



**UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL ROSARIO**

U.T.N. – FACULTAD REGIONAL ROSARIO

DEPARTAMENTO ACADEMICO: Ingeniería en Sistemas de Información

PROGRAMA ANALITICO DE LA ASIGNATURA: Algoritmos y Estructuras de Datos

PLAN DE ESTUDIOS RESOLUCION Nro: ORD 764/95

HORAS SEMANALES: 5 (cinco) DICTADO: Anual

PROFESOR / DIR. CATEDRA: Ing *Miguel Iwanow*

DIRECTOR DEL DEPARTAMENTO: A.U.S. *Conrado Fernández*

RES.Nº 212/99 (31/08/99) F.R.R.

OBJETIVO GENERAL DE LA ASIGNATURA:

(Conocimientos / habilidades que el alumno deberá lograr al concluir el curso).

Los alumnos desarrollan la destreza o habilidad de formular programas, no solo funcionalmente efectivos, sino fundamentalmente correctos desde un punto de vista metodológico, en procura de la eficiencia durante todo su ciclo de vida.

FUNCION DE LA ASIGNATURA EN EL PLAN DE ESTUDIOS:

Algoritmos y Estructuras de Datos es la primer materia del Área Programación, por lo cual recae en ella la compleja misión de introducir al alumno en la ardua tarea de la programación.

Si bien la carera no tiene una terminalidad orientada a la programación en sí, nuestros profesionales deberán especificar, controlar, orientar las tareas de los programadores. Además de fijarles pautas a sus tareas.

Por lo cual prevalecen como metas de la asignatura los criterios inherentes a la metodología modular por refinamientos sucesivos, coherente con la propia del Análisis y Diseño de los Sistemas de Información.

1- CONTENIDOS

UNIDAD DIDACTICA

EJE CONCEPTUAL:

Comprensión del problema.

OBJETIVOS:

*Que el alumno conozca las partes que integran los problemas. Diferencie distintos tipos de problemas. Sea capaz de formalizar la especificación de problemas.

TEMAS:

- 1. 1. Problemas elementales.
 - 1.1. 1. Clasificación de los problemas.
 - 1.1. 1.1. Problemas de evaluación.
 - 1.1.1.2. Problemas de decisión.
 - 1.1.2. Partes principales de un problema.
 - 1.1.2.1. Objetivo o requerimiento del problema.
 - 1.1.2.2. Datos disponibles. Noción de clase o tipo de datos.
 - 1.1.2.3. Condiciones establecidas.
 - 1. 2. Problemas compuestos. Noción de descomposición.
 - 1.2.1. Pautas básicas para la comprensión del problema.
 - 1.2.2. Especificación formal del problema.

UNIDAD DIDACTICA 2

EJE CONCEPTUAL:

Diseño de algoritmos.

OBJETIVOS:

Que el alumno conozca las propiedades de los autómatas.
Subdivida los problemas en otros más simples.
Aplice las estructuras más adecuadas.
Valore formalización de la especificación de la solución del problema.

TEMAS:

- 2.1. Definición de autómata. Autómata secuencial.
 - 2.1.1. Concepto de entorno y estado.
 - 2.1.2. Concepto de Acción.
 - 2.1.2. 1. Clasificación de las acciones.
 - 2. 1.2.1.1. Acciones simples.
 - 2.1.2.1.1.1. Acciones primitivas del autómata.
 - 2.1.2.1.1.2. Acciones inducidas o definidas por el programador.
 - 2.1.2. 1.2. Acciones compuestas.
 - 2.1.2.1.2.1. Secuencia de acciones.
 - 2.1.2.1.2.2. Secuencia de estados.
- 2.2. Definición de algoritmo.
 - 2.2.1. Representación de los algoritmos.
 - 2.2.1.1. Diagramación esquemática.
 - 2.2. 1.2. Diagramación gráfica. Diagramas NS o de Chapin.
 - 2.2.1. Pautas básicas para el diseño general de un algoritmo.

