



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004
NEUQUÉN, 01 ABR. 1998

VISTO el expediente N° 29673/98 mediante el cual la Facultad de Economía y Administración solicita la creación de la carrera "Licenciatura en Ciencias de la Computación"; y,

CONSIDERANDO:

Que, la carrera se dicta actualmente en convenio con la Universidad Nacional del Sur, hasta el cuarto año;

Que, el Departamento de Informática y Estadística de la Facultad, se encuentra en condiciones académicas de implementar el quinto año;

Que, la implementación de la carrera completa de Licenciatura en Ciencias de la Computación en la Universidad Nacional del Comahue, no implica ninguna modificación en el convenio con la Universidad Nacional del Sur;

Que, la concreción de la carrera en esta Universidad redundará en beneficio de la región y para la comunidad universitaria;

Que, el Consejo Directivo de la Facultad de Economía y Administración, en su Resolución N° 261/97, solicita la aprobación de creación de la carrera "Licenciatura en Ciencias de la Computación";

Que, la Comisión de Docencia y Asuntos Estudiantiles emitió despacho aconsejando aprobar la creación de la carrera y su respectivo plan de estudios;

Que, el Consejo Superior, en su sesión ordinaria de fecha 12 de marzo de 1998, trató y aprobó el despacho producido por la Comisión;


Por ello:

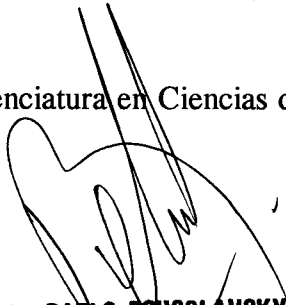
**EL CONSEJO SUPERIOR DE LA UNIVERSIDAD NACIONAL DEL COMAHUE
ORDENA:**

ARTICULO 1°: APROBAR la creación de la carrera "Licenciatura en Ciencias de la Computación" en el ámbito de la Facultad de Economía y Administración de la Universidad Nacional del Comahue.

ARTICULO 2°: APROBAR el plan de estudios de la carrera "Licenciatura en Ciencias de la Computación", que se incorpora a la presente como Anexo.

ARTICULO 3°: REGÍSTRESE, comuníquese y archívese.


CRISTINA S. JUHASZ
Secretaria del Consejo Superior
Universidad Nacional del Comahue


Lic. PABLO BOHOSLAVSKY
PRESIDENTE
CONSEJO SUPERIOR



Universidad Nacional del Comahue

Consejo Superior

01004

ORDENANZA N°

PLAN DE ESTUDIOS DE LA CARRERA LICENCIATURA EN CIENCIAS DE LA COMPUTACION

Fundamentación

Las Ciencias de la Computación han tenido un crecimiento explosivo desde su borroso punto de partida en las primeras décadas de este siglo. Desde sus comienzos mantuvo una conexión significativa con el formalismo lógico - matemático. Las aplicaciones fundamentales de sus comienzos fueron los procesos de cálculo para la resolución aproximada de problemas. Sin embargo, y aunque este área todavía hoy representa un segmento importante de las aplicaciones, éstas han delimitado a un área que requiere conocimientos matemáticos profundos lo que ha devenido en la aparición dentro de la Matemática de un área con problemática propia que utiliza a la computadora como herramienta y no como objetivo. Por otra parte, las Ciencias de la Computación han evolucionado hacia el tratamiento de problemas cuya resolución es de naturaleza simbólica, esencialmente no numérica. Este desarrollo ha dado cada vez más relevancia al formalismo lógico dentro de sus fundamentos.

El área del conocimiento estudiada por las Ciencias de la Computación posee una dinámica de desarrollo que provoca como efecto lateral la obsolescencia en los planes de estudio de las carreras asociadas. Estas carreras deben ser actualizadas de manera permanente para poder poner en el mercado laboral profesionales idóneos. Es necesario discutir brevemente esta dinámica puesto que es fácilmente confundida, por el no especialista, con el mero devenir de nuevas versiones en los paquetes de programación que ofrece el "establishment" comercial.

Las Ciencias de la Computación se encuentran en una etapa fundacional como ciencia. En las dos últimas décadas se han descubierto ciertos principios fundamentales con consecuencias trascendentes para toda la currícula. Lentamente, estas consecuencias han sido incorporadas en los cursos respectivos cuando esto fue posible. En los casos más extremos ciertos cursos debieron eliminarse o pasar a formar parte de cursos optativos, debiendo incluirse otros. Como se ha fundamentado más arriba, el área más notablemente afectada ha sido la de los métodos numéricos que formaron parte de la currícula elemental de las carreras de computación durante muchos años y lentamente han dejado lugar a otros temas que son ahora imprescindibles. La restricción de mantener la duración de la carrera en límites razonables ha obligado a eliminar muchos de estos cursos.

Uno de los principios que fueron reconocidos desde la delimitación epistemológica de la nueva ciencia es que existe una relación íntima entre la Lógica y las Ciencias de la Computación. Esta vinculación es doble. En principio y como hecho general, el razonamiento lógico es reconocidamente la base de cualquier ciencia. Por otra parte, y



Universidad Nacional del Comahue

Consejo Superior

ORDENANZA N° 01004

como relación particular, la lógica ha penetrado profundamente en las diferentes áreas de las Ciencias de la Computación dando bases científicas sólidas y permitiendo un desarrollo sostenido de las mismas. La contribución de la Lógica ha sido retribuida con expansiones de la propia Lógica realizadas por los investigadores en Ciencias de la Computación. Por esta razón, y como consecuencia natural, resulta imprescindible introducir los fundamentos del razonamiento lógico tempranamente en la currícula de las carreras de grado. Esta inyección permite la utilización del formalismo en otros cursos tales como los correspondientes a Bases de Datos, Métodos Formales de Desarrollo de Software, Inteligencia Artificial, Semántica de Lenguajes de Programación, etc.

El área de las Ciencias de la Computación reconocida como Programación ha visto el desarrollo de nuevos paradigmas para describir procesos computacionales, v.g., la programación orientada a objetos, la programación en lógica y el refinamiento y consolidación de la programación funcional. Esta evolución de alto contenido creativo ha motivado cambios en diferentes cursos de la currícula tales como todos los relacionados al aprendizaje de la programación y también Bases de Datos, Sistemas Operativos, y los relacionados al Análisis, Diseño y Desarrollo de Software. Motivado en la incumbencia profesional, hoy resulta absolutamente imprescindible incluir un curso específico sobre programación orientada a objetos.

Las áreas de Sistemas se han visto sometidas a una carrera ininterrumpida detrás de los avances tecnológicos que ponen su pie en el mercado. Estos avances prácticamente cambian la perspectiva de la disciplina cada vez en períodos más cortos de tiempo. Ello implica que el alumno debe recibir los conceptos básicos de la ciencia desarrollando la capacidad de aprendizaje, adaptación y manejo de las nuevas técnicas en concordancia con esos conceptos básicos.

Resulta evidente que el ritmo de avance tecnológico, por lo tanto la obsolescencia de los conocimientos adquiridos sobre alguno de ellos hacen que cualquier enseñanza específica que el alumno asimile en algunos de los primeros años de la carrera, posiblemente no tendrán vigencia en el momento de su graduación provocando cierta frustración.

Y es por todo esto, igual que en otras áreas, que la mejor forma de asegurar el resultado de la carrera es generar una conciencia de aprendizaje continuo en el alumno que va mas allá de su graduación.

