

Pcia. Roque Sáenz Peña, 25 de julio de 2011

RESOLUCIÓN N° 175/11 – R.

VISTO:

Las actuaciones iniciadas por la Mg. Ing. Patricia Zachman, medio por el cual eleva el Programa Analítico de la Asignatura Algoritmos y Estructuras de Datos, correspondiente a la carrera Ingeniería en Sistemas de Información de la Universidad Nacional del Chaco Austral, para su aprobación; y

CONSIDERANDO:

Que el mencionado Programa se ajusta a los contenidos mínimos y carga horaria de la citada carrera.

Que se consideran adecuados los objetivos, métodos pedagógicos, métodos de evaluación, programa analítico y bibliografía que forman parte de la propuesta.

POR ELLO:

EL RECTOR ORGANIZADOR

DE LA UNIVERSIDAD NACIONAL DEL CHACO AUSTRAL

RESUELVE

ARTICULO 1°. Aprobar el Programa Analítico de la Asignatura **Algoritmos y Estructuras de Datos**, que tendrá vigencia a partir del ciclo lectivo 2011 y que corresponde a la carrera **Ingeniería en Sistemas de Información**, de la Universidad Nacional del Chaco Austral, y que como Anexo Único forma parte de la presente Resolución.

ARTÍCULO 2°. Regístrese, comuníquese a la Mg. Ing. Patricia Zachman y a las Áreas Correspondientes. Cumplido, archívese.





ALGORITMOS Y ESTRUCTURAS DE DATOS

Resolución N° 175/11 – R.
ANEXO

Carga Horaria: 150 hs		Programa vigente desde: 2011	
Carrera		Año	Cuatrimestre
INGENIERÍA EN SISTEMAS DE INFORMACIÓN		Primero	Segundo
CORRELATIVA PRECEDENTE (*)		CORRELATIVA SUBSIGUIENTE (*)	
Asignaturas		Asignaturas	
Para cursar		Para rendir	Análisis de Sistemas Sistemas de Representación Paradigmas de Programación Diseño de Sistemas Sintaxis y Semántica de los Lenguajes
Regularizada	Aprobada	Aprobada	
-----	-----	-----	
DOCENTES:		Profesor Adjunto: Mg. Ing. Patricia ZACHMAN Jefe de Trabajos Prácticos: Prof. Magdalena BROLLO Ing. Fabián LEGUIZAMÓN	
OBJETIVOS:		<p>Objetivos Generales</p> <p>Que el alumno conozca:</p> <ul style="list-style-type: none"> • Las principales estructuras de datos y la aplicación adecuada según la problemática a resolver. • La administración eficiente y eficaz del espacio y tiempo computacionales. • Los principales algoritmos a utilizar según el problema a resolver. • Los conceptos relacionados con el análisis de algoritmos. • Listas enlazadas y Árboles pudiendo desarrollar, programar y optimizar soluciones a problemas de aplicación, usando apuntadores y nodos. • Algoritmos de alto nivel utilizando recursividad, grafos y estructuras de memoria para la resolución de problemas. • Las relaciones que existen entre los datos involucrados en problemas, las estructuras de los medios de almacenamiento y las máquinas. • Métodos útiles para representar los datos en almacenamiento y las técnicas para manipular las estructuras de datos. <p>Que el alumno aprenda:</p> <ul style="list-style-type: none"> • Conceptos y teorías vinculados el uso de las estructuras de datos y su representación en memoria. • Prácticas tecnológicas en QBasic y Pascal como lenguajes procedurales. • Búsqueda, evaluación y selección de información desde las redes de Internet. • Derivaciones metodológicas, tecnológicas y técnicas de las aplicaciones informáticas. • A comprender perspectivas y valoraciones en esta área. <p>Que el alumno desarrolle:</p> <ul style="list-style-type: none"> • Una estima duradera por el aprendizaje. • Un sentido de responsabilidad por el propio comportamiento. • El respeto por el otro. • Un compromiso por la honestidad. • Herramientas metodológicas importantes en esta disciplina. • Eficazmente su trabajo. • El uso de racional y efectivo de herramientas tecnológicas 	

