

Universidad Tecnológica Nacional - Facultad Regional Córdoba - Departamento de Sistemas

Modalidad Académica

Asignatura: **Algoritmos y Estructuras de Datos**

Area: **Programación**

Coordinador de cátedra: **Ing. Tymoschuk, Jorge Pablo**

Fundamentación : En ninguna actividad profesional se "nace sabiendo". En general, son el resultado de muy largos procesos de avance, experimentación, búsqueda y evaluación de las mejores opciones. Muchas veces, sobre todo en la "ciencias blandas", coexisten prácticas diversas, incluso enfrentadas.

No es diferente en nuestra asignatura. Existe el modelo de computador de Von Newman, que parece muy difícil de superar. A su alrededor se instalan con éxito los lenguajes que siguen el paradigma imperativo, y evolucionan. Estos lenguajes, inicialmente muy cerca de la Máquina, esforzadamente no han dejado de acercarse al Hombre. Sin embargo, todavía están lejos del lenguaje "natural", incluso de sus formas más explícitas.

La resolución de esta "distancia" del lenguaje codificado para la máquina al natural (codificado para el hombre) es uno de los ejes temáticos de la asignatura.

Los dos párrafos anteriores han sido extraídos de la última planificación presentada. Y este año estamos dando un paso importante en la reducción de esta distancia.

El paso consiste en enfocar el contenido algorítmico de esta asignatura bajo la óptica de la Programación Orientada a Objetos. (POO).

Hace ya más de un par de décadas que este paradigma viene afirmándose como la forma correcta de encarar desarrollos de software importantes. Lenguajes actuales muy difundidos de programación (Por citar unos pocos: Java, Power Builder, Delphi) no contemplan la posibilidad de realizar desarrollos de programación en una modalidad que no fuese la POO.

La Programación Estructurada, paradigma en el cual hemos desarrollado los contenidos de nuestra asignatura hasta el presente, es la base, el cimiento sobre la cual se construye el nuevo paradigma. Sus herramientas están muy presentes en los métodos que caracterizan el comportamiento de los objetos. Pero la forma de organizar la programación es diferente, el edificio terminado se ve diferente.

Este año, al dar las clases a recursantes en el curso alterno, "contracurso", en varios temas hemos hecho "paralelos", resolución de problemas en ambas modalidades. Los alumnos entienden bien, pero prefieren mantenerse en terreno conocido, no arriesgar.

Hasta ahora, y como consecuencia de la desactualización del contenido curricular, se ha dictado toda la parte algorítmica en primer año, luego en Paradigmas se ve Orientación Objetos, se resuelven algunos pocos tópicos de AED en esta orientación, y luego se tratan temas nuevos. Digamos que al alumno "le queda pendiente" el traslado de lo que sabe hacer en forma estructurada al enfoque OO.

Al incorporar OO objetos a AED estamos solucionando esta falencia. Por tornarse el contenido demasiado extenso para AED trasladamos un por de puntos del contenido algorítmico de AED (Arboles y Grafos) a Paradigmas, donde vienen muy bien para ejemplificar algoritmos en lenguaje Java.

Objetivos Generales.

. Satisfacer, en lo que le compete, los aspectos del Perfil Profesional relacionados con la Programación, de manera que le quede al alumno un máximo provecho del conocimiento impartido.

. Desarrollar la capacidad de razonamiento y lógica, abordando problemas reales, analizándolos, definiendo las estructuras de clases y objetos, definiendo el comportamiento de estas clases, codificando los métodos necesarios, verificando su correcto funcionamiento.

CONTENIDOS

OBJETIVOS DE LA SEMANA 01 (cuatr.)

Clase teórica

- Presentación de la materia, condiciones de cursado, regularización, examen, etc.
- Objetivos de la materia.
- Repaso conceptos vistos en Matemática Discreta

UNIDAD I - PROBLEMAS CON TIPOS DE DATOS SIMPLES

OBJETIVOS DE LA SEMANA 02 (cuatr.)

Clase teórica

- Resolución de problemas (Análisis del problema, Diseño del algoritmo y resolución del problema por computadora).
- Análisis y programación orientada a objetos.

Clase práctica

- Ejercicios de identificación de entidades.
- Repaso de Matemática Discreta (diagramas de flujo).

OBJETIVOS DE LA SEMANA 03 (cuatr.)

Clase teórica

- Algoritmos (Concepto, representación gráfica, escritura de algoritmos, estructuras de control)
- Programas (Concepto, Organización, Programas tipos, Elementos básicos y Esquema general).
- Lenguajes de programación.

Clase práctica

- Ejercicios con clases (en diagramas) y diagramas de flujo.

