

Universidad de Colima
Facultad de Ingeniería Mecánica y Eléctrica
Facultad de Ingeniería Electromecánica
Licenciatura en Ingeniería en Sistemas Computacionales

PROGRAMA ANALÍTICO

I. DATOS GENERALES

MATERIA: Administración de centros de cómputo		UBICACIÓN: 6º Semestre
Antecedentes: Administración y recursos humanos.	Paralelas: Ética y normatividad jurídica.	Consecutivas: Tópicos de contabilidad y manejos financieros.
PLAN	CLAVE	CRÉDITOS
E903	-	7
HORAS	SEMANA	SEMESTRE
Teóricas:	2	36
Prácticas:	3	54
Total:	5	90

Elaborado por:	M.C. Oswaldo Carrillo Zepeda. L.I. Luis Arvizu Amescua.
Fecha:	Agosto/2006.

II. PRESENTACIÓN

La presente materia les dará información acerca de los centros de cómputo, de los departamentos que pueden o deben componer un centro de cómputo, de las funciones que cada uno de ellos deberán realizar, así como del perfil profesional que deberán tener las personas que en determinado momento puedan ocupar esos puestos.

Así como también contiene información acerca de la norma que rigen las leyes de derechos de autor, garantías y restricciones acerca del uso del hardware y el software.

También contiene información acerca de los principales problemas que se pudieran presentar en un centro de cómputo.

III. PROPÓSITO DEL CURSO

Estudiar las unidades informáticas desde la óptica de las organizaciones. Para ello se aplicarán y particularizarán los conceptos generales de éstas a las

unidades de informática, sean parte de una organización mayor o empresas independientes.

IV. CONTENIDOS PROGRAMÁTICOS

Objetivo por unidad	Contenidos
El alumno conocerá todos los aspectos informáticos de una organización.	UNIDAD I. Unidades de informática en las organizaciones <ol style="list-style-type: none"> 1.1. Misión y visión. 1.2. Función y objetivos. 1.3. Ubicación en la organización. 1.4. Reorganización de la unidad de informática. 1.5. Funciones informáticas centralizadas, distribuidas, independientes. 1.6. Funciones de compras, mantenimiento, integración de soluciones, desarrollo. 1.7. Vínculos formales e informales de la unidad informática con los demás órganos. 1.8. Personal informático para las unidades de informática.
El alumno aprenderá a analizar los factores que involucren a los proyectos informáticos de cualquier empresa.	UNIDAD II. Empresas y proyectos informáticos <ol style="list-style-type: none"> 2.1. Tipos de empresas de servicios informáticos. 2.2. Principios rectores para el desarrollo de tecnología nacional: <ol style="list-style-type: none"> 2.2.1. adaptación, 2.2.2. adecuación, 2.2.3. autonomía, 2.2.4. independencia. 2.3. Empresas pequeñas: <ol style="list-style-type: none"> 2.3.1. la figura de los socios-técnicos; 2.3.2. la figura de los dueños-directores. 2.4. Recursos y servicios externos (outsourcing). 2.5. Captación de necesidades. 2.6. Definición de productos y servicios para el mercado. 2.7. Productos a la medida. 2.8. Productos para consumo final.
El alumno aprenderá a	UNIDAD III. Administración de unidades

administrar las unidades informáticas tanto de hardware como de software.	informáticas <ol style="list-style-type: none"> 3.1. Planeación de la unidad. 3.2. Análisis de requerimientos y definición de los servicios de la unidad. 3.3. Determinación de volúmenes de información a manejar. 3.4. Hardware y software: selección, licitaciones, compra, renta, licencias de uso. 3.5. Normas de operación. 3.6. Controles. 3.7. Elaboración de reportes para los diferentes niveles jerárquicos de la organización.
El alumno adquirirá los conocimientos para llevar a cabo un buen control administrativo, sobre los bienes informáticos de la empresa.	UNIDAD IV. Bienes informáticos <ol style="list-style-type: none"> 4.1. Hardware y software adquirido, rentado, bajo licencia de uso. 4.2. Contratos y licencias. 4.3. Mantenimiento de equipos. 4.4. Seguros especiales para hardware, para software, para la información particular de la organización. Resguardo y respaldo de la información.
El alumno adquirirá los conocimientos para manejar el personal informático de cualquier empresa.	UNIDAD V. Administración del personal informático <ol style="list-style-type: none"> 5.1. Manejo de personal: búsqueda, selección, capacitación, supervisión, actualización. 5.2. Departamentos internos. 5.3. Descentralización de funciones, equipos y personal. 5.4 Preparación formal en niveles profesional y técnico. 5.5. Captación y retención de personal calificado. 5.6. Políticas de sueldos y salarios. 5.7. Estímulos. 5.8. Penalizaciones por delitos informáticos.
El alumno comprenderá la importancia de la constante capacitación y adiestramiento del personal informático en las	UNIDAD VI. Actualización del personal informático <ol style="list-style-type: none"> 6.1. Actividades para la actualización continua

organizaciones.	<p>de la organización.</p> <p>6.2. Importancia y necesidad de la actualización bajo la forma de estudios formales, capacitación, entrenamiento.</p> <p>6.3. Políticas específicas para actualización. Inversión en actualización.</p>
El alumno conocerá los métodos de auditoria de los bienes informáticos y seguridad en los accesos.	<p>UNIDAD VII. Auditoría informática</p> <p>7.1. Concepto y metodologías generales para auditoría.</p> <p>7.2. Medios y herramientas para realizar auditoría. Obtención y análisis de la información.</p> <p>7.3. Dictamen.</p> <p>7.4. Evaluación de la definición de la unidad informática y de su misión.</p> <p>7.5. Evaluación de los sistemas.</p> <p>7.6. Equipos y sistemas.</p> <p>7.7. Desempeño.</p> <p>7.8. Accesos, protecciones, y seguros.</p> <p>7.9. Riesgos de ambientes de PC aisladas.</p> <p>7.10. Redes organizacionales: diseño y accesos.</p> <p>7.11. Contratos y licencias.</p> <p>7.12. Empresas auditoras.</p>

V. LINEAMIENTOS DIDÁCTICOS

Estrategias didácticas					
Discusión dirigida	*	Exposición	*	Corrillo	
Lluvia de ideas	*	Phillip 66		Demostración	
Debates	*	Discusión en pequeños grupos		Otra _____	
Mesa redonda		Lectura dirigida	*	Otra _____	
Experiencias de aprendizaje					
Investigación	*	Prácticas	*	Mapa conceptual	
Lectura		Resolución de problemas		Examen	*
Reporte de lectura		Ensayo		Tareas	*
Proyecto	*	Exposición	*	Otras _____	
Recursos didácticos					