<p>OBJETIVOS</p>	<p>Objetivos Específicos. Que el alumno logre:</p> <p>UNIDAD 1: Estructuras de Datos</p> <ul style="list-style-type: none">• Reconocer las diferentes fases en la solución de problemas de computadora.• Conceptualizar algoritmos y buscar analogías reales.• Conocer los atributos de los algoritmos.• Diseñar buenos algoritmos, que cumplan los objetivos y las propiedades.• Comprender el significado de la Heurística y sus beneficios al aplicarla en la resolución de problemas.• Identificar las estrategias heurísticas y aplicarlas a problemas reales.• Conocer y clasificar técnicas de diseño de algoritmos. <p>UNIDAD 2: Estructuras Primitivas</p> <ul style="list-style-type: none">• Clasificar las estructuras de datos, valorar sus utilización e identificar sus diferentes formas de representación.• Diseñar soluciones informáticas con el uso de estructuras primitivas.• Identificar las operaciones sobre estructuras primitivas y sus alcances en programación.• Indagar sobre las formas de representación en memoria.• Comparar algoritmos de búsqueda.• Comparar algoritmos de ordenamiento de elementos. <p>UNIDAD 3: Estructuras Complejas: Pilas Y Colas</p> <ul style="list-style-type: none">• Identificar las propiedades del uso de pilas como estructuras de datos.• Diferenciar su implementación en memoria con otras estructuras.• Identificar los problemas comunes que se resuelven mediante pilas.• Describir las formas gráficas de representación y el manejo de datos.• Analizar la estructura LIFO.• Indagar sobre la coexistencia de pila de en memoria y sus consideraciones respecto a la reserva de espacio.• Comparar la estructura de las pilas con las colas.• Distinguir sus elementos y propiedades respecto de las pilas.• Analizar la estructura de entrada salida de elementos FIFO.• Identificar los problemas generados al almacenar pilas y colas en arreglos.• Observar las características puntuales de las colas circulares.• Indagar sobre cómo se mide el comportamiento de una cola.• Comprender los alcances de la Teoría de Colas. <p>UNIDAD 4: Estructuras Complejas: Listas</p> <ul style="list-style-type: none">• Conceptualizar listas y los elementos de esta estructura.• Identificar ventajas y desventajas de este almacenamiento.• Diferenciar representación física de representación lógica.• Conocer y aplicar las operaciones básicas que se realizan sobre listas encadenadas.• Identificar la función de espacio disponible.• Identificar las propiedades en el uso de punteros.• Diferenciar listas simple encadenadas de listas circulares.• Diferenciar listas simple encadenadas de listas doble encadenadas.
-------------------------	---