OBJETIVOS DE LA SEMANA 04 (cuatr.)

Clase teórica

- Lenguaje C++.
- Elementos del lenguaje.
- Clases en C++ (Estructura general, declaración y definición, cuerpo de una clase. Instancias de una clase (objetos) (acceso a los miembros, la vida de un objeto), declaración de miembros de clases (modificadores de acceso a miembros de clases) y atributos.

Clase práctica

- Ejercicios de clases con código y diagramas de flujo.
 - Mínimo y Máximo en una secuencia de números
 - Pares, Impares en una secuencia de números

OBJETIVOS DE LA SEMANA 05 (cuatr.)

Clase teórica

- Clases en C++ (continuación métodos: Declaración y definición, el cuerpo, llamadas a métodos el objeto actual (this), métodos especiales: sobrecargados, constructores y destructores).
- Problemas tipos.
- Funciones básicas de caracteres.

Clase práctica

- Ejercicios de clases con código y diagramas de flujo.
 - Secuencia ascendente de caracteres
 - Términos en serie aritmética
 - Tratamiento de caracteres

UNIDAD II - PROBLEMAS SIMPLES RELACIONANDO OBJETOS

OBJETIVOS DE LA SEMANA 06 (cuatr.)

Clase teórica

- Implementado relaciones "tiene un" (Class Alternan "tiene un" Caracter.)
- Tratamiento de palabras (Class Palabras "tiene un" Caracter)
- Tratamiento de palabras, enunciados complejos.
- HERENCIA - INTRODUCCION, BENEFICIOS DE LA HERENCIA, REUSABILIDAD DEL SOFTWARE, COMPARTICION DE CODIGO, CONSISTENCIA DE LA INTERFAZ, COMPONENTES DE SOFTWARE, MODELADO RAPIDO DE PROTOTIPOS, OCULTACION DE INFORMACION, POLIMORFISMO
- HEURISTICAS PARA CREAR SUBCLASES. HERENCIA Y JERARQUIA DE CLASES. TIPOS DE HERENCIA. CLASES DERIVADAS.

Clase práctica

- class Caracter
- Class Alternan

OBJETIVOS DE LA SEMANA 07 (cuatr.)

Clase teórica

- DECLARACION DE UNA CLASE . DERIVADA. ATRIBUTOS Y METODOS.
- DERIVACION PUBLICA, PRIVADA, PROTEGIDA. REDEFINICION Y SOBRECARGA DE METODOS.
- HERENCIA SIMPLE. CONSTRUCTORES. DESTRUCTORES. UN EJEMPLO DE JERARQUÍA DE CLASES: HERENCIA SIMPLE. FUNCIONES VIRTUALES .
- CLASES ABSTRACTAS. POLIMORFISMO. APLICACIONES.

Clase práctica

- Class Palabra
- Class Palabra1

UNIDAD III - ARRAYS

OBJETIVOS DE LA SEMANA 08 (cuatr.)

Clase teórica

- DECLARACIÓN DE ARRAYS. SUBINDICACIÓN. TAMAÑO DE ARRAYS. VERIFICACIÓN DEL RANGO DE ÍNDICES. INICIALIZACIÓN DE UN ARRAY.
- CONCEPTO DE CADENA. LECTURA. ARRAYS y CADENAS COMO PARÁMETROS DE FUNCIONES. USO DE PUNTEROS PARA PASAR UNA CADENA. ASIGNACIÓN DE CADENAS. La función strlen()

Clase práctica

- IMPLEMENTACIÓN DE UN CONJUNTO.

OBJETIVOS DE LA SEMANA 09 (cuatr.)

Clase teórica

- Las funciones strcat y strncat, strcmp y strncmp. Una clase tratando cadenas. BÚSQUEDA EN ARRAYS. UNA CLASE ITEMS. BÚSQUEDA SECUENCIAL Y BINARIA. COMPARACIÓN ENTRE BÚSQUEDAS.
- ORDENAMIENTO - Introducción. Algoritmos de ordenamiento básicos y mejorados. Ordenamiento por Método Burbuja. Ordenamiento por Método "sacudida". Ordenamiento por Método "Peinado". COMPARANDO TIEMPOS DE ORDENAMIENTO.

Clase práctica

- class MisCadenas. class Items. class BusqSeq. class BusqBin.
- class OrdBurb. class OrdSac. class OrdPein. class CalcTime.

OBJETIVOS DE LA SEMANA 10 (cuatr.)

Clase teórica

- ARRAYS MULTIDIMENSIONALES. INICIALIZACIÓN, ACCESO. LECTURA. USO DE BUCLES. ARRAYS DE MAS DE DOS DIMENSIONES. Caso practico: matriz de temperaturas.

