



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

ALGORITMOS, DATOS Y PROGRAMAS

Carrera: *Licenciatura en Informática*
Plan 2007
Licenciatura en Sistemas
Plan 2007

Año 2010

Año: 1°
Duración: *Anual*
Coordinador: *Ing. Armando De Giusti*
Profesores: *Esp. Cristina Madoz, Esp. Gladys Gorga, Dra. Laura De Giusti, Lic. Maria Virginia Ainchil, Mg. Alejandro Héctor González*
Hs. semanales: *6 hs.*

Objetivos:

- Analizar problemas resolubles con computadora, poniendo énfasis en la modelización, abstracción de funciones y en la descomposición funcional de los mismos. Obtener una expresión sintética y precisa de los problemas, con una documentación de una metodología de trabajo por el alumno.
- Estudio, expresión simbólica, implementación y evaluación de algoritmos, orientando los mismos a la resolución de las partes (módulos) en que se descomponen los problemas, a partir de un paradigma procedural/imperativo.
- Introducción de las nociones de estructuras de datos, tipos de datos y abstracción de datos.
- Introducción de los conceptos de corrección y eficiencia de algoritmos.
- Introducción de los conceptos básicos de un segundo paradigma de programación (orientación a objetos) con énfasis en la noción de reusabilidad.
- Combinar los elementos mencionados anteriormente a fin de que el alumno complete el ciclo del problema a su solución con computadora, analizando simultáneamente algoritmos y datos.

A- Introducción. Conceptos básicos

Definiciones.

Modelización de problemas del mundo real.

Del problema real a su solución por computadora.

Software.

B - Algoritmos. Acciones elementales

Estructuras de control. Modelo de máquina abstracta.

El robot como máquina abstracta. Operaciones elementales del robot.

Estructura esquemática de un programa para una máquina abstracta (robot).

Corrección de algoritmos. Importancia de la verificación.

Eficiencia de un algoritmo.



Importancia de la documentación de un algoritmo.
Relación de los conceptos anteriores con el modelo de máquina abstracta.
Ejemplos.

C –Datos y Tipos de datos

Constantes y variables.
Tipos de datos simples y compuestos.
Tipos de datos primitivos.
Tipos de datos definidos por el usuario.
Funciones predefinidas.
Tipos ordinales.

D- Modularización. Procedimientos y funciones. Parámetros

Descomposición de problemas. Utilidad e importancia.
La noción de reusabilidad.
Subprogramas o módulos.
Procedimientos.
Funciones.
Conceptos de argumentos y parámetros.
Conceptos de variables locales y variables globales.
Procedimientos y funciones con parámetros.
Manejo de memoria en ejecución.

E- Estructuras de datos

Introducción y clasificación de las estructuras de datos.
Registros.
Arreglos.
Arreglos. Operaciones con arreglos de una dimensión.
Matrices. Tratamiento de información estructurada en vectores y matrices.
Algoritmos de búsqueda.
Algoritmos de ordenación. Ordenación por índice.

F- Estructura de datos enlazadas: listas.

Alocación dinámica. Punteros.
Listas. Operaciones con listas.
Listas doblemente enlazadas y circulares. Características y operaciones.

G- Recursividad

Características.
Ejecución de un programa y la pila de activación. Manejo de memoria en ejecución.
Análisis comparativo entre soluciones iterativas y recursivas.
Ejemplos.



H- Concepto de Corrección. Análisis de algoritmos: concepto de eficiencia

Concepto de corrección. Técnicas para medir corrección. Verificación de Programas.

Concepto de eficiencia.

Análisis de eficiencia de un algoritmo.

Análisis de algoritmos según su tiempo de ejecución y su utilización de memoria.

Análisis de Algoritmos: Análisis asintótico, comportamiento en el mejor caso, caso promedio y peor caso. Notación $O()$.

Análisis de eficiencia en algoritmos recursivos.

Análisis de eficiencia en algoritmos de búsqueda y ordenación sobre vectores.

Métodos de ordenación eficientes.

Algoritmos numéricos y propagación de error.

J- Estructura de datos no lineales: árboles.

Introducción al concepto de datos no lineales.

Terminología y definiciones básicas del tipo de dato árbol.

Arboles binarios. Representación y operaciones.

Arboles binarios ordenados. Representación y operaciones.

Problemas que combinen árboles, listas y arreglos.

K- Tipos de datos abstractos

Abstracción de datos.

Conceptos sobre tipos de datos.

Módulos, interfaz e implementación. Encapsulamiento de datos.

Diferencia entre tipo de dato y tipo abstracto de dato.

Requerimientos y diseño de TADs. Ejemplos TAD pila, TAD cola.

L- Introducción a la Programación Orientada a Objetos

Motivación. Reusabilidad de soluciones.

Abstracción de datos y procesos.

La noción de Objeto. Operaciones (métodos) aplicables a un objeto.

Concepto de clases e instancias.

Noción de herencia. Relación con el re-uso.

Aplicaciones.

Características de los lenguajes enfocados a POO.

M - Conceptos iniciales de concurrencia

Motivación: arquitecturas de computadoras actuales, aprovechamiento de los procesadores.

Definiciones.

Ejemplos.



Metodología de Enseñanza

El curso tiene clases teóricas, explicaciones de práctica y prácticas experimentales.

Existe una reglamentación detallada que es aprobada por el Consejo Directivo de la Facultad y conocida por los alumnos desde el inicio de año.