Implementando la Licenciatura en Ciencias de la Computación en base al plan propuesto cabe señalar dos aspectos importantes: primero, se mantiene la concordancia necesaria con el plan de la Licenciatura de la Universidad Nacional del Sur, con lo que seguirá siendo factible tanto la colaboración académica como la migración de aquellos alumnos que desean terminar sus estudios en la otra unidad académica y segundo, se preserva la correspondencia con el plan de la carrera de Analista en Computación como título intermedio de la Licenciatura en Ciencias de la Computación.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Título

Licenciado en Ciencias de la Computación

Nivel

Grado

Perfil del Egresado

El Licenciado en Ciencias de la Computación deberá poseer los suficientes recursos técnicos y metodológicos que lo habiliten para desempeñar eficazmente la profesión y la capacitación científica en el área.

Conocimientos de:

- Matemática para Ciencias de la Computación
- Resolución de Problemas
- Algoritmia
- Paradigmas de programación
- Probabilidad y Estadística
- Lógica para Ciencias de la Computación
- Lenguajes de programación
- Compiladores e Intérpretes
- Inglés técnico
- Organización y Arquitectura de Computadoras
- Sistemas Operativos
- Redes y Teleprocesamiento
- Desarrollo de Software (Análisis, Diseño, Implementación, normas de calidad en la gestión)
- Procesamiento distribuido

Capacidad para:

- Actuar profesionalmente tanto en industrias como en organismos nacionales y privados de todo el país.
- Planificar, dirigir y auditar Proyectos de Desarrollo de Software de cualquier escala.
- Integrar y aplicar los conocimientos científicos del área.
- Diseñar, desarrollar y mantener programas básicos y de aplicación (software).
- Evaluar y poner en funcionamiento el software ya desarrollado.
- Efectuar estudios técnico - computacionales de proyectos que involucren uso de computadoras.
- Promover las aplicaciones de la informática a nuevas áreas.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Incumbencias

El título de Licenciado en Ciencias de la Computación poseerá validez Nacional y habilitará para actuar profesionalmente tanto en industrias como en organismos nacionales y privados de todo el país.

En la actividad profesional tanto independiente como en relación de dependencia, podrá:

- Planificar, dirigir y auditar Proyectos de Desarrollo de Software de cualquier escala.
- Diseñar, desarrollar y mantener programas básicos y de aplicación (software).
- Evaluar y poner en funcionamiento el software ya desarrollado.
- Efectuar estudios técnico - computacionales de proyectos que involucren uso de computadoras.
- Promover las aplicaciones de la informática a nuevas áreas.

Distribución de Materias por Áreas Temáticas

Área Algebra

Elementos de Álgebra
Matemática Discreta

Área Análisis Matemático

Análisis Matemático I
Análisis Matemático II

Área Programación

Resolución de Problemas y Algoritmos
Elementos de Programación
Estructuras de Datos y Algoritmos
Programación Orientada a Objetos
Estructura y Conceptos de Base de Datos*

Área Estadística

Probabilidad y Estadística

Área Fundamentos Teóricos

Fundamentos de Ciencias de la Computación
Lógica para Ciencias de la Computación
Lenguajes de Programación
Inteligencia Artificial
Compiladores e Intérpretes
Algoritmos y Complejidad



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004

Área Idiomas

Inglés Técnico

Área Sistemas

Organización de Computadoras
Arquitectura de Computadoras
Arquitectura de Computadoras
Sistemas Operativos
Redes y Teleprocesamiento
Sistemas Distribuidos
Sistemas de Tiempo Real*

Área Ingeniería de Software

Análisis y Diseño de Sistemas
Teoría y Diseño de Bases de Datos
Desarrollo de Software
Administración y Gestión de Proyectos de Software
Computación Gráfica*

Área Métodos Numéricos

Métodos Computacionales para Optimización Numérica*
Simulación y Modelos*
Métodos Numéricos*
Optimización*