Clase práctica

- class Temperat. class DiasSem. class TempMas

UNIDAD IV – PILAS, COLAS, LISTAS

OBJETIVOS DE LA SEMANA 11 (cuatr.)

Clase teórica

- INTRODUCCION. CONCEPTO DE PILA, COLA Y LISTA. CLASE NODO Y LISTA ENLAZADA DE NODOS. Implementando una clase Pila. Implementando una clase Cola.

Clase práctica

- class Nodo. Class ListEnl. Class Pila. Class Cola.

OBJETIVOS DE LA SEMANA 12 (cuatr.)

Clase teórica

- Implementando una clase Lista. Enunciados combinando estructuras de datos.

Clase práctica

- Class Lista.

OBJETIVOS DE LA SEMANA 13 (cuatr.)

Clase teórica

- Enunciados combinando estructuras de datos.

Clase práctica

- class Enunc01 .

Unidad V - ARCHIVOS

OBJETIVOS DE LA SEMANA 14 (cuatr.)

Clase teórica

- **Concepto. Archivos secuenciales. Acceso secuencial y directo.**
- **Archivos de texto y estructurados.**
- **Entrada, salida utilizando funciones estándar.**

Clase práctica

- class Registro. class Archivo. Uso de archivos de entrada y salida

OBJETIVOS DE LA SEMANA 15 (cuatr.)

Clase teórica

- Acceso directo en archivos secuenciales. Aplicaciones con varios archivos. Archivos conteniendo arrays. Integración con objetos de otras clases.

Clase práctica

- Casos integrando archivos y arrays
- Casos integrando archivos y estructuras lineales.

OBJETIVOS DE LA SEMANA 16 (cuatr.)

Clase teórica

- Integración con objetos de otras clases. Consultas. Regularización

Clase práctica

- Integración con objetos de otras clases. Consultas. Regularización

Bibliografía

Material Teórico/Práctico de la Cátedra impreso.

Material Teórico/Práctico de la Cátedra en WWW (labsys.frc.utn.edu.ar / Sitios de las cátedras / AED 2004)

Programación en C++, Luis Joyanes Aguilar, Mc Graw Hill/Interamericana de España.

Como programar en C / C++ H.M. Deitel / P.J. Deitel, PRENTICE HALL HIPANOAMERICANA S.A.

Plantel docente

Situación al 06/07/2004, (sujeta a eventuales modificaciones en el inicio de clases)

Curso	<<<==== HORARIOS =====>>>	A CARGO	1er ayudante	2do ayudante
Mañana				
01-202	lun 4,5,6 mar 3,4,5,6 jue 1,2,3	Teicher, Romina	Beth, Federico	Fernandez, Julieta
03-206	mie 1,2,3 jue 4,5,6 vie 3,4,5,6	Frittelli, Valerio	Teicher, Romina	
16-220	lun 1,2,3 mar 4,5,6 mie 3,4,5,6	Carena, Gonzalo	Beth, Federico	Martinez, Valeria
02-204	lun 1,2,3 mie 1,2,3 vie 1,2,3,4	Carena, Gonzalo	Ligorria, Karina	Romani, German
04-208	mie 4,5,6 jue 1,2,3 vie 1,2,3,4	Peralta, Tomás	Parraga, Adriana	Palaver, Jorge
12-217	lun 1,2,3 mar 4,5,6 jue 1,2,3,4	Ligorria, Karina	Berton, Cynthia	Rodriguez, Jauregui
14-218	lun 3,4,5,6 Jue 5,6 vie 3,4, 5,6	Guzman, Analía	Ligorria, Karina	Palmeri, Nicolas

15-232	lun 3,4,5,6	mar 1,2,3	vie 1,2,3	Marciszack, Marc. Corso, Cynthia	Castillo, Julio Serrano, Diego
--------	-------------	-----------	-----------	----------------------------------	-----------------------------------

Tarde

05-202	mie 4,5,6	jue 1,2,3	vie 1,2,3,4	Arias, Silvia	Ligorria, Karina	Porfirio, Sebastian Romani, Germán
06-204	lun 1,2,3	mie 1,2,3	vie 1,2,3,4	Tymoschuk, Jorge	Brochero, Carlos	Villagrán, Nestor Palmeri, Nicolas
07-206	Cuatrimestre Alterno			Tymoschuk, Jorge	Ligorria, Karina	Solana, Dante Jáuregui, David

Noche

09-202	mar 4,5,6	mie 1,2,3	jue 1,2,3,4	Vazquez, Juan C.	Tartabini, Marcela	Serrano, Diego
10-204	mar 1,2,3,4	mie 4,5,6	vie 4,5,6	Serra, Silvio	Párraga, Adriana	Brochero, Carlos Porfirio, Sebastian
C.Paz	mar 1,2,3,4,5,6,		jue 3,4,5,6	Cresta, Tomás	Solana, Dante	Diharce, Bernabe

Condiciones de Regularización

a) Realizar los trabajos prácticos individuales y grupales propuestos por el Jefe de Trabajos Prácticos respectivo, individuales (Escrito) y grupales (Medio magnético).

