	Koffman, Elliot.	Addison Wesley,	1996	
El Lenguaje de Programación C.	Kernighan, Brian.	Prentice Hall,	1988	
Estructuras de Matemáticas Discretas para la Computación.	Kolman, Bernard.	Prentice Hall,	1986	
Teoría de la Computación, Lenguajes Formales y Complejidad.	Brookshear, J. Glenn.	Addison Wesley,	1993	
Borland C++ 3.1 Programación Orientada a Objetos.	Faison, Ted.	Prentice Hall,	1992	
Programación Orientada a Objetos.	Cox, Bradd.	Addison Wesley, Estados Unidos,	1993	
Lisp. Addison Wesley,	Winston, Patrick Henry.	Estados Unidos		
Inteligencia Artificial.	Winston, Patrick Henry.	Addison Wesley, Estados Unidos		
Teoría de autómatas y lenguajes formales.	Kelley, Dean.	Pearson	1995	
Lenguajes, gramáticas y autómatas.	Cases	Alfaomega.		



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERÍAS
DIVISIÓN DE ELECTRÓNICA Y COMPUTACIÓN



REVISIÓN REALIZADA POR:

NOMBRE DEL PROFESOR	FIRMA
Alfredo Gutiérrez Hernández	
María Magdalena Murillo Leño	

Vo.Bo. Presidente de Academia

Lic. Sabrina Lizbeth Vega
Maldonado

Vo.Bo. Jefe del Departamento

Dr. Marco Antonio Pérez Cisneros

miércoles, 15 de julio de 2009