Material impreso	*	Proyector multimedia	*	Videocasetera	
Material virtual		Proyector de acetatos		Láminas	
Pintarrón	*	Televisión		Fotocopias	*
Computadora	*	Otros		Otros_____	

VI. CRITERIOS DE EVALUACIÓN CONTINUA

Aspectos a evaluar	Ponderación		
	1er parcial	2ª parcial	3ª parcial
Examen escrito	50%	50%	50%
Examen oral	-	-	-
Examen práctico	-	-	-
Tareas	10%	10%	10%
Prácticas	-	-	-
Proyecto	-	-	-
Participación individual	-	-	-
Participación en equipo	10%	10%	10%
Ensayo	-	-	-
Investigación	30%	30%	30%
Otros _____	-	-	-
TOTAL	100%	100%	100%

VII. BIBLIOGRAFÍA

Bibliografía básica
Li, D. & Pérez Moreno, L. (1990). <i>Auditoría en centros de cómputo: objetivos, lineamientos y procedimientos</i> . (Quinta reimpresión 2001). México: Trillas.
Fine, L. & Aldrete Bernal, F. (1990) <i>Seguridad en centros de cómputo: políticas y procedimientos</i> . (Segunda edición). México: Trillas.
Deakin, R. (1988). <i>Bases de datos: uso y administración en centros de cómputo</i> . México: Trillas.
Hernández Jiménez, R. (1991) <i>Administración de centros de cómputo</i> . (Tercera edición). México: Trillas.
(1998). <i>Auditoría informática: un enfoque práctico</i> . México: Alfaomega grupo editor.
Hernández Hernández, E. (1995). <i>Auditoría en informática: un enfoque metodológico y práctico</i> . (Tercera reimpresión 1999). México: Compañía editorial continental.

Echenique García, J. (1990). <i>Auditoría en informática</i> . México: McGraw – Hill.
Bibliografía complementaria
Arroyo Luis. (1980). <i>Del bit a la telemática</i> . España: Ed. (1a. Ed). (278 Págs.). Alhambra.
Asociación de bibliotecarios de instituciones de enseñanza superior e investigación. U.A.S.L.P. y Conacit. (1986). <i>La interacción entre la biblioteca y la informática</i> . Ed. México: (280 Págs.).
Barber, Paul J. Legge, David. (1980). <i>Percepción e información</i> . México: (1a. Edición). Cecsacompañía editorial continental, S. A.
Links de Internet
http://www.monografias.com/trabajos11/admicomp/admicomp.shtml
http://html.rincondelvago.com/administracion-de-centros-de-sistema.html
http://www.sistema.itesm.mx/va/graduados/plan99/sinteticos/sin-si.html

Prácticas de laboratorio:

Horas de utilización de infraestructura computacional:
144 horas / semestre.

Universidad de Colima
Facultad de Ingeniería Mecánica y Eléctrica
Licenciatura en Ingeniería en Sistemas Computacionales

PROGRAMA ANALÍTICO

I. DATOS GENERALES

MATERIA: Bases de datos distribuidas		UBICACIÓN: 6º Semestre
Antecedentes: Manejadores de base de datos, Redes de área local.	Paralelas: Compiladores y traductores.	Consecutivas: Ninguna.
PLAN	CLAVE:	CRÉDITOS:
E903	-	7
HORAS	SEMANA	SEMESTRE
Teóricas:	2	36
Prácticas:	3	54
Total:	5	90

Elaborado por:	M.C. Martha E. Evangelista Salazar, M.C. Juan García Virgen.
Fecha:	Agosto/2006.

II. PRESENTACIÓN

El conocimiento de las características de cada modelo de datos permitirá al alumno evaluar sistemas manejadores de base de datos tanto para su adquisición, como su rendimiento; además establecerá las bases teóricas para el diseño y desarrollo de bases de datos distribuidas.

III. PROPÓSITO DEL CURSO

El alumno conocerá y comprenderá los diferentes aspectos de control de concurrencia, definirá estrategias de recuperación adecuadas y diseñará esquemas de seguridad e integridad de las bases de datos distribuidas. El alumno conocerá y comprenderá un enfoque de las técnicas utilizadas en el diseño de manejadores de base de datos, así como los principios de las bases de datos distribuidas.

IV. CONTENIDOS PROGRAMÁTICOS

Objetivo por unidad	Contenidos
El alumno conocerá los fundamentos teóricos básicos de las Bases de Datos Distribuidas.	UNIDAD I. Introducción a las bases de datos distribuidos 1.1.- El contexto de las bases de datos distribuidas. 1.2.- Comparación entre Bases de Datos distribuidas y Bases de Datos Centralizadas. 1.3.- Objetivos de una Base de Datos Distribuida. 1.4.- Modelo Cliente/Servidor. 1.5.- Problemas de los Sistemas Distribuidos. 1.6.- Soporte del SQL.
El alumno realizará trabajos y prácticas que le ayudarán a experimentar y diseñar aplicaciones distribuidas.	UNIDAD II. Diseño de bases de datos distribuidas 2.1.- Introducción. 2.2.- Consideraciones para la Distribución de una Base de Datos. 2.3.- Tipos de Fragmentación. 2.3.1.- Fragmentación Horizontal. 2.3.2.- Fragmentación Vertical. 2.3.3.- Fragmentación Mixta. 2.3.4.- Asignación de Fragmentos. 2.3.5.- Modelo de Asignación. 2.4.- Procesamiento de Consultas Distribuidas.
El alumno diseñará y desarrollará consultas distribuidas, empleando métodos y técnicas que le ayudarán a optimizar sus bases de datos.	UNIDAD III. Diseño y optimización de consultas 3.1.- Introducción a la Optimización de Consultas. 3.2.- Transformaciones Equivalentes. 3.3.- Métodos de Ejecución de Join. 3.4.- Optimización de Consultas. 3.5.- Conmutación de JOIN's y UNIONES.
El alumno será capaz de diseñar sistemas transacciones en base de datos distribuidas, resolviendo además los problemas que suelen presentarse en este	UNIDAD IV. Procesamiento de transacciones en bases de datos distribuidas 4.1.- Manejo de Transacciones. 4.2.- Control de Concurrencia.

tipo de implementaciones (conurrencia, seguridad, integridad, etc.).	4.3.- Recuperación. 4.4.- Seguridad. 4.5.- Integridad. 4.6.- Normas.
--	---