<p>OBJETIVOS</p>	<p>UNIDAD 5: Estructuras Complejas: Arboles y Grafos</p> <ul style="list-style-type: none"> • Definir arboles y grafos desde su representación no lineal. • Distinguir los componentes y propiedades de los grafos respecto de los arboles. • Aprender las formas de recorrido en los grafos en amplitud y en profundidad. • Resolver caminos críticos. • Reconocer tipos de arboles. • Recorrer árboles en orden y post orden. • Identificar sus formas de representación, características y condiciones particulares.
<p>CONTENIDOS MINIMOS</p>	<p>Concepto de Dato. Tipos de Datos Simples. Tipo Abstracto de datos. Estructuras de Control Básicas: secuencial, condicional, cíclica. Estrategias de Resolución. Estructuras de Datos: registros, arreglos y archivos. Abstracciones con procedimientos y funciones. Pasaje de Parámetros. Estructuras de Datos lineales (Pilas-Colas). Algoritmos de Búsqueda, Recorrido y Ordenamiento. Archivos de Acceso Secuencial y Aleatorio: organizaciones y accesos. Procesamiento Básico. Recursividad. Nociones de Complejidad Computacional. Noción de Orden de Complejidad.</p>
<p>MÉTODOS PEDAGÓGICOS:</p>	<p>Aspectos pedagógicos y didácticos</p> <p>Se propone:</p> <ul style="list-style-type: none"> • El desarrollo del pensamiento crítico, reflexivo, práctico y autónomo sobre la práctica educativa. • El desarrollo de habilidades y competencias propias de la programación para su aplicación en problemas computacionales reales. <p>En esta propuesta, el aula se entiende como un espacio de taller para la construcción, en el que se trabaja interactuando permanentemente. La retroalimentación se concreta con una estructura bidireccional, donde tanto los alumnos como el docente se consideran fuente de información. En base a ello se han seleccionado las siguientes técnicas metodológicas:</p> <ul style="list-style-type: none"> _ Discusión dirigida. _ Trabajo en grupo. _ Exposiciones abiertas orales. _ Estudio dirigido mediante itinerarios didácticos. _ Investigación Bibliográfica. _ Trabajo individual y grupal de computadora. <p>La discusión dirigida, las exposiciones abiertas y el estudio dirigido se emplearán por lo general para presentar temas nuevos, mientras que el trabajo en grupo, la investigación bibliográfica y en Internet, se emplearán para su aplicación en ejercitaciones reales y profundizar en un tema ya presentado. En las clases prácticas la técnica metodológica por excelencia será el trabajo taller grupal que permite promover la construcción compartida del conocimiento y lograr así no sólo la apropiación activa del mismo por parte de los miembros del grupo, sino también la indispensable socialización del estudiante, ya que toda su vida deberá transcurrir en contacto y en cooperación con sus semejantes.</p>

MÉTODOS PEDAGÓGICOS

Actividades de los Alumnos y de los Docentes

- *Actividades de los docentes*

La asignatura está a cargo de un equipo docente conformado por un Profesor Adjunto, y dos docentes responsables de la práctica. El rol que desempeñan los docentes es de:

- Facilitador del aprendizaje,
- Observador del proceso grupal,
- Propiciador de la comunicación,
- Asesor grupal,
- Proporcionador de las técnicas de búsqueda de información.

Las funciones específicas de cada docente son:

- Profesor Adjunto:

- Desarrollar las clases teóricas.
- Atender consultas de los alumnos.
- Coordinar el equipo cátedra
- Seleccionar el material bibliográfico.
- Preparar material didáctico.
- Evaluar permanentemente.
- Supervisar el desarrollo de las clases prácticas.
- Supervisar la preparación de los trabajos prácticos.
- Coordinar el desarrollo del taller.

- Jefe de Trabajos Prácticos:

- Desarrollar las clases prácticas.
- Preparar los trabajos prácticos.
- Participar en el desarrollo de los talleres.
- Atender consultas de los alumnos.
- Colaborar en la preparación de material didáctico.
- Colaborar y participar en el proceso de evaluación.

Hay actividades que se llevan a cabo en forma conjunta, como la planificación de la asignatura, la preparación del plan de evaluación y el análisis de las diferentes evaluaciones efectuadas con el fin de mejorar la calidad de la enseñanza.

- *Actividades de los alumnos*

Las actividades a desarrollar por los alumnos en las clases son:

- Participar de las discusiones sobre los temas que se traten en cada clase.
- Preparar y exponer los prácticos que los docentes se lo requieran.
- Resolver los trabajos prácticos.
- Realizar investigación bibliográfica solicitada por los docentes.
- Formular problemas.
- Sintetizar.
- Estudiar grupal e independientemente.

Recursos Didácticos

Se utilizan como recursos didácticos:

- Bibliografía actualizada (libros, revistas y publicaciones científicas). Estos se utilizarán como una manera de acercar a los alumnos a los avances producidos dentro de la disciplina; como una forma que el alumno adquiera habilidad para sintetizar e integrar informaciones e ideas; como un medio para que conozcan distintas perspectivas y valoraciones en el área de la Informática y la Tecnología