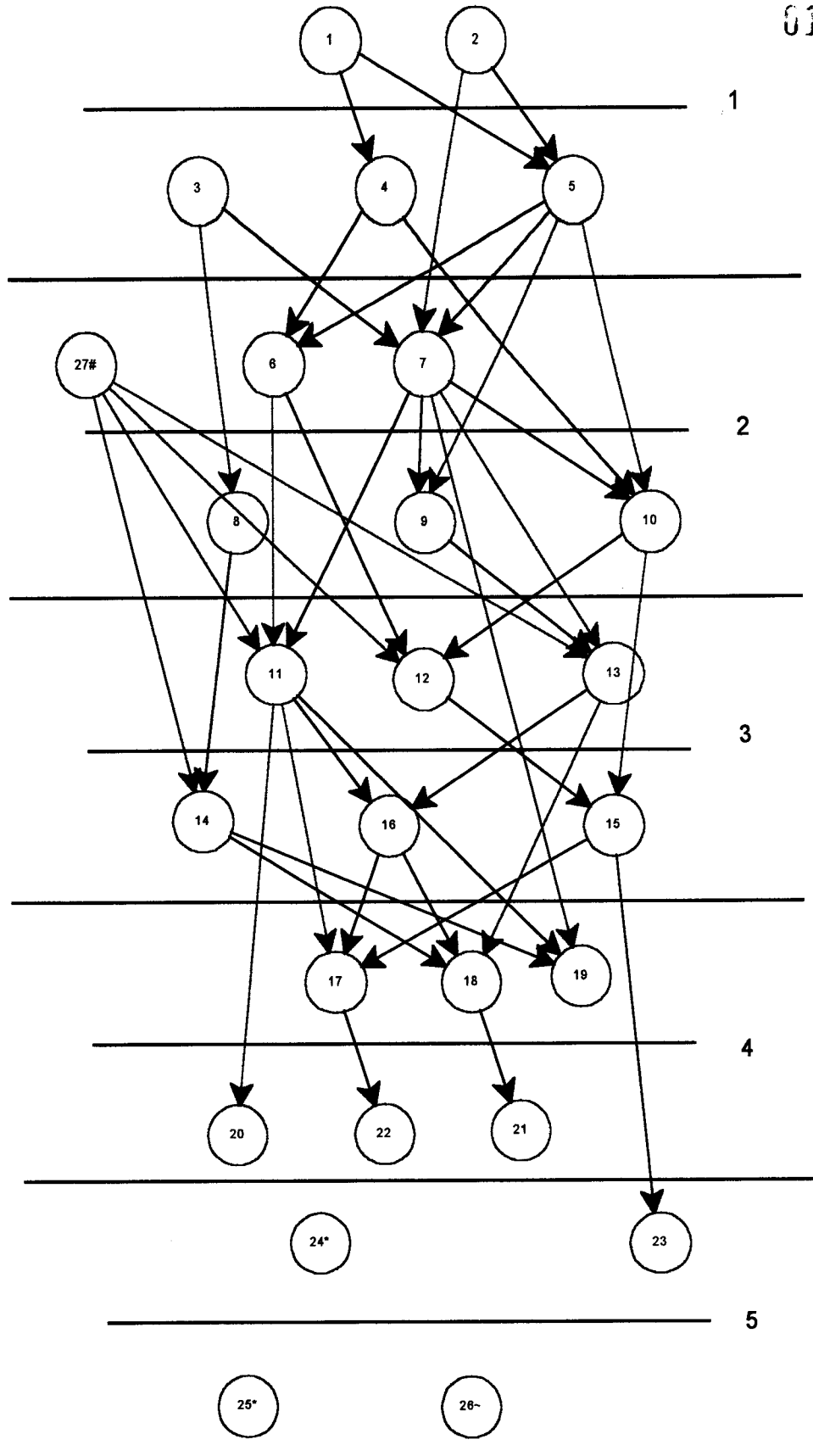
 * Indica Materia Optativa 

Licenciatura en Ciencias de la Computación

Primer Cuatrimestre				Segundo Cuatrimestre					
#	Asignatura	Curs.	Aprob	hs.	#	Asignatura	Curs.	Aprob	hs.
Primer Año									
1	Elementos de Álgebra			12	3	Análisis Matemático I			12
2	Resolución de Problemas y Algoritmos			12	4	Matemática Discreta	1		12
					5	Elementos de Programación	1-2		12
				336 hs.					
Segundo Año									
6	Fundamentos de Ciencias de la Computación	4-5		12	8	Análisis Matemático II		3	12
7	Estructuras de Datos y Algoritmos	3-5	2	12	9	Programación Orientada a Objetos	7	5	12
27	Inglés Técnico			8	10	Organización de Computadoras	7-4	5	12
				448 hs.					
Tercer Año									
11	Lógica para Ciencias de la Computación	7	6-27	12	14	Probabilidad y Estadística	8	27	12
12	Arquitectura de Computadoras	10-6	27	12	15	Sistemas Operativos	12	10	12
13	Análisis y Diseño de Sistemas	9	7-27	12	16	Teoría y Diseño de Bases de Datos	11-13		12
				504 hs.					
Cuarto Año									
17	Lenguajes de Programación	15-16	11	12	20	Inteligencia Artificial		11	12
18	Desarrollo de Software	16-14	13	12	21	Administración y Gestión de Proyectos de Software	18		12
19	Algoritmos y Complejidad	11-14	7	12	22	Compiladores e Interpretes	17		12
				504 hs.					
Quinto Año									
23	Redes y Teleprocesamiento	15		12	25	Optativa*		Según optativa	12
24	Optativa			12	26	Tesis de Licenciatura			12
				336 hs.					

Total de horas aulicas 4480

*La oferta de materias optativas es dinámica y cada año el Departamento de Informática y Estadística elaborará los programas correspondientes.



→ cursada
→ rendida

Inglés
* según optativa
~ Tesis de Licenciatura

[Handwritten signature]

[Handwritten signature]



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004

Metodología de las clases

Todas las materias tendrán dictado teórico – práctico.

Contenidos Mínimos de las Asignaturas

1.- ELEMENTOS DE ÁLGEBRA

Objetivos

Que el alumno adquiera soltura en el manejo de la formalización algebraica del concepto de número y generalizaciones.

Contenidos Mínimos

Conjuntos

Relaciones binarias. Relaciones de orden y equivalencia.

Funciones.

Introducción al Cálculo Proposicional.

Números Reales, Naturales, Enteros, Racionales.

Representación numérica en distintas bases.

Principio de inducción.

Divisibilidad de enteros.

Números Complejos.

Polinomios y ecuaciones algebraicas.

Cálculo combinatorio.

Sistemas de ecuaciones lineales, matrices y determinantes.

2.-RESOLUCIÓN DE PROBLEMAS Y ALGORITMOS

Objetivos

El objetivo fundamental es la resolución de problemas de simple complejidad mediante la computadora. Debido a la gran brecha conceptual existente entre el enunciado del problema a resolver por la computadora y su correspondiente programa este objetivo fundamental se debe cumplir en tres etapas : a) Adquirir habilidad en la detección de una situación de problema y en el planteo de los posibles caminos de solución mediante las técnicas generales de resolución de problemas. b) Resolver los problemas dados en un lenguaje de diseño de algoritmos orientado a un paradigma procedural. Es ESENCIAL formular un algoritmo que represente una solución al problema planteado. c) Transformar el algoritmo de programación escrito en un lenguaje de diseño a un programa escrito en el lenguaje elegido (Pascal). Además se pretende que el alumno adquiera conocimientos de los conceptos y terminología básicos en computación y que resuelva problemas que serán básicos en asignaturas posteriores.



Universidad Nacional del Comahue
Consejo Superior

01004
ORDENANZA N°

Contenidos Mínimos

Problemas, modelos y abstracciones. Representación de problemas. Gráficos, diagramas, modelos matemáticos, descripción verbal. Búsqueda de soluciones de problemas. Inferencia, analogía, similitud entre problemas, generalización y particularización.

Algoritmos. Conceptos. Algoritmos computacionales.

Resolución de Problemas. Métodos. Diseño estructurado y modular. Refinamiento paso a paso.

Programas. Lenguajes de programación. Lenguajes procedurales. Reseña histórica. Estructura de una computadora. Lenguaje PASCAL. Estructura de un programa. Estructuras estáticas.

3.- ANÁLISIS MATEMÁTICO I

Objetivos

Que el alumno adquiera una comprensión de los conceptos del cálculo en una variable y soltura en el cálculo de límites, derivadas, integrales y algunas de sus aplicaciones.

Contenidos Mínimos

Número Real.

Funciones de una variable.

Sucesiones y series.

Límite.

Derivada.

Integral.

Ecuaciones diferenciales.

Curvas.

4.- MATEMÁTICA DISCRETA

Objetivos

Continuando el trabajo formativo comenzado en Elementos de Álgebra se busca que el alumno desarrolle la comprensión de las estructuras algebraicas en general.

Contenidos Mínimos

Multigrafos y Multidigrafos.

Nociones de Álgebra universal.

Reticulados distributivos.

Álgebra de Boole

Estructuras algebraicas. Grupos, anillos, cuerpos, espacios vectoriales.

Espacios vectoriales. Transformaciones lineales, matriz asociada, transformaciones lineales simétricas, autovalores y autovectores.



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004

5.-ELEMENTOS DE PROGRAMACIÓN

Objetivos

El objetivo fundamental es la resolución de problemas de mediana complejidad mediante la computadora. Se introduce además el concepto de recursividad como una nueva metodología de resolución de problemas.

Debido a la creciente complejidad de los problemas a los que deben enfrentarse se plantea luego una extensión a la metodología de diseño, vista ya en resolución de problemas y algoritmos, cuya característica relevante es la división de programa que resuelve el problema planteado en un programa abstracto y un módulo de implementación del tipo de dato abstracto.

Cuando se realiza esta implementación primero se tratan problemas con datos no estructurados y luego, al ir aumentando gradualmente la complejidad de los problemas surge la necesidad de estructuración de los datos y por lo tanto el alumno incorpora gradualmente estructuras de datos fundamentales.

Todos los conceptos se refuerzan con ejemplos de aplicación en tópicos fundamentales de Ciencias de la Computación.

El diseño de un pequeño proyecto durante el cuatrimestre permite consolidar los conceptos vistos.

Contenidos Mínimos

Recursividad. Concepto. Definiciones y planteos recursivos. Iteración vs. recursividad.

Estructura de datos. Concepto y motivación. Identificación y caracterización de los objetos de un problema. **Tipo de Datos.** Concepto. Clasificación. Estructuras estáticas y dinámicas. Representación de datos e implementación de las operaciones.

Ciclo de vida del software.

Programación modular y encapsulamiento de datos.

Tipo de datos abstractos.

Resolución de Problemas. Problemas matemáticos. Problemas numéricos. Procesamiento de datos. Ordenamiento y búsqueda.

6.- FUNDAMENTOS DE CIENCIAS DE LA COMPUTACIÓN

Objetivos

El contenido de este curso corresponde a los elementos clásicos y contemporáneos de fundamentos de ciencias de la computación.

Los tópicos cubiertos son: teoría de autómatas, lenguajes formales, computabilidad de máquinas de Turing, redes de Petri y funciones recursivas.

El objetivo de desarrollar estos temas en la currícula del 2do. año es familiarizar a los alumnos con los problemas de la computación clásica y mostrarles la contemporaneidad de los mismos. Así mismo se busca enseñarles los elementos técnicos mínimos que fundamentan la ciencia de la computación para que asimilen



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004

como los desarrollos técnicos clásicos son aún hoy un peldaño hacia nuevas aplicaciones.

Contenidos Mínimos

Autómatas Finitos. Lenguajes Regulares.