V. LINEAMIENTOS DIDÁCTICOS

Estrategias didácticas					
Discusión dirigida	*	Exposición		Corrillo	
Lluvia de ideas	*	Phillip 66		Demostración	
Debates	*	Discusión en pequeños grupos		Otra	
Mesa redonda	*	Lectura dirigida		Otra	
Experiencias de aprendizaje					
Investigación	*	Prácticas	*	Mapa conceptual	
Lectura	*	Resolución de problemas	*	Examen	*
Reporte de lectura	*	Ensayo		Otras	
Proyecto	*	Exposición	*	Otras	
Recursos didácticos					
Material impreso	*	Proyector multimedia	*	Videocasetera	
Material virtual	*	Proyector de acetatos		Láminas	*
Pintarrón	*	Televisión		Fotocopias	*
Computadora	*	Otros		Otros	

VI. CRITERIOS DE EVALUACIÓN CONTINUA

Aspectos a evaluar	Ponderación		
	1er parcial	2ª parcial	3ª parcial
Examen escrito	20%	30%	20%
Examen oral	-	-	-
Examen práctico	30%	-	-
Tareas	10%	10%	-
Prácticas	10%	10%	-
Proyecto	-	30%	50%
Participación individual	10%	5%	10%
Participación en equipo	10%	5%	10%
Asistencia	5%	5%	-

Ensayo	-	-	-
Investigación	5%	5%	10%
Otros _____	-	-	-
TOTAL	100%	100%	100%

VII. BIBLIOGRAFÍA

Bibliografía básica
<p>Ozsu M. Tamer. (1999). <i>Principles of distributed database systems</i>. USA Prentice hall.</p> <p>Umar. (2001). <i>Distributed computing and client-server systems</i>. Prentice hall.</p> <p>Silbertschatz, A., Korth Sudarshan S. (2002). <i>Fundamentos de bases de datos</i>. Cuarta edición. México. Mc Graw Hill.</p> <p>Piatini Mario, Castañón Adoración de Miguel. (1999) <i>Fundamentos y modelos de bases de datos</i>. (Segunda edición). México. Alfaomega ra-ma.</p> <p>Kroenke. (2000). <i>Procesamiento de bases de datos</i>. Octava edicion. Pearson, prentice hall.</p>
Bibliografía complementaria
<p>Hernández Barbosa Oscar Daniel. (2002). <i>Fundamentos de las base de datos distribuidas</i>. México.</p> <p>G. Hansen, J.Hansen. (1998) <i>Diseño y administración de base de datos</i>. Prentice hall.</p>
Links de Internet
<p>http://es.tldp.org/Postgresql-es/web/navegable/tutorial/sql-language.html</p> <p>www.iespana.es/tsdgeos/sql/basico.html</p> <p>http://www.cs.cinvestav.mx/SC/prof_personal/adiaz/Disdb/temario.html</p> <p>http://www.lawebdejm.com/prog/delphi/memoriavirtual.html</p> <p>http://sistemas.ing.ula.ve/%7Eibc/bd/index.html</p> <p>http://faea.uncoma.edu.ar/materias/tdbd/Apuntes.html</p>

Prácticas de laboratorio:
<ol style="list-style-type: none"> 1.- Diseño de una Base de Datos Centralizada. 2.- Diseño de una Base de Datos Distribuida. 3.- Fragmentación Horizontal y Vertical de una Base de Datos. 4.- Instalación de SQL Server. 5.- Implantación de una Base de Datos en SQL. 6.- Implantación del modelo Cliente/Servidor. 7.- Implementación de un código de seguridad al sistema (jerarquía de usuarios) 8.- Implementación de la seguridad e integridad de las bases de datos 9.- Base de datos con múltiples servidores 10.- Pruebas del sistema y revisión final.

Horas de utilización de infraestructura computacional:	
Horas prácticas curriculares h/s/m	: 3
Horas prácticas extracurriculares h/s/m	: 5
Horas por semestre	: 112

Universidad de Colima
Facultad de Ingeniería Mecánica y Eléctrica
Facultad de Ingeniería Electromecánica
Licenciatura en Ingeniería en Sistemas Computacionales

PROGRAMA ANALÍTICO

I. DATOS GENERALES

MATERIA: Compiladores y traductores		UBICACIÓN: 6º Semestre
Antecedentes: Programación de sistemas, Teoría de autómatas.	Paralelas: Bases de datos distribuidas, Ingeniería de software.	Consecutivas: Inteligencia artificial.
PLAN	CLAVE	CRÉDITOS
E903	-	7
HORAS	SEMANA	SEMESTRE
Teóricas:	2	36
Prácticas:	3	54
Total:	5	90

Elaborado por:	Dr. Nicandro Farías Mendoza, M.C. Juan García Virgen, M.C. Mónica Cobián Alvarado, M.C. Walter Alexander Mata López.
Fecha:	Agosto/2006.

II. PRESENTACIÓN

El desarrollo de Software de Sistemas, como los sistemas operativos o los compiladores, no es muy común en la mayoría de las aplicaciones que demandan las empresas. Este tipo de desarrollo se deja a las grandes organizaciones de desarrollo de software. Sin embargo, sí es importante estudiar cómo están estructurados internamente, para que el alumno tenga un panorama más amplio del campo de las ciencias computacionales y además comprenda el proceso que se lleva a cabo al convertir un programa fuente a un programa ejecutable.

El estudio de los compiladores y los traductores no es un campo fuera del ámbito de un ingeniero en sistemas computacionales. Es más bien un complemento que le ayudará a reforzar sus bases sobre la naturaleza de este tipo de software.

III. PROPÓSITO DEL CURSO

El alumno tendrá las bases sólidas sobre la naturaleza funcional de los compiladores y traductores que emplea en los lenguajes de programación y aplicará su filosofía para poder describir elementos de la realidad y poder implementar un compilador o traductor aplicando los conocimientos adquiridos en otras materias. De esta forma se lleva a cabo un aprendizaje integral.