///...RESOLUCIÓN N° 175/11 – R. – ANEXO

<p>MÉTODOS PEDAGÓGICOS</p>	<p>Educativa y desarrollen una actitud de apertura hacia nuevas ideas, logrando así una comprensión informada de la disciplina.</p> <ul style="list-style-type: none"> • Cañón, transparencias, tiza y pizarrón, afiches, software POWERPOINT. Estos se usarán para presentar los temas en las clases expositivas y para que los alumnos presenten sus trabajos de taller. • Blogs a través de Internet para canalizar consultas e inquietudes. • Equipos de computadoras – conexión a Internet para la realización de prácticos.
<p>MÉTODOS DE EVALUACIÓN:</p>	<p>Evaluación Diagnóstica: se llevará a cabo al comenzar la asignatura buscando analizar el punto de partida de los distintos estudiantes a fin de adaptar la enseñanza a esas condiciones, ya que se parte del supuesto que los alumnos necesitan relacionar la nueva información con conocimientos y experiencias previas. La evaluación diagnóstica será especialmente diseñada, oral y objetiva.</p> <p>Evaluación Formativa: es de carácter continuo y está más dirigida a evaluar el proceso de enseñanza-aprendizaje, por lo que se llevará a cabo durante todo el desarrollo de la asignatura.</p> <p>Condiciones para regularizar la asignatura: para regularizar la asignatura los alumnos deben cumplir con los siguientes requisitos:</p> <ol style="list-style-type: none"> 1. Asistencia al 75 % de las clases de Trabajos Prácticos. 2. Aprobación del 100 % de los Trabajos Prácticos. 3. Aprobación de los exámenes parciales. <ol style="list-style-type: none"> 3.1. Requisitos previos: para rendir cada examen parcial el alumno deberá tener aprobados los Trabajos Prácticos realizados con anterioridad a los mismos, pudiendo adeudar como máximo uno de ellos, sea por ausencia a clase o por desaprobación del mismo. 3.2. Número y temario: Se establecen dos (2) parciales. El temario incluirá problemas similares a los desarrollados en las clases y podrán incluir preguntas conceptuales sobre aspectos teóricos. 3.3. Fechas: Las fechas de los exámenes parciales serán fijadas en la Planificación de la Asignatura. 3.4. Evaluación: En el temario se otorgará el puntaje de modo de totalizar 10 puntos, fijándose en seis (6) puntos el mínimo a obtener para aprobar el examen. La calificación conceptual será "Aprobado" o "Desaprobado" según corresponda. 3.5. Recuperatorio: cada alumno tendrá derecho a dos recuperatorios. 3.6. Validez de la regularidad: Obtenida la condición de alumno regular de acuerdo con los requisitos anteriores, la misma tendrá validez por el término de cinco (5) cuatrimestres lectivos, pudiendo rendirla como tal en cualquiera de los turnos de exámenes ordinarios o extraordinarios que se habiliten, pero en un número máximo de seis (6) oportunidades.