Autómatas a Pila. Lenguajes Libres del Contexto. Lenguajes Sensibles al Contexto.

Computabilidad: Máquinas de Turing. Tesis de Turing-Church.

Redes de Petri.

Funciones Recursivas.

7.- ESTRUCTURA DE DATOS Y ALGORITMOS

Objetivos

Es el tercer curso de la currícula dedicado a temas específicos de Computación. Se desarrollan con detalle los conceptos de estructuración de datos y algoritmos. Como tercer curso de programación se desarrolla siguiendo los principios de modularización, abstracción y encapsulamiento de datos, ocultamiento de la información, diseño descendente de algoritmos y adecuada documentación, iniciados en los dos cursos anteriores.

Las diferentes estructuras, conjuntamente con los algoritmos para su creación, acceso y modificación son presentados para resolver la implementación de tipos de datos abstractos.

Los alumnos desarrollan un proyecto que implementan en Modula II.

Contenidos Mínimos

Conceptos básicos. Tipo de Dato. Estructura de Datos. Tipo de Dato Abstracto (TDA). Estructuras de información dinámica. Implementación.

Programación modular.

Diseño y análisis de algoritmos. Tiempo de ejecución de un programa.

Estructuras lineales. Listas, pilas, colas.

Árboles.

Conjuntos.

Grafos.

Archivos.

8.-ANÁLISIS MATEMÁTICO II

Objetivos

Busca que el alumno continúe el desarrollo del manejo de los elementos del cálculo diferencial integral en varias variables.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Contenidos Mínimos

- Conceptos básicos de geometría analítica.**
- Vectores en el plano y en el espacio.**
- Funciones vectoriales.**
- Funciones reales de varias variables reales.**
- Diferenciación**
- Funciones implícitas.**
- Teorema de Taylor.**
- Integrales dobles.**
- Integrales de línea.**
- Ecuaciones diferenciales ordinarias de primer orden.**

9.-PROGRAMACION ORIENTADA A OBJETOS

Objetivos

El objetivo de la materia es el estudio de las Técnicas y herramientas pertenecientes al paradigma de programación determinado por el modelo orientado a objetos. Se comienza por una introducción a los conceptos generales propios del paradigma: encapsulamiento, abstracción, herencia, polimorfismo y vinculación dinámica. Luego se estudia la aplicación de estos conceptos a lenguajes de programación particulares (Smalltalk, Eiffel, C++, etc.). Por último se analizan las implicaciones que estas nuevas técnicas tienen en el ciclo de vida del software, estudiando en particular el diseño basado en responsabilidades.

Como resultado de ello el alumno deberá ser capaz de programar en los lenguajes estudiados, haciendo uso de las técnicas y herramientas de programación propias del paradigma.

Contenidos Mínimos

- Elementos básicos.**
- Elementos avanzados.**
- Técnicas de diseño.**
- Lenguajes puros e híbridos.**
- Aplicaciones a las Ciencias de la Computación.**

10.- ORGANIZACIÓN DE COMPUTADORAS

Objetivos

Proveer al estudiante de los conceptos básicos sobre la estructura funcional de una computadora y enseñar programación en lenguaje ensamblador, introduciéndolo a través de ello, en la arquitectura. Se cubrirán conceptos y técnicas aplicables en distintas computadoras, y se desarrollará una extensa práctica de programación en lenguaje ensamblador, sobre sistemas PC.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

La materia abarca una serie de temas que son los puntos de partida de las materias Arquitectura de Computadoras y Sistemas Operativos.

Contenidos Mínimos

Estructura de una computadora.

Sistemas de representación de números y caracteres.

La unidad central de proceso.

El lenguaje ensamblador. Ensamblado. Vinculación y carga.

Subrutinas y macros. Pasaje de parámetros. Asignación estática y dinámica de memoria. Corutinas

Entrada - Salida. Programada. Interrupciones Acceso directo a memoria

Periféricos y Sistemas de almacenamiento secundario.

11.- LÓGICA PARA CIENCIAS DE LA COMPUTACIÓN

Objetivos

Esta materia introduce al alumno a las teorías lógico - matemáticas que fundamentan las Ciencias de la Computación. Se presentan rigurosamente la Lógica Proposicional, la Lógica de Predicados y el Cálculo Lambda desde tres puntos de vista: Sintáctico, Semántico y Computacional. Además se realiza una introducción a la verificación formal de programas.

El alumno realizará proyectos de programación, utilizando un lenguaje simbólico (PROLOG, Lisp, etc.), que desarrollen su comprensión de los temas teóricos.

Contenidos Mínimos

Teorías Formales. Definiciones Preliminares. Caracterización de una Teoría Formal. Axiomas. Reglas de Inferencia. Deducción. Teoremas e Interpretaciones. Consistencia. Sanidad. Decibilidad.

El Cálculo Proposicional. El lenguaje del Cálculo Proposicional L. Fórmulas bien formadas. Conectivos. Interpretaciones. Tablas de Verdad. Deducción, axiomas y modus ponens. Teorema de la deducción. Computación en el Cálculo Proposicional. Resolución. Refutaciones. Resultados acerca de completitud. Equivalencia entre las nociones sintácticas, semánticas y computacionales.

El Cálculo de predicados. El lenguaje del Cálculo de Predicados P. La Noción de verdad en P. Interpretaciones y modelos. Satisfabilidad y validez lógica. Consecuencia lógica. Deducción en P. Axiomas y reglas de inferencias. Consistencia de P. El teorema de la deducción en P. Computación en P. Resolución. Substituciones. Forma prenex. Dominio de Herbrand. Unificación. Procedimiento de Robinson. Resolventes. Refutaciones. Equivalencia entre las nociones sintácticas, semánticas y computacionales.

El Cálculo Lambda. El lenguaje. La abstracción funcional. Sintaxis del cálculo. Reglas de conversión. Reducción. Teorema de Church-Rosser. Introducción a la Programación Funcional. Uso de lenguajes funcionales (Lisp, ML, Haskell, etc.)



Universidad Nacional del Comahue
Consejo Superior

61004

ORDENANZA N°

Correctitud de Programas. Construcción correcta a través del método de Aserciones. Inducción como método de prueba de programas. Prueba de programas. Prueba de programas recursivos.

12. ARQUITECTURA DE COMPUTADORAS

Objetivos

Es objetivo de la materia alcanzar una comprensión del diseño y construcción de un sistema de cómputo. Se comienza con un desarrollo de temas básicos, fundamentación en técnicas digitales, para que el alumno entienda en los distintos temas a desarrollar, los cuales en gran medida son compromisos que se asumen según el nivel tecnológico presente.

Estudio en profundidad del procesador dedicado a la ejecución de instrucciones, con sus alternativas de aceleramiento. Se continua con las fases que acomete el procesador en el procesamiento de las instrucciones y la forma de optimizar su funcionamiento. Se desarrolla una clasificación de los sistemas en función del paralelismo presente. A continuación se aborda el tema de memoria, implementación de memoria RAM. Concepto de jerarquía de memorias, Memoria Cache, Memoria Principal, Memoria Secundaria. Manejo de la misma. Memoria Virtual. Por último se desarrollan aspectos generales de comunicación y de Entrada/Salida.

Como resultado de todo ello el alumno deberá alcanzar un entendimiento como para poder analizar y evaluar / comparar distintas arquitecturas.

Contenidos Mínimos

Evolución y Modelos de Computadoras

Metodologías de diseño

El nivel de lógica digital. Sistemas combinacionales y secuenciales

Algebra de Boole. Expresiones canónicas. Métodos de simplificación. Circuitos de integración de media y gran escala. Multiplexores, RAM, ROM, PLA. Circuitos secuenciales por pulsos y por nivel.