IV. CONTENIDOS PROGRAMÁTICOS

Objetivo por unidad	Contenidos
El alumno conocerá y analizará la evolución de los compiladores así como los elementos que conforman un lenguaje de programación, tomando como base el lenguaje C.	UNIDAD I. Introducción 1.1.- Antecedentes históricos 1.2.- El lenguaje 1.2.1.- Operadores y expresiones 1.2.2.- Entrada y salida de datos 1.2.3.- Instrucciones de control 1.2.4.- Funciones 1.2.5.- Arrays 1.2.6.- Punteros 1.2.7.- Estructuras y uniones 1.2.8.- Archivos de datos 1.2.9.- Macros 1.3.- El preprocesador de C
El alumno aprenderá y aplicará los conceptos fundamentales relacionados con los compiladores y traductores, así como también las herramientas de software que se encuentran disponibles para desarrollar un compilador.	UNIDAD II. Conceptos fundamentales 2.1.- Código fuente 2.2.- Código ejecutable 2.3.- Compiladores 2.4.- Traductores 2.5.- Tabla de símbolos 2.6.- Autómatas 2.7.- Fases del proceso de compilación 2.8.- Herramientas para construir compiladores.
El alumno conocerá y analizará la importancia del análisis léxico y las acciones involucradas en dicha fase, así como también aprenderá a utilizar una herramienta que le permita llevar a cabo análisis léxico.	UNIDAD III. Análisis léxico 3.1.- Componentes léxicos 3.2.- Lenguajes 3.3.- Expresiones regulares 3.4.- Definiciones regulares 3.5.- Autómatas finitos no determinísticos 3.6.- Autómatas finitos determinísticos 3.7.- Recuperación de errores en el análisis léxico 3.8.- Uso de la herramienta LEX

El alumno conocerá y aplicará los diferentes métodos para llevar a cabo el análisis sintáctico. Además utilizará una herramienta que le permita llevar a cabo el análisis sintáctico.	UNIDAD IV. Análisis sintáctico 4.1.- Función principal 4.2.- Gramáticas independientes del contexto 4.3.- Gramáticas ambiguas 4.4.- Análisis sintáctico descendente 4.5.- Análisis sintáctico ascendente 4.6.- Análisis sintáctico por precedencia de operadores 4.7.- Recuperación de errores en el análisis sintáctico 4.8.- Uso de la herramienta YACC
El alumno estudiará los diferentes métodos para llevar a cabo el análisis semántico y de esta forma proporcionarle las bases necesarias para que pueda desarrollar un programa que permita llevar a cabo este tipo de análisis.	UNIDAD V.- Análisis semántico 5.1.- Función principal 5.2.- Comprobación de tipos estática 5.3.- Comprobación de tipos dinámica 5.4.- Sobrecarga de operadores y funciones 5.5.- Recuperación de errores en el análisis semántico
El alumno conocerá y analizará el proceso que se lleva a cabo para obtener código máquina de las diferentes instrucciones de un lenguaje de programación.	UNIDAD VI. Generación de código 6.1.- Lenguaje ensamblador 6.2.- Generación de código objeto 6.3.- Generación de código ejecutable
El alumno integrará todo el proceso llevado a cabo durante el curso a una aplicación computacional apoyando su proyecto en los elementos vistos en esta unidad.	UNIDAD VII. Diseño de un compilador 7.1.- Especificación del lenguaje 7.2.- Especificación de la gramática 7.3.- Especificación de la sintaxis 7.4.- Especificación del análisis léxico 7.5.- Especificación del análisis sintáctico 7.6.- Especificación del análisis semántico 7.7.- Especificación de la generación de código

V. LINEAMIENTOS DIDÁCTICOS

Estrategias didácticas					
Discusión dirigida	*	Exposición	*	Corrillo	

Lluvia de ideas	*	Phillip 66		Demostración	*
Debates	*	Discusión en pequeños grupos	*	Otra _____	
Mesa redonda	*	Lectura dirigida		Otra _____	
Experiencias de aprendizaje					
Investigación	*	Prácticas	*	Mapa conceptual	*
Lectura	*	Resolución de problemas	*	Examen	*
Reporte de lectura	*	Ensayo		Elaboración de material didáctico	*
Proyecto	*	Exposición	*	Otras _____	*
Recursos didácticos					
Material impreso	*	Proyector multimedia	*	Videocasetera	
Material virtual	*	Proyector de acetatos		Láminas	*
Pintarrón	*	Televisión		Fotocopias	*
Computadora	*	Otros		Otros _____	

VI. CRITERIOS DE EVALUACIÓN CONTINUA

Aspectos a evaluar	Ponderación		
	1er parcial	2ª parcial	3ª parcial
Examen escrito	30 %	30 %	20 %
Examen oral	10 %	-	-
Examen práctico	-	-	-
Tareas	10 %	10 %	10 %
Prácticas	20 %	20 %	20 %
Proyecto	10 %	10 %	30 %
Participación individual	10 %	10 %	10 %
Participación en equipo	10 %	10 %	5 %
Ensayo	-	-	-
Investigación	-	10 %	5 %
Otros _____	-	-	-
TOTAL	100 %	100 %	100 %

VII. BIBLIOGRAFÍA

Bibliografía básica

<p>Aho, Alfred V., Sethi, Ravi., Ullman, Jeffrey D. Et. Al. (1990). <i>Compiladores, principios, técnicas y herramientas</i>. (1ª reimpresión 1998). Addison-wesley Iberoamericana.</p> <p>Leland L. Beck. (1988). <i>Software de sistemas, understanding and writing compilers: a do it yourself guide</i>. México: Addison-wesley iberoamericana.</p> <p>Temblay & Sorenson. (1985). <i>Compilers writing</i>. USA: Ed. Mc Graw Hill.</p> <p>Guillermo Levine Gutiérrez; (2001). <i>Computación y programación moderna perspectiva integral de la informática</i>. México: Pearson educación.</p> <p>Fischer, LeBlanc., Et. Al. (1991). <i>Crafting a compiler with C</i>. USA: Cummings publishing company, Inc.</p> <p>Thomas Pittman, James Peters. (1992). <i>The art of compiler design theory and practice</i>; USA: Prentice hall.</p> <p>Jesús Salas Parrilla. (1988). <i>Sistemas operativos y compiladores</i>. Madrid: McGraw Hill.</p> <p>Beck, Leland L. (1988). <i>Software de sistemas, introducción a la programación de sistemas</i>. Madrid: Addison-wesley iberoamericana.</p> <p>Teufel, B., Schmidt, S., Teufel, T. & Morales Peake, E. (1995). <i>Compiladores conceptos fundamentales</i>. Caracas: Addison-wesley iberoamericana.</p> <p>Kenneth C. Loudon. (2004). <i>Construcción de compiladores principios y práctica</i>. México: Thomson.</p> <p>Kenneth C. Loudon. (2004). <i>Lenguajes de programación principios y práctica</i>. México: Thomson.</p> <p>Ronald Mak. (1996). <i>Writing compilers and interpreters</i>. (Segunda edición). New York: John Wiley & Sons.</p>
Bibliografía complementaria
<p>Gottfried, B. (1991). <i>Programación en c</i>. México: McGraw Hill.</p> <p>John R. Levine, Tony Mason, Doug Brown. (1992). <i>Lex y yacc</i>. Sebastopol, California: O'Reilly & associates.</p>
Links de Internet
<p>www.upseros.net/compiladores/compiladores.php</p> <p>www.ucse.edu.ar/fma/compiladores/</p> <p>www.monografias.com/trabajos11/compil/compil.shtml</p> <p>www.cgrsoftware.com/compiladores.htm</p> <p>www.angelfire.com/ar/CompiladoresUCSE/COMPILADORES.html</p>