- 2.3. Concepto de módulo e independencia. Vinculación entre los módulos.
 - 2.3.1. Invocaciones entre los módulos.
 - 2.3.2. Comunicación de datos entre los módulos.
 - 2.3.3. Procedimientos y funciones.
 - 2.3.4. Concepto de diseño descendente.
- 2.4. Estructuras de control.
 - 2.4.1. Estructura secuencial.
 - 2.4.2. Estructuras de interacción.
 - 2.4.3. Estructuras de decisión.
- 2.5. El teorema fundamental de la programación estructurada.
 - 2.5.1. Unicidad del punto de entrada de cada estructura.
 - 2.5.2. Unicidad del punto de salida de cada estructura.
 - 2.5.3. Unicidad de los puntos de entrada y salida de cada módulo.
- 2.6. Especificación recursiva de módulos.

UNIDAD DIDACTICA 3

EJE CONCEPTUAL:

Algoritmos computacionales.

OBJETIVOS:

Que el alumno conozca la función que cumplen: las variables y las asignaciones en el paradigma imperativo.

Interprete la importancia de las expresiones.

Aplique las expresiones más adecuadas. Valore la formalización de la especificación de la solución de un problema.

TEMAS:

- 3.1. Concepto de variable. La operación de asignación.
 - 3.1.1. Asignación interna. Importancia de la adecuación del tipo de datos
 - 3.1.1.1. Definición de expresión.
 - 3.1.1.1.1. Constantes.
 - 3.1.1.1.2. Variables.
 - 3.1.1.1.3. Funciones.
 - 3.1.1.1.3.1. Funciones primitivas
 - 3.1.1.1.3.2. Funciones inducidas o definidas por el programador.
 - 3.1.1.1.4. Operaciones explícitas. Concepto de operador.
 - 3.1.2. Asignación externa o ingreso de datos.
 - 3.1.3. Salida de la información.
- 3.2. Una computadora algorítmica: la prueba de escritorio.
- 3.3. Vinculación con un equipo computacional físico real.
- 3.4. Concepto de programa.
 - 3.4.1. Lenguaje de programación.
 - 3.4.1.1. Noción de lenguaje de máquina.
 - 3.4.1.2. Noción de lenguaje de programación orientados al problema.
 - 3.4.1.3. Noción de lenguaje de programación de alto nivel.
 - 3.4.1.4. Operaciones suplementarias.

UNIDAD DIDACTICA 4

EJE CONCEPTUAL:

Tipo de Datos y Estructuras de datos.

OBJETIVOS:

Que el alumno conozca la función que cumplen las variables y las asignaciones en el paradigma imperativo. Interprete la importancia de las expresiones. Aplique las expresiones más adecuadas. Valore Informatización de la especificación de la solución de un problema.

TEMAS:

- 4.1. Definición de Tipo de Datos. Propiedades de los Tipos de Datos.
- 4.2. Clasificación de Tipo de Datos.
 - 4.2. 1. Tipo de Datos estático.
 - 4.2.1.1. Tipo de Datos simple o escalar
 - 4.2.1.1.1. Tipo de Datos Real.
 - 4.2.1. 1.2. Tipo de Datos Ordinal.
 - 4.2.1.1.2.1. Tipo de Datos Entero
 - 4.2.1.1.2.2. Tipo de Datos Booleano.
 - 4.2.1.1.2.3. Tipo de Datos Carácter.
 - 4.2.1.1.2.4. Tipo de Datos Inducidos o definidos por el programador.
 - 4.2.1.1.2.4.1. Tipo de Datos definidos por enumeración.
 - 4.2.1.1.2.4.2. Tipo de Datos definidos por extensión o subrango.
 - 4.2.1.2. Tipo de Datos estructurado.
 - 4.2.1.2.1. Clasificación de estructuras.
 - 4.2.1.2.1.1. Tipo Registros.
 - 4.2.1.2.1.2. Tipo arreglos.
 - 4.2.1.2.1.3. Tipo Conjuntos.
 - 4.2.1.2.2. Operaciones con las estructuras y con los elementos que las componen.
 - 4.2.1.2.2.1. Tratamiento de las Cadenas de caracteres.
 - 4.2.1.2.2.2. Métodos de búsqueda y ordenamiento de los datos contenidos en estructuras.
 - 4.2.1.2.3. Criterios y Métodos de selección de las estructuras más adecuadas.
 - 4.2.2. Tipo de Datos dinámico. El puntero. Definición de tipos abstractos. Operaciones.
 - 4.2.2.1. Colas.
 - 4.2.2.2. Filas.
 - 4.2.2.3. Árboles.
 - 4.2.2.4. Grafos.
 - 4.2.3. Comparación de las características y las propiedades. Criterios y Métodos de selección de las estructuras más adecuadas.