Unidad Lógica Aritmética. Representación de números e implementación de las operaciones aritméticas básicas: Suma, Resta, Multiplicación y División en punto fijo. Aritmética en punto flotante. Diseño de la unidad lógica aritmética

Unidad de Control.

Unidad de control cableada y microprogramada.

Segmentación

Concepto de segmentación Riesgos de segmentación

Memoria.

Jerarquía de memoria. Memoria cache Memoria principal. Memoria virtual.

Concepto. Protección y ejemplos de memoria virtual.

Interfases de Procesadores y Periféricos



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Tipos y características de periféricos Buses, Ancho de banda. Transferencia de datos entre dispositivos y memoria. Acceso directo a memoria. Direcciones futuras en los sistemas de E/S. RAID

Arquitectura de Computadoras Avanzadas

13.- ANÁLISIS Y DISEÑO DE SISTEMAS

Objetivos

Estudiar la metodología de Análisis Estructurado de Sistemas, y Análisis Esencial, orientado al desarrollo de Software de tal modo que su aplicación no sea sólo en los sistemas de información sino en cualquier proyecto de desarrollo o mantenimiento de software.

Introducir al alumno en los problemas reales de desarrollo de sistemas, principalmente en la actividad de análisis, realizando la modelización de un sistema real.

Contenidos Mínimos

Sistemas de información. Sistemas. Definición. Conceptos y clasificación de sistemas manuales y automáticos. Software. Componentes. Ingeniería de Software. Principios de Ingeniería de Software.

Procesos de desarrollo de Sistemas. Sistemas de desarrollo de Software. Componentes. Entorno. Modelos, métodos, herramientas. Ciclo de vida para el desarrollo de sistemas. Ciclo de vida en cascada, estructurado, incremental de prototipos. Modelo espiral. Modelo transformacional. Ciclo de vida orientado a objetos.

Ciclos de vida. Requerimientos y especificación. Obtención de requerimientos. Objetivos. Especificación. Objetivos. Enfoques para la especificación de Software. Estilos de especificación. Verificación y validación de especificaciones.

Análisis de Sistemas. Etapa de Análisis. Análisis Estructurado. Métodos estructurados. Herramientas. Diagrama de Flujos de Datos. Diccionario de Datos. Modelo de Entidad Relación. Descripciones estructuradas de procesos. Diagrama de transición de estados. Análisis esencial. Partición por eventos.

Análisis orientado a objetos. Metodologías de análisis orientado a objetos. En esta unidad el alumno deberá resolver el análisis de una aplicación sencilla mediante la metodología de orientación objetos. Deberá finalmente elaborar conclusiones sobre las metodologías estructuradas y de orientación a objetos.

Diseño estructurado. Proceso de transición entre el análisis y el diseño. Diseño estructurado. Módulos. Caja negra. Cohesión. Acoplamiento. Heurísticas de diseño. Estrategias de diseño.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

14.- PROBABILIDAD Y ESTADÍSTICA

Objetivos

Que el alumno desarrolle la comprensión de la intuición detrás de los procesos aleatorios. Además, continuar con la consecuencia lógica que conlleve el análisis estadístico de datos.

Contenidos Mínimos

Cálculo elemental de probabilidades. Variables aleatorias. Momentos de variables aleatorias. Variables aleatorias bidimensionales.

Distribuciones discretas y continuas importantes.

Función generadora de momentos.

Ley de los grandes números.

Aproximación normal a la distribución binomial.

Teorema central del límite.

Muestras aleatorias. Estadísticos.

Estimación de parámetros. El coeficiente de correlación. Intervalos de confianza.

Ensayo de hipótesis.

Teoría de errores de medición.

15.- SISTEMAS OPERATIVOS

Objetivos

El curso no está desarrollado en base a ningún Sistema Operativo en particular. Se discuten conceptos fundamentales que son aplicables para una gran cantidad de sistemas. Se discuten temas como: Servicios de sistemas operativos, sistemas de archivos, planificación de CPU, manejo de memoria y memoria virtual, planificación de disco, interbloqueos, procesos y programación concurrente, protección, sistemas distribuidos, etc. Se presentan como ejemplos de estos conceptos, UNIX y Windows NT mostrando los matices de implementación entre estos sistemas. El curso requiere conocimientos de arquitectura, organización de computadores y estructura de datos.

Contenidos Mínimos

Introducción. Conceptos básicos: Sistemas operativos, multiprogramación, tiempo compartido, sistemas distribuidos, tiempo real, sistemas monousuarios.

Estructura de Sistemas de Cómputo. Sistemas basados en interrupciones. Estructura de las entradas y salidas. Operación en modo dual. Protección por hardware. Arquitectura general del sistema. Diferentes clases de computadoras.

Estructura de Sistemas Operativos. Componentes. Servicios. Llamadas. Programas. Tipos de estructura.

Procesos. Concepto. Concurrencia. Planeamiento. Algoritmos de planeamiento para la CPU. Multiprocesamiento. Evaluación y comparación.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Coordinación de Procesos. Sección crítica. Hardware para sincronización. Semáforos. Problemas de sincronización. Constructores de alto nivel. Comunicación entre procesos.

Interbloqueos. Modelos del sistema. Caracterización. Prevención. Evasión. Detección. Recuperación. Manejo combinado.

Manejo de memoria. Swapping. Alocaación con partición simple y múltiple. Paginado, segmentado y segmentado paginado.

Memoria virtual. Motivación. Demanda de páginas. Rendimiento. Algoritmos de reemplazo y alocaación de cuadros. Thrashing. Demanda de segmentos.

Manejo de almacenamiento secundario. Estructura del disco. Manejo del espacio libre. Métodos de alocaación. Algoritmos de planificación. Rendimiento y confiabilidad. Jerarquías.

Sistemas de Archivos. Organización. Operaciones. Métodos de acceso. Consistencia semántica. Organización en directorios. Protección. Implementación.

Protección. Protección y seguridad. Modelos.

Caso de estudio. UNIX.

Sistemas operativos distribuidos. Objetivos. Hardware y software. Diseño.

16.- TEORÍA Y DISEÑO DE BASES DE DATOS

Objetivos

El contenido de este curso corresponde a los elementos clásicos y contemporáneos de Bases de Datos, tratándose de incorporar las nuevas tecnologías de manejo de datos. Se presentan los distintos modelos de datos existentes, tanto implementados en algún Sistema de Manejo de Base de Datos, SMBD, como modelos netamente teóricos. Se hace fundamental hincapié en el modelo Entidad - Relación, como una herramienta de diseño, y en el modelo relacional, como una implementación práctica eficiente. Se presentan las relaciones entre los SMBD, Arquitectura de Computadoras, Lenguaje de Programación y el Sistema Operativo, a través del estudio de Manejo de Transacciones, Concurrencia, Seguridad y Lenguajes de Consulta.

Contenidos Mínimos

Introducción a los conceptos de Bases de Datos. Arquitectura lógica de un SMBD. Vistas. Diferencias con Sistemas de Archivos. Tipos de Usuarios. Lenguajes de BASES DE DATOS: Definición y manipulación.

Modelo Entidad Relación. Motivaciones. Entidades. Relaciones. Tipos de Relaciones. Atributos. Llaves.

Modelo relacional. Relaciones. Vinculación con el modelo E/R. Ejemplos. Lenguajes teóricos de consulta. Algebra relacional. Cálculo de Tuplas. Cálculo de dominios. Infinitud de relaciones.

Lenguajes de consulta relacionales comerciales. ISBL. QBE. SQL. Ejemplos. Implementación. Relación con los lenguajes de programación tradicionales.