Prácticas de laboratorio:
<ol style="list-style-type: none"> 1.- Análisis de los componentes de un compilador. 2.- Validación de Expresiones Regulares. 3.- Desarrollo de un analizador de léxico. 4.- Desarrollo de un analizador sintáctico. 5.- Generación de código intermedio. 6.- Optimización de Código. 7.- Generación de código final. 8.- Manejo de la tabla de símbolos. 9.- Integración de todas fases de un compilador en un proyecto.

Horas de utilización de infraestructura computacional:
4 horas semana – mes.

Universidad de Colima
Facultad de Ingeniería Mecánica y Eléctrica
Facultad de Ingeniería Electromecánica
Licenciatura en Ingeniería en Sistemas Computacionales

PROGRAMA ANALÍTICO

I. DATOS GENERALES

MATERIA: Ingeniería de Software		UBICACIÓN: 6º Semestre
Antecedentes: Multimedia.	Paralelas: Compiladores y traductores.	Consecutivas: Ninguna.
PLAN	CLAVE	CRÉDITOS
E903	-	7
HORAS	SEMANA	SEMESTRE
Teóricas:	2	36
Prácticas:	3	54
Total:	5	90

Elaborado por:	Dr. Nicandro Farías Mendoza.
Fecha:	Agosto/2006.

II. PRESENTACIÓN

La demanda de Software de alta calidad y de mayor complejidad se está incrementando a un ritmo sin precedentes, por esta razón se presenta la necesidad de aplicar un conjunto de herramientas, métodos y procedimientos estructurados para automatizar el Software de calidad y de costo efectivo. La Ingeniería de Software realiza un estudio de los principios y metodologías, mediante la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de sistemas de software. Considerando varios paradigmas sucesivos; en donde cada desarrollo sucesivo pretende hacer el proceso de la Ingeniería más sencillo o extender la complejidad de las aplicaciones posibles de construir. La ingeniería de Software busca obtener técnicas más eficientes y poderosas, especialmente para las aplicaciones de mayor demanda.

III. PROPÓSITO DEL CURSO

Presentar al alumno los diferentes modelos y paradigmas útiles para el desarrollo eficiente de un proyecto de software desde propuesta, evaluación, planeación, hasta el diseño e implementación y mantenimiento del mismo,

asegurando la calidad del producto obtenido.

IV. CONTENIDOS PROGRAMÁTICOS

Objetivo por unidad	Contenidos
Instruir al alumno con los conceptos, modelos, teorías y principios básicos estudiados en la Ingeniería de Software.	UNIDAD I. Principios de la ingeniería de software 1.1 Conceptos generales 1.1.1 Conceptos: Ingeniería, Software e Ingeniería de Software 1.1.2 Características del software 1.1.3 Componentes del software 1.1.4 Aplicaciones del software 1.1.5 Problemas del software 1.1.6 Herramientas que utiliza la ingeniería del software 1.1.7 Importancia de la ingeniería de software 1.1.8 Objetivos de la Ingeniería de Software 1.1.9 Características de la Ingeniería de Software 1.1.8 Resultados esperados de la aplicación de la ingeniería de software 1.2 Taxonomía de la Ingeniería de Software 1.2.1 Sistemas en Batch 1.2.2 Sistemas Reactivos 1.2.3 Sistemas de tiempo Real 1.2.4 Sistemas Distribuidos 1.3 Procesos del desarrollo de Software 1.3.1 Procesos de especificación 1.3.2 Procesos de implementación 1.3.3 Procesos de validación y verificación 1.4 Paradigmas del ciclo de vida del Software 1.4.1 Modelo del ciclo de vida 1.4.2 Modelo de cascada 1.4.3 Modelo incremental 1.4.4 Modelo de espiral 1.4.5 Modelo operacional 1.4.6 Modelo transformacional 1.5 Estándares de la Ingeniería de Software 1.5.1 Características de un buen estándar 1.5.2 Estándares de la IEEE para la

	<p>Ingeniería de Software</p> <p>1.5.3 Adopción de estándares</p>
<p>Aprender los procedimientos y modelos que habiliten al alumno para gestionar proyectos de software, considerando los aspectos que garanticen la calidad del proyecto.</p>	<p>UNIDAD II. Administración de proyectos de Software</p> <p>2.1 Proyectos de software</p> <p>2.2 Actividades de la administración</p> <p>2.3 Planeación del proyecto</p> <p>2.4 Administración de riesgos</p> <p>2.5 Control de calidad del Software</p> <p>2.5 Verificación y validación externa</p> <p>2.7 Estimación de costos del Software.</p>
<p>Estudiar los procedimientos para determinar y representar con precisión el problema que se propone resolver, garantizando que el modelo que representa la solución cumpla con las condiciones iniciales</p>	<p>UNIDAD III. Análisis de requerimientos del Software</p> <p>3.1 Determinación de los requerimientos del Software</p> <p>3.2 Procesos de la ingeniería de requerimientos</p> <p>3.3 Modelos del Sistema</p> <p>3.4 Construcción de prototipos de Software.</p> <p>3.5 Especificación Formal de prototipos</p> <p>3.5.1 Estándar Estelle</p> <p>3.5.2 Lógica Modal</p> <p>3.5.3 Lenguaje B</p> <p>3.5.4 Lenguaje Z.</p>
<p>Estudiar los Conceptos, Modelos y tecnologías que faculten al alumno para especificar de manera precisa el procedimiento para obtener una solución apropiada del problema que se esta resolviendo.</p>	<p>UNIDAD IV. Técnicas de diseño e implementación del software</p> <p>4.1 Conceptos básicos</p> <p>4.2 Abstracción, estructura, modularidad, concurrencia</p> <p>4.3 Diseño arquitectónico</p> <p>4.4 Características de un buen diseño</p> <p>4.5 Notaciones para el diseño de software</p> <p>4.5.1 Diagrama de flujo de datos</p> <p>4.5.2 Diagramas de flujo estructurados</p> <p>4.5.3 Pseudo código</p> <p>4.6 Técnicas de diseño</p> <p>4.6.1 Diseño estructurado</p> <p>4.6.2 Diseño Orientado a Objetos (OMT, UML, etc.)</p> <p>4.6.3 Diseño Orientado Agentes</p> <p>4.7 Diseño de Interfaces</p> <p>4.8 Programación de componentes e interfaces</p>