Las clases teóricas son dictadas por los Profesores de la asignatura y no son obligatorias, aunque los alumnos que aspiren a la promoción de la teoría deben registrar asistencia a las mismas. El acceso a la promoción de la teoría depende de los resultados obtenidos en el curso de Expresión de Problemas y Algoritmos del Curso de Ingreso.

Las explicaciones de práctica son dictadas por los Jefes de Trabajos Prácticos, quienes también coordinan las actividades de los Auxiliares Docentes de la asignatura.

Las actividades prácticas son obligatorias. Comprenden el análisis de diseño de problemas del mundo real, la expresión sintética de los mismos y el diseño e implementación de algoritmos en diferentes lenguajes.

Los alumnos disponen de diferentes materiales didácticos para el desarrollo de las distintas actividades propuestas. Se puede mencionar entre ellos, la Biblioteca que integra servicios por Internet, los Laboratorios con PCs para trabajo experimental, los recursos de Software que se utiliza en las prácticas experimentales, el EVEA WebUNLP y aulas con equipamiento multimedia e Internet.

Propuesta de evaluación

Para aprobar el curso de Trabajos Prácticos, los alumnos deben aprobar 2 exámenes parciales correlativos. Para la aprobación de la asignatura deberá rendir un examen final escrito.

Los alumnos se estructuran en grupos (Ingresantes/recursantes). Todos los alumnos tienen un seguimiento continuo, utilizando EVEA WebUNLP.

La cursada presenta tres modalidades:

Modalidad A (no aprobaron la prueba diagnóstica del ingreso), deben realizar tareas grupales. Estas actividades se registran y sus resultados son conocidos y explicados, si bien no condicionan la presentación a los exámenes parciales.

Modalidad B (aprobaron la prueba diagnóstica del ingreso) pueden optar por rendir 2 exámenes teóricos y en caso de aprobarlos deben presentarse a un coloquio, donde defenderán los temas que no han sido desarrollados satisfactoriamente en estos exámenes.

Modalidad C (alumnos recursantes), se sigue una cursada semipresencial utilizando el entorno WebUNLP. Deben realizar trabajos grupales que son evaluados y los resultados de los mismos son conocidos y explicados y serán tenidos en cuenta para la presentación a los exámenes parciales obligatorios.



Bibliografía Básica

Algoritmos, datos y programas con aplicaciones en Pascal, Delphi y Visual Da Vinci.
De Giusti, Armando et al. 1er edición. Prentice Hall 2001.

Estructuras de Datos y Algoritmos.
Hernández R., Dormido R., Lazaro J. Ros S. Pearson Education. 2000.

Introduction to algorithms
Comen, Leiserson. MIT Press 2001.

Estructuras de Datos y Algoritmos.
Aho Alfred, Hopcroft John y Ullman Jeffrey. Addison Wesley Publishing Company. EUA. 1998.

Análisis y Diseño Orientado a Objetos con Aplicaciones.
Booch Grady. Addison Wesley. 1998.

Programación en Pascal
Joyanes Aguilar, Luis. Mc Graw Hill. 2006

Fundamentos de Programación. Libro de Problemas.
Joyanes Aguilar L., Fernandez M., Rodríguez L. Mc Graw Hill. 2003.

Programación Orientada a Objetos.
Joyanes Aguilar, L. Mc Graw Hill. 1998

Aprendiendo Programación Orientada a Objetos
Anthony Sintés, Prentice Hall. 2002.

Data structures, algorithms and software principles.
Standish, T. A. Addison Wesley Publishing Company. 1994.

Estructuras de Datos y Algoritmos
Weiss, M.A. Addison Wesley. 1995.

Fundamentos de Programación.
Joyanes Aguilar L., Fernandez M., Rodríguez L. Mc Graw Hill. 1999.

Multithreaded, Parallel, and Distributed Programming
Andrews, G. Addison Wesley Publishing Company. 2000.



Bibliografía Adicional

Introduction to Computer Science with applications in Pascal.

Garland, S.J. Addison Wesley Publishing Company. 1986.

Estructuras de Datos.

Franch Gutierrez, Xavier. Alfaomega Grupo Editor Argentino. 2002

Estructura de Datos.

Joyanes Aguilar C., Zahonero Martinez I. Mc Graw Hill. 1998.

Fundamentos de Programación. Libro de Problemas en Pascal y Turbo Pascal.

Joyanes Aguilar L., Fernande M., Rodríguez L. Mc Graw Hill. 1999.

Estructuras de Datos. Libro de Problemas.

Joyanes Aguilar L., Fernandez M., Rodríguez L. Mc Graw Hill. 1999.

UML y Patrones. Una introducción al análisis y diseño orientado a objetos y al proceso unificado.

Larman, C. Prentice Hall. 2006.

PASCAL Programming and Problem Solving.

Leestma Sanford. Macmillan Publishing Company. 1984.

Estructuras de Datos.

Lipschutz, S. Mc Graw Hill. 1997.

Programación estructurada en Turbo Pascal 7.

Lopez Roman, L. Alfaomega Grupo Editor Argentino. 1998.

Estructuras de Datos.

Martinez Román, Quiroga Elda. Thomson International. 2002

Estructura de Datos y Algoritmos.

Sisa, Alberto Jaime. Editorial Prentice. 2002.

Pascal Estructurado.

Tremblay, Jean Paul. Mc Graw Hill. 1980.

Software orientado a objetos.

Winbled Ann, Edwards S, King D. Addison Wesley. 1993.

Data structures, algorithms and performance.

Wood, D. Addison Wesley Publishing Company. 1993.