UNIDAD DIDACTICA 5

EJE CONCEPTUAL:

Implementación en computadoras de la solución de problemas.

OBJETIVOS:

Que el alumno conozca la función que cumple cada uno de los objetos de programación en un lenguaje estructurado. Interprete las sentencias del lenguaje. Aplique las expresiones más adecuadas. Valore la especificación estructurada de los programas.

TEMAS:

- 5.1. Estructura de los programas. Partes que lo componen. Características generales de la sintaxis de un lenguaje estructurado de programación.
- 5.1.1. Encabezamiento.
- 5.1.2. Cuerpo del programa.
- 5.1.2.1. Parte declarativa y de definiciones de los objetos de computación. Representación interna de los datos.
- 5.1.2.2. Parte de especificación de las acciones.
- 5.1.2.2.1. Sentencias simples o primitivas.
- 5.1.2.2.2. Características de los operadores aritméticos, lógicos y de relación.
- 5.1.2.2.3. Características de las expresiones de distinto tipo.
- 5.1.2.2.4. Sentencias compuestas. Implementación de los módulos.
- 5.1.2.2.4.1. Procedimientos.
- 5.1.2.2.4.2. Funciones.
- 5.1.2.2.4.3. Argumentos y parámetros, comunicaciones de datos entre las unidades de programación.
- 5.1.2.2.4. Parámetros por referencia y parámetros por valor.

2- TRABAJOS PRACTICOS

Se realizan una buena cantidad de resoluciones de ejercicios. Algunos pocos en modalidad de demostración. La mayoría en cambio se resuelven en clase en forma conjunta con los alumnos. Además la guía de TP presenta ejercicios para ser resueltos por el alumno en forma individual.

La resolución de ejercicios se propone con el fin de afianzar el conocimiento de las características y propiedades de los distintos conceptos y objetos vistos en las clases teóricas. Estos ejercicios se diseñan para exaltar las propiedades en forma individual.

Para todos los temas se prevé una gran cantidad de resoluciones de problemas, aplicando las propiedades y características de los elementos en forma combinada por lo cual se convierten en verdaderas herramientas que facilitan su resolución.

La dificultad y complejidad se pretende graduar, de manera tal de ir aumentándola a medida que los alumnos se vayan afianzando en la disciplina. Dado que la dificultad en la programación es una manifestación de la distancia que separa a las características internas del problema con la capacidad del autómata que se usa para su resolución. Resulta adecuado comenzar con la resolución de problemas por medio de autómatas virtuales muy poderosos. Luego durante el transcurso del curso, a medida que la destreza de los alumnos lo vaya permitiendo, se encara a la resolución de los mismos problemas mediante otros autómatas más básicos lo cual ofrece la posibilidad consolidar el concepto de refinamiento sucesivo en la solución de un problema.

3- BIBLIOGRAFÍA

3.1. BIBLIOGRAFIA BÁSICA

3.1.1	Introducción a la programación sistemática	WIRTH, Nicklaus	EL ATENEO
3.1.2	Introducción a la programación y a las estructuras de datos	BRUNSTEIN Y GIOIA	EUDEBA
3.1.3	Introducción a la programación y estructura de datos	BIONDI Y CLAVEL	MASSON
3.1.4	Introducción a la computación y programación estructurada	GUTIERREZ, Levin	MC GRAW HILL
3.1.5	Fundamentos de la programación	JOYANES AGUILAR, Luis	MC GRAW HILL

3.1 BIBLIOGRAFIA GENERAL

3.2.1	El arte de programar ordenadores VOL. 1	KNUTH D. E	PARANINFO
3.2.2	Algoritmos y representación de datos tomo1	LUCAS, PEYRIN Y SCHOLL	MASSON
3.2.3	Introducción a la programación y a la solución de problemas.....	SCHNEIDER, WEINGART Y PERL	LIMUSA

3.3. BIBLIOGRAFIA ESPECIFICA

3.3.1	PASCAL Manual del usuario e informe	WIRTH, Nicklaus	EL ATENEO
3.3.2	El arte de programar ordenadores VOL III	KNUTH D.E.	PARANINFO
3.3.3	Técnicas recursivas en programación	BROWN D.W.	KAPELUZ
3.3.4	PASCAL	DALE Nell y WEEMS	McGraw Hill
3.3.5	Programación PASCAL	GROGONO	Fondo educativo Latinoamericano