	4.9 Síntesis de componentes y módulos.
Aplicar los conceptos y procesos con los cuales garantizar que las posibles adecuaciones y reutilización de los componentes del sistema a desarrollar se lleven a cabo de manera eficiente.	UNIDAD V Mantenimiento y reusabilidad 5.1 Introducción al mantenimiento y reusabilidad del Software 5.2 Clasificación del Software para la reutilización 5.2.1 El proceso de la reutilización 5.2.2 Ingeniería de dominios 5.2.3 Construcción de componentes reutilizables 5.2.4 Clasificación y recuperación de componentes 5.2.5 Economía de la reutilización de software 5.3 Reingeniería de Software 5.3.1 Ingeniería Inversa 5.3.2 Reestructuración 5.3.3 Ingeniería progresiva 5.3.4 Economía de la reingeniería.
Estudiar los modelos y conceptos con los cuales comprobar que los resultados obtenidos son congruentes con respecto las descripciones iniciales	UNIDAD VI. Verificación y validación 6.1 Conceptos y técnicas de verificación y validación 6.1.1 Métodos formales y sistemas críticos 6.1.2 Validación de la fiabilidad 6.1.3 Afianzamiento de la seguridad 6.1.4 Valoración de la protección 6.2 Pruebas del Software 6.2.1 Pruebas de defectos 6.2.2 Pruebas de integración 6.2.3 Inspecciones del Software 6.2.4 Análisis estático 6.2.5 Bancos de trabajo de pruebas 6.4 Ingeniería del Software de sala limpia. 6.4.1 El enfoque de sala limpia 6.4.2 Especificación funcional 6.4.3 Refinamiento y verificación del diseño 6.4.4 Comprobación de sala limpia.
Investigar los conceptos modelos y procedimientos con los cuales el alumno este capacitado para analizar, evaluar y seleccionar las	UNIDAD VIII. Automatización de herramientas como estrategia para la productividad 8.1 Automatización del Software de prueba

herramientas y los procedimientos que le sirvan de apoyo para la producción de software de una manera eficiente.	8.2 Ingeniería de Software Asistida por Computadora (Herramientas CASE) 8.2.1 Tecnología CASE 8.2.2 Herramientas CASE existentes 8.2.3 Herramientas CASE emergentes 8.2.4 Tendencias futuras de las herramientas CASE 8.3 Las etapas hacia la productividad 8.4 La administración de la calidad 8.4.1 Aseguramiento y estándares de calidad 8.4.2 Planeación de la calidad 8.4.3 Control de la calidad 8.4.4 Medición y métricas de Software 8.5 Las metas de la administración de la calidad 8.6 Construyendo un sistema de administración de la calidad 8.7 La Ingeniería de la información.
--	---

V. LINEAMIENTOS DIDÁCTICOS

Estrategias didácticas					
Discusión dirigida	*	Exposición	*	Corrillo	*
Lluvia de ideas	*	Phillip 66	*	Demostración	*
Debates	*	Discusión en pequeños grupos	*	Otra _____	
Mesa redonda	*	Lectura dirigida	*	Otra _____	
Experiencias de aprendizaje					
Investigación	*	Prácticas	*	Mapa conceptual	*
Lectura	*	Resolución de problemas	*	Examen	*
Reporte de lectura	*	Ensayo	*	Elaboración de material didáctico	*
Proyecto	*	Exposición	*	Otras: tareas	*
Recursos didácticos					
Material impreso	*	Proyector multimedia	*	Videocasetera	
Material virtual	*	Proyector de acetatos		Láminas	*
Pintarrón	*	Televisión		Fotocopias	*
Computadora	*	Otros		Otros _____	

VI. CRITERIOS DE EVALUACIÓN CONTINUA

Aspectos a evaluar	Ponderación		
	1er parcial	2ª parcial	3ª parcial
Examen escrito	20 %	20 %	-
Examen oral	10 %	10 %	10 %
Examen práctico	20 %	20 %	20 %
Tareas	10 %	10 %	10 %
Prácticas	10 %	10 %	10 %
Proyecto	10 %	10 %	10 %
Participación individual	10 %	10 %	10 %
Participación en equipo	10 %	10 %	10 %
Ensayo	-	-	10 %
Investigación	-	-	10 %
Otros _____	-	-	-
TOTAL	100%	100%	100%

VII. BIBLIOGRAFÍA

Bibliografía básica
Bruegge B., Dutoit Allen H (2002). <i>Ingeniería de software orientada a objetos</i> . Prentice hall.
Grady Booch Robert(1998). <i>Practical software metrics for project management and process improvement</i> . Prentice hall.
Lawrence Pfleeger S. (2002). <i>Ingeniería de software teoría y práctica</i> . Prentice hall.
Pressman Roger (2002). <i>Ingeniería del software un enfoque práctica</i> . (Sexta edición). Mc Graw Hill.
Sommerville Ian (2002). <i>Ingeniería de Software</i> , Addison Wesley.
Yourdan, Edward (1999). <i>Análisis estructurado moderno</i> . Ed. Prentice-hall.
Jacobson,Ivar. (2000). <i>El Proceso unificado de desarrollo de software</i> . Ed. Addison wesley.
Fowler, Martin, (1999). <i>UML gota a gota</i> . Ed. Addison wesley.
Larman, Craig (1999). <i>UML y patrones</i> . Pearson.
Humphrey, Watts S. (2000). <i>Introducción al proceso software personal</i> . Ed. Addison wesley.
Pfleeger, Shari Lawrence (2002). <i>Ingeniería de software teoría y práctica</i> . Ed. Ptentice-hall.
Bruegge Bernd (2001). <i>Ingeniería de software orientada a objetos</i> . Ed. Prentice-

hall.

Braude, Eric (2003). *Ingeniería de software una perspectiva orientada a objetos*. Ed. Alfaomega.

Meyer, Bertrand (1999). *Construcción de software orientada a objetos*. Ed. Prentice hall.

Bibliografía complementaria

Fairley Richard. (1988). *Ingeniería de software*. México: Mc Graw Hill.

Williams David. (1994). *Tools and techniques for structured systems analysis and design*. Prentice Hall.