A partir del año 1999, la cátedra pasa a exigir la realización de 2(dos) prácticos grupales antes de cada parcial, los cuales, mientras entregados al docente a cargo de los prácticos antes y hasta la fecha por él definida serán devueltos a los alumnos con las correcciones y observaciones pertinentes, permitiendo de esta manera el adecuado proceso de enseñanza/aprendizaje y familiarizando al alumno con los aspectos considerados en las evaluaciones parciales.

A partir del año 2004, se dispone en el sitio labsys.frc.utn.edu.ar /Sitios de las cátedras/ AED 2004, del siguiente material:

- Programa / Cronograma.
- Material por unidad
 - Carpeta Apunte: Texto teórico / practico, formato pdf
 - Carpetas Clases codificadas C++ : Para ser bajadas por los alumnos e incluidas en sus prácticos.
 - Carpeta Prácticos: Enunciados propuestos por la cátedra a desarrollar por los alumnos.

El promedio de prácticos proporciona una nota, la cual será promediada con las notas de los parciales. Esta nota, resultado de la entrega y aprobación de todos los prácticos solicitados, es un **requisito imprescindible para la regularización**.

b) Aprobar 2 (dos) parciales individuales. Existirá un parcial recuperatorio para quien haya aprobado un único parcial, el tema a ser desarrollado será independiente del parcial a recuperar. La nota del recuperatorio **no reemplaza** la del parcial recuperado. Se regulariza materia con dos parciales y prácticos aprobados y promedio no inferior a 4(cuatro).

Las fechas previstas son :

1er. parcial (Unidades 1,2) . Consistirá en la codificación de métodos de objetos de una clase. La clase puede contener y heredar objetos de otras clases.

Sábado 18 de setiembre.

2do. parcial (Unidades 2, 3, 4) Consistirá en la codificación de métodos de objetos vistos en estas unidades, integrándolos. Por ejemplo, array / listas, listas / archivos, array /archivos, este tipo de combinaciones.

Sábado 13 de noviembre.

Recuperatorio: se tomará en el primer o tercer turno de exámenes de noviembre/diciembre, (opción del alumno), exclusivamente. Los temas a evaluar son independientes del parcial a recuperar.

Se fija como única fecha la del 6to llamado, (semana 49 del Calendario Académico) Se puede recuperar un único parcial.

Los temas de los parciales son definidos por el profesor a cargo de curso, contemplando eventuales problemas en el dictado de la asignatura. Deben ser elevados a la Dirección de la Cátedra para su consideración/unificación.

Examen final

Los temas de los exámenes finales son elaborados por una comisión interna de la cátedra y distribuidos a los tribunales, contemplándose la existencia de los distintos horarios. El alumno es evaluado contemplándose su desempeño en la cursada.

- Promedio igual o mayor a 8 (ocho) y ninguna nota menor a 7(siete), promueven la evaluación práctica.
- Promedio inferior a 8 (ocho): examen práctico y teórico completo. Se toma todo el programa; contenidos teóricos/prácticos de acuerdo al presente programa de la cátedra. Cualquier tribunal de la cátedra puede examinar a cualquier alumno.

TRABAJOS DE INVESTIGACIÓN

La idea es la de encauzar y premiar la dedicación a la investigación en aquellos alumnos que están dispuestos a invertir tiempo en esta actividad. El tema podría ser una iniciativa del grupo o de la cátedra. En todo caso, será una programación funcionando en computadora en lenguaje C++. (aplicaciones en entorno gráfico, animación, uso de interrupciones del sistema operativo, etc.).

El profesor a cargo del curso y el jefe de prácticos son responsables por:

Definir la forma de presentación del trabajo. Controlar la originalidad del trabajo. Orientar, asesorar al grupo. Evaluar el trabajo realizado. Presentar un informe al director de cátedra.

Como motivación a estos trabajos, la cátedra considerará promovidos a aquellos alumnos que:

- Promedio igual o mayor a 8 (ocho) y ninguna nota menor a 7(siete)
- Se apruebe el trabajo de investigación.

Ing. TYMOSCHUK, Jorge
Director de cátedra