Universidad Nacional del Comahue

Consejo Superior

01004

ORDENANZA N°

Teoría de Diseño de Base de Datos Relacionales. Integridad. Dependencias. Llave. Dependencias funcionales. Axiomas. Cláusula. Cubrimientos. Formas normales. Dependencias multivaluadas. Axiomas. Concepto de Base de dependencias.

Manejo de Transacciones. Modelo de transacción. Modelo de sistemas centralizados. Propiedades: ACID. Atomicidad. Consistencia. Independencia. Durabilidad. La Bitácora. Modificación inmediata y diferida. Recuperación de fallos mediante bitácora.

Manejo de Concurrencia. Planificaciones seriales. Planificaciones serializables. Planificaciones no serializables. Resolución de serializabilidad en conflictos y en vistas. Implementación. El tiempo en las transacciones. Deadlocks. Prevención y detención.

Sistemas distribuidos. Modelo. La red. Las comunicaciones. Recuperación de fallos. Protocolo de compromiso. Selección del coordinador. Manejo de concurrencias. Locks. Prevención y detección de Deadlocks. Estampillas de tiempo. Asignación de un reloj común. Sistemas distribuidos heterogéneos.

Otros modelos de datos. Modelo de red. Lenguaje de consulta. Modelo jerárquico. Comparación entre los modelos de datos. Modelo orientado a objetos. Identificación de objetos. Diseño orientado a objetos. Métodos de acceso como consultas.

Organización física de los datos. Organización heap. Organización hash. Índices. Índices densos y ralos. B-árboles. Registros de longitud variable. Estructuras para campos no claves.

17.- LENGUAJES DE PROGRAMACION

Objetivos

Existen varias maneras de enfocar y organizar esta materia. La mayor parte de la bibliografía está estructurada de acuerdo a una de las dos siguientes alternativas:

* **Enfoque Vertical:** Cada uno de los tópicos inherente al lenguaje de programación (LP) es tratado individualmente y se ilustra tomando como ejemplo LP de diferentes características. Este enfoque permite efectuar un buen análisis comparativo de los diferentes conceptos planteados.

* **Enfoque Horizontal:** Se describe en forma más o menos detallada características de una serie de LP. Cada lenguaje es descrito en forma individual. En este enfoque es posible resaltar claramente las motivaciones que dieron lugar a un LP particular o a una clase de LP.

Se cubren ambos enfoques paralelamente. Los capítulos I - IX se presentaran paralelamente igual los X, XI y XII.

La intención es que al terminar la materia ambos enfoques se hayan complementado en un intento de alcanzar las ventajas de cada uno de ellos. Cada concepto va a ser presentado en dos contextos diferentes.

Conocer las modificaciones de diseño de un LP permite hacer un uso más inteligente del LP. Conocer las de varios LP permite selección, el más adecuado para cada aplicación en particular y estudiar un nuevo lenguaje con la capacidad de poder



Universidad Nacional del Comahue

Consejo Superior

01004

ORDENANZA N°

realizar un análisis comparativo - evaluar y comparar -. Conocer las características de implementación de un LP permite hacer un uso más eficiente del mismo.

El objetivo del enfoque Horizontal no es de ninguna manera presentar uno o varios lenguajes en forma exhaustiva. Los LP que se tratan han sido elegidos porque:

- 1.- Ilustran características importantes de los LP tradicionales.
- 2.- Constituyen el punto de partida de muchas ideas.

Contenidos Mínimos

Introducción. Descripción de la estructura general de la materia- motivaciones. Enfoque vertical. Enfoque horizontal.

Etapas en el proceso de desarrollo de software. El proceso de desarrollo de software. Metodologías de diseño de software. Arquitecturas de computadoras.

Evolución de los lenguajes de programación. El rol de la abstracción en la evolución de los lenguajes de programación. Paradigmas de programación.

Semántica operacional. Sintaxis y semántica. Procesamiento de lenguajes de programación: intérpretes y compiladores. El concepto de la ligadura - binding. Entidades y atributos. Clasificación de los lenguajes en función del esquema de alocaación de memoria y de las reglas de alcance. Implementación.

Estructuras de control. Estructuras de control a nivel expresión. Estructuras de control a nivel sentencia. Estructura de control a nivel unidad. Tipos de parámetros y tipos de pasaje.

Datos y tipo de datos. Clasificaciones. Datos simples y estructurados. Dominio y operaciones. Definición e implementación. Tipos redefinidos y definidos por el programador. Equivalencia entre equipos. Abstracción de datos. Generisidad. Polimorfismo.

Especificaciones formales de los lenguajes de programación. Sintaxis y semántica. Especificaciones formales. Metalenguajes. Semántica formal. Semántica axiomática. Semántica denotacional. Semántica translacional. Semántica operacional.

Motivaciones y evaluación de lenguajes de programación. Diseño e implementación. El énfasis en la eficiencia: Fortran. Elegancia y generalidad: Algol 60. Simplicidad: Pascal. Modularidad: Módula-2. Abstracción de datos y concurrencia.: Ada. Programación orientada a objetos: Smalltalk Lenguaje de Programación Declarativos. Lenguajes de cuarta generación. Programación Funcional: LISP. Programación en Lógica: PROLOG.

Principios del Diseño de lenguajes de programación. Abstracción. Encapsulamiento de datos. Ortogonalidad. Portabilidad. Regularidad. Seguridad. Simplicidad. Poder Expresivo. Consistencia sintáctica y semántica. Programación a gran escala.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

18.- DESARROLLO DE SOFTWARE

Objetivos

El contenido de este curso corresponde a lo que se conoce como " Ingeniería de Software". Su objetivo básico es el estudio de los métodos, técnicas y herramientas para diseño, desarrollo y mantenimiento de Software, de tal forma de obtener un producto de Software confiable y de bajo costo de mantenimiento.

Contenidos Mínimos

Software y desarrollo de software. Ingeniería del Software. Crisis y mitos.

Fundamentos del diseño del software. Diseño y calidad. Procesos y fundamentos del diseño. Diseño modular. Diseño de datos. Diseño arquitectónico y procedural. Ocultamiento de la información.

Especificaciones formales. Desarrollo de una especificación formal. Especificaciones algebraicas. Especificaciones basadas en las modelos. Método de desarrollo (RAISE).

Diseño orientado al flujo de datos. Diseño y flujo de información. Proceso del diseño. Análisis de transformación. Análisis de transacción. Heurísticas de diseño. Cohesión y acoplamiento. Optimización del diseño.

Diseño orientado a objetos. Orígenes. Conceptos básicos. Ventajas. Métodos de Wirf-Brooks y por los datos.

Prueba del software. Fundamentos. Objetivos. Prueba de caja blanca. Diseño de casos. Prueba de camino básico. Prueba de caja negra. Prueba de corrección. Herramientas automáticas.

Estrategias de prueba del software. Enfoque estratégico. Verificación y validación. Organización de la prueba. Prueba de unidad. Prueba de integración. Prueba de validación. Prueba del sistema. Prueba ascendentes y descendentes. Pruebas alfa y beta. Arte de la depuración.

Arquitectura de software.

19.- ALGORITMOS Y COMPLEJIDAD

Objetivos

Se introduce al alumno en las técnicas de análisis de algoritmos, se ilustra con diversas estrategias de solución de problemas y se estudia la noción de clases de complejidad.

Contenidos Mínimos

Orden de un algoritmo. Notación. $O(f(n))$. Otras notaciones. Operaciones con notaciones asintóticas. Notación asintóticas condicional. Análisis de recurrencias.

Algoritmos Greedy. Ejemplo. Corrección. Análisis de eficiencia.

Algoritmos dividir y conquistar. Ejemplos: Búsqueda y ordenamiento. Análisis de eficiencia.