<p>MÉTODOS DE EVALUACIÓN</p>	<p><u>De la aprobación mediante Examen Final</u></p> <p><i>Alumno Regular</i></p> <ul style="list-style-type: none"> • Requisitos: Tener acreditada su condición de alumno regular en la asignatura y cumplir con el Régimen de Correlatividades establecido en el Plan de Estudio de la Carrera. • Modalidad: El examen versará sobre el contenido total del programa vigente al momento de la regularización. Revestirá el carácter de teórico o teórico - práctico, escrito u oral. <p><i>Alumno Libre:</i></p> <ul style="list-style-type: none"> • Se hará cumplir los Artículos 29° a 33° de la Resolución Nº 007/09 R. <p>Criterios de Evaluación</p> <p>Los criterios de evaluación son:</p> <ul style="list-style-type: none"> • El nivel de comprensión de los conceptos básicos sobre Procesamiento, Algoritmos, Técnicas, Heurísticas e Informática • El correcto diseño de soluciones para problemas computacionales • La adecuada ejemplificación de los conceptos aprendidos. • La adecuada interpretación del problema. • La aplicación de herramientas tecnológicas con eficiencia • La presentación (la documentación entregada deberá ser clara, libre de errores de ortografía, ordenada, concisa y acotada a lo que se le solicita). <p>En la exposición se evaluará:</p> <ul style="list-style-type: none"> • Que sea ordenada. • Que los alumnos conozcan el problema y la solución propuesta. • Que sea clara. • Que se utilicen herramientas adecuadas durante la exposición. <p>Promoción: la asignatura admite la promoción de la materia, bajo los siguientes lineamientos:</p> <ul style="list-style-type: none"> • Asistencia del 80% a las clases teóricas y prácticas • Aprobar los parciales y obtener una nota promedio de ocho (8). • Presentación del 100% de trabajos prácticos estipulados por la asignatura.
<p>PROGRAMA ANALÍTICO:</p>	<p>UNIDAD 1: Estructuras de Datos</p> <p>Introducción a al procesamiento de datos y resolución de problemas computacionales. Fases en la Resolución de Problemas. Algoritmos. Partes, propiedades, elementos, clasificación y heurísticas. Técnicas de diseño de gráficas de problemas computaciones. Estructuras básicas de control: secuencial, condicional y repetitiva.</p> <p>Complejidad Computacional. Teoría de la Complejidad Computacional. Análisis - Eficiencia de Algoritmos. Clasificación de algoritmos. Comportamiento y Orden de algoritmos. Función de costo. Función de evaluación. Propiedades. Medidas en tiempo y espacio. Elección de la función de evaluación.</p> <p>Estructuras de datos. Clasificación según la complejidad, la representación en memoria. Estructuras estáticas y dinámicas.</p> <p>Estructuras Primitivas. Los enteros: conceptualización desde la Ciencias de la Computación. Alcances desde la programación.</p>

<p>PROGRAMA ANALÍTICO</p>	<p>Operaciones. Formas de representación de los enteros en memoria. Tipos de datos flotantes. TAD. Estructuras definidas por el programador. Los booleanos. Operaciones. Tablas de Verdad. Formas de representación en memoria. Los Caracteres. Características. Operaciones definidas. Formas de representación en memoria.</p> <p>UNIDAD 2: Estructuras Primitivas Estructuras de Datos Simples. Cadenas. Instrucciones básicas de manipulación de cadenas en programación. Operaciones con cadenas. Delimitación de cadenas. Alfabeto y Vocabulario. Almacenamiento de cadenas en memoria. Los arreglos. Propiedad de Ordenación. Mapeo. Subíndices. Límite Superior e Inferior. Clasificación. Rango de un arreglo. Operaciones. Representación en memoria de arreglos unidimensionales, bidimensionales y multidimensionales. Representación por fila y por columna en matrices. Matrices esparcidas. Matriz traspuesta y triangular. Implementación de polinomios en arreglos. Registros. Características. Analogías con los arreglos. Llaves de identificación. Archivos. Representación en memoria. Búsqueda y Ordenamiento. Argumentos. Resultados. Búsqueda secuencial. Mejor y Peor caso. Comparaciones promedio. Búsqueda binaria. Requisitos. Ordenamiento interno y ordenamiento externo. Algoritmos de Ordenamiento: por selección, por Selección con Intercambio, por Inserción. Comparaciones, ventajas y desventajas. Método de la Burbuja. Método del Quicksort: objetivos, comparaciones y rendimiento.</p> <p>UNIDAD 3: Estructuras Complejas: Pilas y Colas Estructuras de Datos Complejas Lineales: Pilas. Aplicaciones de Pilas en Sistemas Informáticos. Conceptos, operaciones. Tope. Longitud. Metodología de inserción y supresión de elementos. Alternativas gráficas de representación. Restricciones del alojamiento de una pila en un arreglo. Lógica de apareamiento de paréntesis. Procedimientos recursivos. Papel cumplen las pilas en el tratamiento de funciones recursivas. Notación infija y postfija. Algoritmo de conversión. Almacenamiento de pilas en memoria. Coexistencia de dos pilas en memoria. Colas. Frente y Fondo de una cola. Longitud. Aplicaciones de colas en sistemas informáticos. Operaciones básicas para manipulación de colas. Modalidades de representación gráfica. Alojamiento de colas en arreglos. Problemas. Cola circular. Procedimiento de inserción y eliminación de elementos en colas circulares Comportamiento FIFO de una cola. Parámetros para medir el comportamiento de una cola. Teoría de colas. Bicolos y colas de prioridad.</p> <p>UNIDAD 4: Estructuras Complejas: Listas Estructuras de datos Complejas Lineales. Listas. Representación secuencial y no secuencial. Costos en que debe incurrir el programador al utilizar las estructuras encadenadas. Listas encadenadas. Representación. Nodo. Lista Vacía. Apuntadores. Variables tipo apuntador. Operaciones con punteros. Operaciones con listas: inserción y Supresión de nodos. Manejo del espacio disponible. Almacenamiento compartido. Pila de disponibilidad. Manipulaciones con listas: localización de un nodo, inserción al final de la lista, inversión de una lista. Listas encadenadas circulares. Características del nodo principal. Listas doble encadenadas. Elementos. Operaciones de inserción y</p>
----------------------------------	---