Keyes Jessica. (1993). *Software engineering productivity handbook*, USA: Mc Graw-Hill.

Sodhi Jag. (1991). *Software engineering methods, management and case tools*. Mc Graw-Hill.

Wasserman Anthony, (1988). *Tutorial on software design techniques*. (Tercera edición). Freeman peterand, -IEEE computer society.

Links de Internet

ISO/IEC 15939:2001 Information technology – Software engineering – Software measurement process.

<http://www.sei.cmu.edu>

<http://www.byte.com>

<http://dmi.uib.es/~bbuades/calidad/index.htm>

<http://www.fing.edu.uy/inco/cursos/gestsoft/ppts/GS02.PPT>

Prácticas de laboratorio:

1. Estándares de la Ingeniería de Software
2. Prototipos formales de Software
3. Especificación de prototipos con UML
4. Diseño de prototipos con UML
5. Validación de Software.
6. Aplicación de herramientas case

Horas de utilización de infraestructura computacional:

5 horas semana-mes.

Universidad de Colima
Facultad de Ingeniería Mecánica y Eléctrica
Facultad de Ingeniería Electromecánica
Licenciatura en Ingeniería en Sistemas Computacionales

PROGRAMA ANALÍTICO

I. DATOS GENERALES

MATERIA: Interconectividad de redes		UBICACIÓN: 6° Semestre
Antecedentes: Redes de área local.	Paralelas: Ninguna.	Consecutivas: Telemática.
PLAN	CLAVE	CRÉDITOS
E903	-	7
HORAS	SEMANA	SEMESTRE
Teóricas:	2	36
Prácticas:	3	54
Total:	5	90

Elaborado por:	M.C. Juan García Virgen, M.C. Juan Antonio Díaz Hernández, Esp. Dora Magaly Mejía Carrasco, M.C. Ruth Berenice Castillo Vega, M.C. Oswaldo Carrillo Zepeda.
Fecha:	Agosto/2006.

II. PRESENTACIÓN

La necesidad de configurar y operar adecuadamente la infraestructura de red se ha convertido en un requerimiento estratégico para las organizaciones. Muchos de los procesos críticos de las empresas se basan en el uso de redes de datos y ahora se realizan en línea (on-line). La adecuada configuración y operación de los dispositivos de interconectividad constituye en un factor clave de éxito para lograr que las transacciones se realicen de forma electrónica y para obtener mejoras en el tiempo y en los costos asociados con su ejecución.

III. PROPÓSITO DEL CURSO

Al finalizar este curso el alumno conocerá los componentes y tecnologías de redes de área amplia, así como los principios básicos de operación de los distintos dispositivos que constituyen la infraestructura de una red. Será capaz de configurar y operar adecuadamente cada uno de estos dispositivos de acuerdo con los requerimientos de su empresa. Además podrá integrar diferentes tecnologías de red en una sola infraestructura proporcionando

diferentes niveles de servicio, seguridad e integridad a cada uno de ellos.

IV. CONTENIDOS PROGRAMÁTICOS

Objetivo por unidad	Contenidos
El alumno conocerá los estándares y tecnologías para redes de área amplia.	UNIDAD I. Servicios de comunicaciones 1.1.- Estándar ITU H.320. 1.2.- Redes de alta velocidad. 1.3.- Frame Relay para transmisión integral. 1.4.- ATM (Asynchronous Transfer Mode). 1.5.- Redes RDSI (Red Digital de Servicios Integrados) o ISDN (Integrated Services Digital Network).
El alumno comprenderá que es una red WAN. Estudiará sus características más importantes y conocerá los protocolos, servicios, etc., para aplicarlos en el desarrollo y solución de problemas prácticos.	UNIDAD II. Redes amplias (WAN) 2.1.- Dispositivos DTE (Data Terminal Equipment) y DCE (Data Communications Equipment). 2.2.- Capas 3 y 4 de OSI. 2.3.- Medios de transmisión (cableado, microondas, satélite, radio). 2.4.- Mecanismos de acceso al medio para WAN. Protocolos orientados a conexión: X.25, Frame Relay, ATM. 2.5.- Protocolos no orientados a conexión: TCP/IP, SNA. 2.6.- Conexiones punto a punto y multipunto. 2.7.- Conmutación y ruteo 2.8.- Características y servicios para WAN 2.9.- Conceptos WAN 2.9.1 Microondas analógicas y digitales. 2.9.2 Tipos de satélites. Espacios satelitales. Microondas satelitales. 2.9.3 Redes públicas digitales. Redes telefónicas. 2.9.4 Arquitecturas. MAN (Metropolitan Area Network), 2.9.5 WAN (Wide Area Network. 2.9.6 PBX (Private Branch Exchange). Redes amplias mediante PBX.
El alumno integrará los conocimientos adquiridos en las unidades anteriores e iniciará con el diseño de redes de largo alcance haciendo uso	UNIDAD III. Redes Interconectadas 3.1.- Consideraciones de diseño. 3.2.- Análisis de la generación de tráfico producida por los diferentes protocolos.

de los recursos computacionales con que se cuenta y será capaz de diseñar redes para otras aplicaciones, considerando todas sus variantes.	<p>3.3.- Sistemas heterogéneos.</p> <p>3.4.- Administración y supervisión.</p> <p>3.5.- Protocolo SNMP (Simple Network Management Protocol).</p> <p>3.6.- Redes globales.</p> <p>3.7.- Precursoras: ARPAnet. NSFnet, USEnet, MILnet. NIC (Network Information Center).</p> <p>3.8.- Internet.</p> <p>3.9.- Evolución de las redes privadas.</p>
El alumno estudiará y comprenderá teorías, protocolos, procedimientos, esquemas y aspectos relacionados con la interconexión de redes y los aplicará en el planteamiento, diseño y desarrollo de problemas.	<p>UNIDAD IV. Teoría de interconexiones</p> <p>4.1.- Teoría de grafos.</p> <p>4.2.- Protocolos para resolución de direcciones: ARP (Address Resolution Protocol) y RARP (Reverse Address Resolution Protocol) Subredes y máscaras. DNS (Domain Name System).</p> <p>4.3.- Dominio Unix.</p> <p>4.4.- Teoría de enrutamiento. Spanning Tree y enrutamiento de origen.</p> <p>4.5.- Enrutamientos estático, dinámico, exterior e interior.</p> <p>4.6.- Protocolos de enrutamiento: camino más corto, múltiple, centralizado, aislado, distribuido y jerárquico.</p>
El alumno estudiará y comprenderá los aspectos que pueden influir en la integridad de una red y en base a ello será capaz de buscar, probar e implementar software para resolver problemas.	<p>UNIDAD V. Integridad</p> <p>5.1.- El papel de las capas OSI superiores en la transmisión confiable de información.</p> <p>5.2.- Inserción de puntos de verificación por la capa de sesión.</p> <p>5.3.- Conversión de formatos.</p> <p>5.4.- Estándar ASN.1 (Abstract Syntax Notation).</p> <p>5.5.- Métodos para garantizar la integridad en transmisión de textos comprimidos.</p> <p>5.6.- Codificación dependiente del contexto.</p> <p>5.7.- Comunicación confiable entre aplicaciones: traps.</p> <p>5.8.- Software de supervisión.</p>
El alumno estudiará los diferentes elementos de interconexión que lo llevarán a configurar protocolos de	<p>UNIDAD VI. Elementos de interconexión</p> <p>6.1.- Configuración con RIP (Routing Information Protocol), IGRP (Interior</p>