Universidad Nacional del Comahue

Consejo Superior

01004

ORDENANZA N°

Programación dinámica. Ejemplos: problemas de optimización. Análisis de eficiencia.

Algoritmos y estructuras de datos. Grafos. Representaciones de los grafos. Algoritmos de búsqueda. Archivos. Representación de archivos. Algoritmos de ordenamiento externo.

Introducción a la complejidad computacional. Problemas. Problemas de decisión. Ejemplos. Modelos de computación. Clases de problemas. Herramientas para la comparación de clases de problemas. Reducción de problemas. Medidas de complejidad. Revisión concepto de Máquinas de Turing.

Clases de complejidad. Objetivos. Clases P y NP: definición, características, ejemplos, comparación. Problemas NP-Completo. Teorema de Cook. Ejemplos de reducciones a problemas NP-Completo. Algoritmos de aproximación para problemas NP-Completo. Complejidad en el espacio.

20.- INTELIGENCIA ARTIFICIAL

Objetivos

El área de Inteligencia Artificial está formada por una multiplicidad de subáreas. Este curso provee los fundamentos para el posterior desarrollo de esos temas apoyándose en los desarrollos recientes de los formalismos lógicos.

Se presentan los métodos de búsqueda, se discuten diferentes formas de representación de conocimiento y razonamiento, y se estudian algunas aplicaciones importantes tales como las ATMS, la representación de acciones y los sistemas de planeamiento.

El alumno realizará proyectos de programación, utilizando un lenguaje simbólico (PROLOG, Lisp, etc.), que desarrollen su comprensión de los problemas involucrados en la generación de comportamiento inteligente.

Contenidos Mínimos

Introducción y motivaciones. Definiciones. Test de Turing. Problemas filosóficos. Historia.

Búsqueda. Búsqueda ciega. Breath-first, depth-first, profundización iterativa y ampliación iterativa. Procesos de búsqueda heurística. Heurística. Funciones de evaluación. Búsqueda optimal. Algoritmos A*. Admisibilidad y optimalidad de A*. Extensiones e IDA*. Búsqueda adversaria. Minimax. Búsqueda α - β .

Conocimiento declarativo. Conceptualización. Cálculo de predicados. Semántica. Ejemplos. Lenguajes especializados.

Inferencia. Derivabilidad. Procedimientos de inferencia. Implicación lógica. Demostrabilidad. Probando demostrabilidad.

Resolución. Diferentes estrategias. Forma clausal. Unificación. El principio de resolución. Insatisfabilidad. Ejemplos. Sanidad y completitud. Igualdad y resolución. Estrategias de borrado. Resolución unidad, de entrada, lineal, conjunto de soporte, ordenada, dirigida.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Sistemas basados en reglas. Sistemas de producción. Introducción. Definiciones. Ejemplos. Ventajas y problemas asociados. Técnicas y estrategias para la resolución de conflictos. Aplicabilidad.

Razonamiento no monotómico. Introducción y motivaciones. Teorías formales. No monotonicidad procedural en inteligencia artificial. Algunas formalizaciones relevantes: lógica default, Lógica no monotónica, Circunscripción, Razonamiento rebatible.

Conocimiento y creencias. Conceptos preliminares. Lógicas sentenciales de creencias. Métodos de prueba. Propiedades del conocimiento y de las creencias. Semántica de los mundos posibles.

Razonamiento probabilístico. Probabilidad de sentencias. Regla de Bayes. Lógica probabilística. Inferencia probabilística.

Estados y cambio. Planeamiento. Estado. Acciones. El problema del macro. Ordenamiento de acciones. Condicionalidad.

21.- ADMINISTRACIÓN Y GESTIÓN DE PROYECTOS DE SOFTWARE.

Objetivos

Que el alumno pueda integrar las herramientas de desarrollo de software adquiridas con las herramientas de control y gestión de calidad con el empleo de las normas más modernas.

Contenidos Mínimos

El contexto de la administración y control de proyectos. Desarrollo de proyectos de software. Qué es un proyecto. Preparación y evaluación de proyectos.

Introducción a la ingeniería de sistemas. Características de la ingeniería de sistemas. La emergencia de la ingeniería de sistemas. Objetivos de la ingeniería de sistemas. Manejo de sistemas. Manejo de la organización. Estructuras organizacionales. Ciclos de vida del manejo de sistemas. Ingeniería del software. Manejo del desarrollo de sistemas de software.

Factores humanos en la ingeniería del software. Dinámica de grupos.

El planeamiento de proyectos. Actividades del planeamiento. Distribución de tiempos.

Caos y orden el proceso de desarrollo de software. El control. El dilema de la estimación. Proyección de los costos del sistema. El costo de medir.

Modelos y métricas de sistemas. Construcción y usos de modelos. Métricas (de la especificación, el diseño, la implementación y el resultado).

Modelos de costos. Proyección de costos. Modelos de costos sensibles al tiempo. Esquema práctico para trabajar con costos.

Calidad del software. Definición de calidad del software. Búsqueda de la calidad del software. Control y mejoramiento de la calidad. Normas ISO 9000.

Auditoría de Software.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Manejo de la introducción de herramientas automáticas. Detección de la necesidad. Selección y evaluación de productos candidatos. Presentación del producto a superiores y usuarios. Impacto de la automatización del desarrollo de proyectos de software.

22.- COMPILADORES E INTÉRPRETES

Objetivos

Es un curso de diseño de compiladores e intérpretes. Se ofrecen distintas formas de resolver los problemas que se presentan al diseñar el traductor de un lenguaje, independientemente de las máquinas fuente y destino.

A lo largo del curso los alumnos desarrollan un trabajo de aplicación que consiste en un compilador para un subconjunto del lenguaje Pascal (incluye procedimientos, parámetros, recursividad y tipos Integer y Boolean).

La traducción se hace a código intermedio, y se complementa el trabajo con la interpretación de la representación intermedia.

Contenidos Mínimos

Conceptos básicos. Traducción. Compiladores, ensambladores e intérpretes. Lenguaje fuente, objeto y de implementación.

Estructura de un compilador. Especificación de lenguajes. Sintaxis y semántica. Gramática. Fases de un compilador. Pasadas.

Un compilador sencillo de una pasada. Presentación de técnicas básicas de compilación.

Análisis léxico. Construcción de un analizador léxico. Uso de autómatas finitos. Generadores de analizadores léxicos. El analizador LEX. Aplicaciones.

Análisis Sintáctico. Función del analizador sintáctico. El problema de la ambigüedad. Construcción del analizador sintáctico. Métodos ascendentes y descendentes. Manejo de errores. Generadores de analizadores sintácticos. El generador YACC. Aplicaciones.

Traducción dirigida por la sintaxis. Definiciones dirigidas por la sintaxis y esquemas de traducción. Atributos. Definiciones de atributos por izquierda. Evaluación de atributos.

Comprobación de tipos. Sistemas de tipos. Especificaciones de un comprobador de tipos. Conversiones. Sobrecarga.

Ambientes para el momento de ejecución. Organización de la memoria. Asignación de memoria. Tablas de símbolos.

Generación de código intermedio. Lenguajes intermedios.

Generación de código. La máquina objeto. Administración de memoria en ejecución.

Un generador de código simple.

Optimización de código. Fuentes principales de optimización. Algunas soluciones.

Validación de un compilador. Conceptos. Test y validación.



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

23.- REDES Y TELEPROCESAMIENTO

Objetivos

El objetivo principal es que el alumno adquiera los conceptos básicos del funcionamiento de una red de transmisión de datos considerando normas, protocolos y técnicas.

Contenidos Mínimos

Redes de computadoras. Definición de una red de computadoras. Aplicaciones. Arquitectura de una red de computadora. Organismos de estandarización. Clasificación: Redes locales, Redes metropolitanas y Redes de larga distancia (WAN). Redes digitales de servicios integrados (ISDN). Interconexión de redes.