<p>PROGRAMA ANALÍTICO</p>	<p>remoción de nodos con listas doble encadenadas. Arreglos dispersos en listas encadenadas.</p> <p>UNIDAD 5: Estructuras Complejas: Arboles y Grafos Estructuras de Datos Complejas No Lineales. Grafos y Arboles. Grafos. Características. Grafo nulo. Aristas. Condiciones. Trayectorias. Longitud de la trayectoria. Ciclo: condiciones. Grafos dirigidos. Grafos implementados en listas encadenadas. Directorio de nodos. Aristas ponderadas. Recorrido de grados: en amplitud y en profundidad. Caminos críticos. Arboles. Consideraciones generales.: arboles enraizados, subárboles, raíz, árbol nulo, aristas, grado, altura y peso. Propiedades. Formas de representación. Arboles binarios. Arboles generales. Recorrido de un árbol. Recorrido en orden y post orden. Arboles binarios enhilados. Búsqueda directa. Arboles balanceados por su altura y por un límite.</p>
<p>BIBLIOGRAFIA</p>	<p><u>Obligatoria:</u></p> <ul style="list-style-type: none"> - AHO, "Estructuras de Datos y Algoritmos", Editorial PEARSON EDUCACION, Edición 1988 - Alcalde, Eduardo y García, Miguel. "Metodología de la programación. Aplicaciones en Basic, Cobol y Pascal" Edit. Mc. Graw Hill,. México, 1998 - Cairo Battistutti Osvaldo, "Metodología de la Programación", Editorial Alfaomega Grupo Editor Edición 2005 - Hernández Figueroa Zenon José, Diaz Roca Margarita , Gonzalez Dominguez Jose Daniel , Perez Aguilar Jose Rafael , "Fundamentos de las Estructuras de Datos", Editorial Paraninfo. 2005 - Joyanes Aguilar, Luis."Metodología de la programación" Edit.Mc Graw Hill, España,2003. - Loomis, Mary.. "Estructura de datos y organización de archivos", 2nd edition. Prentice Hall, 1991 - Rodríguez Almeida, Miguel A." Metodología de la programación a través del pseudocódigo" Edit. Mc. Graw Hill,. México, 1997 - Sanders, Donald H. "Informática Presente y Futuro." Edit.Mc. Graw Hill,. México, 200 - Ureña, Sánchez, Martín y Mantas "Fundamentos de Informática. " Edit.Ra-Ma, México,1999 - Watkins "Solución de problemas por medio de computadoras." Edit. Limusa, México, 2001 <p><u>Complementaria:</u></p> <ul style="list-style-type: none"> - Enciclopedia Temática de la Informática - Tomo 1. Edit. Maveco S.A. España, 1987. - Presser, Cárdenas y Marín. Ciencias de la Computación. Vol.1 Edit. Limusa, México., 1979 - Manual del Sistema Operativo Window XP Microsoft - Manual de QBASIC Microsoft - Kemeny, John G. Programación Basic Edit. Continental S.A. 1984 - Joyanes Aguilar, L Programación en TURBO PASCAL Versiones 5.5, 6.0 y 7.0. 2º Edición Edit. Mc. Graw Hill,. España, 1993.

(*) Sujeto a cualquier modificación del Plan de Estudio