<p>enrutamiento, programar routers, manejar protocolos, experimentar distintas arquitecturas de red mediante la instalación y configuración de diferentes sistemas operativos, para mediante el desarrollo de prácticas, diagnosticar y solucionar los problemas de conectividad más comunes.</p>	<p>Gateway Rout-ing Protocol) y OSPF (Open Shortest Path First).</p> <p>6.2.- Diseño de una red mediante enrutadores.</p> <p>6.3.- Enrutadores multiprotocolo.</p> <p>6.4.- Intercomunicación entre distintas arquitecturas.</p> <p>6.5.- Sistemas operativos de red.</p>
<p>El alumno estudiará los diferentes mecanismos de seguridad en redes y los aplicará en casos prácticos. Además, será capaz de desarrollar utilerías propias para encriptación y cifrado de datos. Experimentará también con software existente en el mercado.</p>	<p>UNIDAD VII. Seguridad</p> <p>7.1.- Seguridad en un sitio.</p> <p>7.2.- Seguridad de extremo a extremo.</p> <p>7.3.- Codificación y ciframiento de la información (encryption).</p> <p>7.4.- Criptografía: llave secreta, llave pública, certificados de llaves públicas.</p> <p>7.5.- Estándares ISO para codificación DES (Data Encryption Standard).</p> <p>7.6.- Autentificación: código de acceso y confirmación de identidad.</p> <p>7.7.- Firmas digitales.</p> <p>7.8.- Software de protección (firewall y proxy).</p>

V. LINEAMIENTOS DIDÁCTICOS

Estrategias didácticas					
Discusión dirigida		Exposición	*	Trabajo en equipo	*
Lluvia de ideas		Solucionar casos prácticos		Visitar empresas	
Debates		Phillip 66		Desarrollo de un proyecto	
Mesa redonda		Investigar aplicaciones en su entorno social	*	Otra _____	
Experiencias de aprendizaje					
Investigación	*	Prácticas	*	Mapa conceptual	
Lectura	*	Resolución de problemas		Examen	*
Reporte de lectura		Ensayo		Plantear ejemplos	*
Proyecto		Exposición	*	Otras _____	
Recursos didácticos					
Material impreso	*	Proyector multimedia	*	Videocasetera	

Material virtual	*	Proyector de acetatos	Láminas	
Pintarrón	*	Televisión	Fotocopias	*
Computadora	*	Otros	Otros_____	

VI. CRITERIOS DE EVALUACIÓN CONTINUA

Aspectos a evaluar	Ponderación		
	1er parcial	2ª parcial	3ª parcial
Examen escrito	40%	40%	20%
Tareas	10%	10%	10%
Prácticas	30%	30%	-
Proyecto	-	-	70%
Participación individual	10%	10%	-
Participación en equipo	10%	10%	-
TOTAL	100%	100%	100%

VII. BIBLIOGRAFÍA

Bibliografía básica
<p>Douglas, Comer. (2000). <i>Interconectividad de redes con tcp</i>. Purdue university.</p> <p>Roldan, José Manuel David y Huidobro. (2003). <i>Integración de voz y datos: call centers, tecnología y aplicaciones</i>. España: (1ª edición). McGraw Hill.</p> <p>Comer David L., Douglas E. y Stevens (2000). <i>Interconectividad de redes con tcp/ip: diseño e implementación</i>. (1ª edición, tomo 2). México. Prentice hall.</p> <p>VV.AA. (2000). <i>Interconectividad manual para resolución de problemas: una obra de consulta esencial para resolver problemas redes</i>. (1ª edición). México:</p> <p>Academia de networking de cisco systems. (2004). España: Ciscopress. Guía del Primer año. (Segunda edición).</p>
Bibliografía complementaria
<p>Cheswick. W. (1994). <i>Firewalls and internet security</i>. (1ª edición). USA: Addison-wesley.</p> <p>practical unix security (1991). (1ª edición). Garfinkel, S. & Associates Inc.</p> <p>Sheldom, T. Lan times – enciclopedia de redes. (1ª edición). Mc Graw Hill.</p>
Links de Internet
<p>www.cisco.netacad.net</p> <p>www.mundopc.net/cursos/redes/</p> <p>http://enciclopedia.us.es/index.php/Redes_de_computadoras</p> <p>telematica.cicese.mx/internetII/octmm/sld025.htm</p> <p>tecnicyce.com.mx/diplomados.php?id=1</p>

www.infochannel.com.mx/mundos22.asp?id_nota=9517&tecnologia=5
www.mundotutoriales.com/tutorial_interconectividad-mdtutorial1048224.htm

Prácticas de laboratorio:

- | |
|---|
| <ol style="list-style-type: none">1.- Elaborar un diagrama que muestre los servicios proporcionados por el sistema telefónico.2.- Conexiones punto a punto y multipunto utilizando switches y hubs.3.- Configuración de una red con TCP/IP.4.- Configuración de Ruteadores con protocolos, direcciones IP y subneteo.4.- Diseño y configuración de un Proxy y Firewall con Windows.5.- Diseño y configuración de un Proxy y Firewall con Linux.6.- Desarrollo de una aplicación de Encriptamiento aplicado a una red.7.- Interconexión de una red con diferentes sistemas operativos.8.- Instalación y experimentación de diferentes tipos de software para monitorear una red Monitoreo de redes, verificando: tráfico, protocolos activos en la red, identificación de tipos de paquetes en la red, detección de intrusos, bloqueo de intrusos. |
|---|

Horas de utilización de infraestructura computacional:

50 horas al semestre.