Transmisión de datos. Transmisión de datos paralelos. Transmisión de datos series: sincrónica y asincrónica. Análisis de Fourier. Ancho de banda de un canal. Modulación. Técnicas de modulación digital: ASK, FSK, PSK QAM y PCM.

Nivel de enlaces de datos.: Protocolos elementales. Protocolos de ventana deslizante. Protocolos orientados al bit y al carácter ejemplos (BSC, HDLC). Análisis de protocolos.

Nivel de redes I. Redes punto a punto. Circuitos virtuales y datagramas, comparación del servicio. Algoritmos de ruteamiento. Congestión, preadjudicación de buffers, desecho de paquetes, control de flujo. El nivel de red en X.25. Estudio de la red pública ARPAC.

Nivel de redes II. Comunicaciones por satélite y radiotransmisoras. Métodos de adjudicación del canal. Aloha puro y discreto. Retardos y eficiencia de transmisión, estabilidad. Aloha controlado. Esquemas de reservación.

Nivel de redes III. Redes locales. Características principales. Topologías. Normas del IEEE. Redes de topología barra: redes con detección de portadora, métodos persistente, no-persistente, p-persistente. Ethernet. Protocolos sin colisión. Barra con ficha pasante. Redes de topología anillo: anillos con ficha pasante, de contención, discretos, con registro de inserción.

Software de comunicaciones. Nivel de transporte, de sesión, de presentación y de aplicación. Servicios asociados a cada nivel. Drivers. Control de mensajes y buffers. Gateways. Seguridad y privacidad, criptografía, encodificación. Sistemas distribuidos.



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004

Materias Optativas

MÉTODOS COMPUTACIONALES PARA OPTIMIZACIÓN NUMÉRICA

Objetivos

Que en una primera etapa el alumno aprenda la formulación de algunos modelos de optimización discreta ; y en una segunda , adquiera las ideas básicas de los algoritmos que son base de los programas y paquetes de soft específicos, con un mayor énfasis matemático en los modelos lineales.

En una tercera etapa, que el estudiante sea capaz de diseñar e implementar técnicas para resolver problemas clásicos y modelizar aplicaciones innovadoras ; y que adquiera un adiestramiento en el uso de los paquetes y bibliotecas disponibles para el aprendizaje en forma interactiva, la ejecución automática para la validación de los modelos ,el cálculo y el análisis de la soluciones aportadas por los métodos.

Contenidos Mínimos

Elementos del Álgebra Lineal Numérica
Modelos de Programación Lineal y P.L. Entera .Aplicaciones
Algoritmos de Programación No Lineal.
Modelos de Redes de Optimización y de Actividades.
Elementos de Programación Dinámica Determinística.
Técnicas Computacionales
Técnicas para la resolución de problemas tipo de las unidades anteriores; y uso de paquetes de programas y bibliotecas.

SISTEMAS DISTRIBUÍDOS

Objetivos

El objetivo del curso es proveer una introducción a los conceptos y principios de diseño usados en la construcción de sistemas distribuidos basados en las últimas publicaciones y estado del arte.

Contenidos Mínimos

Introducción a los Sistemas Distribuidos
Comunicación en Sistemas Distribuidos
Sincronización de Sistemas Distribuidos
Procesos y procesadores en Sistemas Distribuidos
Sistemas de archivo en Sistemas Distribuidos
Control de concurrencia
Transacciones distribuidas
Recuperación y tolerancia a fallas
Seguridad



Universidad Nacional del Comahue
Consejo Superior

ORDENANZA N° 01004

**Memoria compartida distribuida
Casos de estudio**

COMPUTACION GRÁFICA

Objetivos

El objetivo de esta asignatura es introducir al alumno en temas básicos de Computación Gráfica, dándole una iniciación en esta rama de la computación.

El desarrollo de esta rama de la Computación durante la década pasada ha transformado la interacción hombre - máquina de modo que el futuro Lic. en Ciencias de la Computación debe estar capacitado para hacer uso de la tecnología actual como usuario y diseñador.

Por ello que esta materia pretende capacitar al alumno en los diversos aspectos de la presentación gráfica de la información.

Contenidos Mínimos

Conceptos básicos de computación gráfica
Algoritmos gráficos de rasterización para dibujar primitivas en 2D
Graficación en dos dimensiones
Modelamiento y aproximación de objetos con curvas y superficies
Línea y cara ocultas
Realismo fotográfico
Teoría del color
Técnicas de modelamiento avanzado
Fractales
Sistemas de partículas
Sistemas basados en gramáticas

SISTEMAS DE TIEMPO REAL

Objetivos

Los alumnos ya han adquirido suficientes conocimientos técnicos en los cursos anteriores, luego se pretende integrar estos conocimientos y su aplicación.

Que el alumno adquiera comprensión de los problemas más significativos en el desarrollo de sistemas dedicados de tiempo real.

Que el alumno desarrolle la habilidad de producir pequeños sistemas que impliquen manejo de interrupciones, entrada salida de bajo nivel, concurrencia, manejo de tiempos críticos.

Contenidos Mínimos

Conceptos básicos
Hardware para tiempo real
Características de los lenguajes de tiempo real



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

Herramientas de Ingeniería de Software para tiempo real
Técnicas de especificación y diseño para tiempo real
Kernels para tiempo real
Comunicación y sincronización de tareas
Administración de memoria en tiempo real
Análisis de rendimiento y optimización
Sistemas de multiprocesamiento
Integración hardware - software

SIMULACIÓN Y MODELOS

Objetivos

Un modelo es la representación (en nuestro caso en software) de un sistema. La simulación es la utilización de modelos para el estudio de sistemas.

Los objetivos de este curso son introducir los principios básicos de la simulación, en sus varias formas, con especial énfasis en simulación de eventos discretos, y la construcción de modelos desde el punto de vista del software. También hacer un estudio elemental del análisis estadístico de resultados de la simulación.

Contenidos Mínimos

Elementos de simulación en computadores.
Generadores de números aleatorios.
Introducción al Lenguaje de Simulación GPSS-PC.
Modelización de los procesos de entrada.
Análisis estadísticos de los resultados.

ESTRUCTURA Y CONCEPTOS DE BASE DE DATOS

Objetivos

Que el alumno comprenda y pueda comparar y evaluar las estructuras físicas de la organización de bases de datos en medios magnéticos.

Contenidos Mínimos

Organización de bases de datos en medio magnético.
Seek time – Latencia Rotacional – Transferencia de bloques
Cálculo de tiempos reales de acceso
Manejo de buffers
Sorting externo
Arboles B+
Hashing
Multilistas
Archivos Especiales: B-D – Grid – hB – WOBT
Sistemas de directorio



Universidad Nacional del Comahue
Consejo Superior

01004

ORDENANZA N°

MÉTODOS NUMÉRICOS

Objetivo

Brindar al alumno una introducción a la resolución numérica de problemas matemáticos.

Contenidos Mínimos

El error en el análisis numérico
Sistemas de representación numérica
Solución numérica de ecuaciones no lineales
Solución numérica de sistemas lineales
Métodos directos y métodos iterativos
Cálculo numérico de autovalores y autovectores
Aproximación de funciones
Interpolación
Diferenciación e integración numérica
Solución numérica de sistemas algebraicos no lineales

OPTIMIZACIÓN

Objetivo

Brindar al alumno el conocimiento de problemas clásicos de Investigación Operativa, su implementación y su justificación

Contenidos Mínimos

Programación lineal
Método Simplex
Dualidad – Definición de teoremas fundamentales
Problema del transporte
Redes de optimización
Programación dinámica
Filas de espera