

UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
LICENCIATURA EN CIENCIAS INFORMÁTICAS
PLAN 2009

PROGRAMA DE ESTUDIOS

Resolución 15/03/07-00 Acta 926/16/02/2015 Anexo 05.2

I. - IDENTIFICACIÓN

1. Materia	: Algorítmica I
2. Semestre	: Primero
3. Horas semanales	: 5 horas
3.1. Clases teóricas	: 1 hora
3.2. Clases prácticas	: 4 horas
4. Total de horas disponibles	: 80 horas
4.1. Clases teóricas	: 16 horas
4.2. Clases prácticas	: 64 horas

II. - JUSTIFICACIÓN

La resolución de problemas utilizando una computadora es una de las tareas básicas y transversales en Informática. La misma requiere de la capacidad para analizar y entender el problema, diseñar y establecer una solución adecuada de acuerdo a las restricciones impuestas y realizar una especificación detallada de esa solución. Esta solución detallada podrá ser utilizada en un computador luego de ser transformada de acuerdo a reglas de algún lenguaje de programación.

III. - OBJETIVOS

1. Desarrollar el razonamiento lógico para encontrar soluciones a problemas pasibles a ser resueltos en computadoras.
2. Identificar los elementos necesarios para la representación de soluciones a problemas algorítmicos.
3. Elaborar soluciones a problemas algorítmicos utilizando técnicas de programación estructurada y modular.
4. Adquirir criterios de utilización de estructuras de información adecuadas de acuerdo al tipo de problema.

IV. - PRE - REQUISITO

1. No tiene.

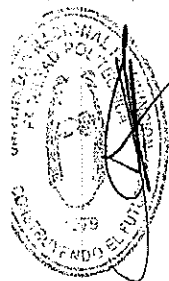
V. - CONTENIDO

5.1. Unidades programáticas

1. Introducción a la solución de problemas con computadoras y su representación.
2. Resolución de problemas algorítmicos.
3. Arreglos.
4. Cadenas.

5.2. Desarrollo de las unidades programáticas

1. Introducción a la resolución de problemas con computadoras y su representación.
 - 1.1. Análisis y elementos del problema.
 - 1.2. Concepto y propiedades de un algoritmo.
 - 1.3. Diseño de algoritmos.
 - 1.4. Representación de algoritmos.
 - 1.4.1. Diagramas de flujo.
 - 1.4.2. Seudocódigo.
 - 1.5. Verificación de los algoritmos. Prueba de escritorio.
2. Resolución de problemas algorítmicos.
 - 2.1. Concepto de programación estructurada y diseño top-down.
 - 2.2. Fundamentos de programación. Elementos básicos.
 - 2.2.1. Concepto de programa. Diferencia entre programa y algoritmo.
 - 2.2.2. Concepto de lenguaje de programación. Formas de traducción de algoritmos a lenguajes de programación.
 - 2.2.3. Sintaxis y semántica de un programa.
 - 2.2.4. Arquitectura de funcionamiento de un programa.
 - 2.2.5. Palabras reservadas.
 - 2.2.6. Identificadores. Reglas para crear identificadores.
 - 2.2.7. Tipo de datos simples: numérico, lógico y cadena.
 - 2.2.8. Constantes y variables.
 - 2.2.9. Operadores y expresiones. Evaluación de expresiones. Reglas de precedencia y asociatividad.
 - 2.2.10. Operación de asignación.
 - 2.2.11. Funciones internas.
 - 2.2.12. Operaciones de entrada y de salida.
 - 2.2.13. Estilo de codificación: identificadores, comentarios e indentación.
 - 2.3. Elementos de un algoritmo.



- 2.3.1. Sentencias secuenciales.
- 2.3.2. Sentencias selectivas.
 - 2.3.2.1. Simples.
 - 2.3.2.2. Múltiples.
- 2.3.3. Sentencias iterativas o repetitivas.
 - 2.3.3.1. Tipos y funcionalidades.
 - 2.3.3.2. Combinación con sentencias secuenciales y selectivas.
- 2.3.4. Subprogramas o subalgoritmos.
 - 2.3.4.1. Programación modular.
 - 2.3.4.2. Funcionamiento.
 - 2.3.4.3. Tipos.
 - 2.3.4.4. Ámbito de variables.
 - 2.3.4.5. Parámetros.
 - 2.3.4.5.1. Tipos.
 - 2.3.4.5.2. Paso de parámetros.
- 3. Arreglos.
 - 3.1.1. Unidimensionales o vectores.
 - 3.1.1.1. Definición e inicialización.
 - 3.1.1.2. Recorrido y procesamiento de elementos.
 - 3.1.1.3. Algoritmos de ordenamiento más conocidos: burbuja, inserción, selección.
 - 3.1.1.4. Vectores abiertos.
 - 3.1.1.5. Uso de vectores con subprogramas.
 - 3.1.2. Bidimensionales o matrices.
 - 3.1.2.1. Definición e inicialización.
 - 3.1.2.2. Recorrido y procesamiento de elementos.
 - 3.1.2.3. Uso combinado de matrices y vectores.
 - 3.1.2.4. Matrices abiertas.
 - 3.1.2.5. Uso de matrices con subprogramas.
- 4. Cadenas.
 - 4.1. Definición y utilidades.
 - 4.2. Juego de caracteres.
 - 4.3. Funciones predefinidas de tratamiento de cadenas.
 - 4.4. Algoritmos asociados con cadenas y vectores.

VI. - ESTRATEGIAS METODOLÓGICAS

1. Ciclo de Aprendizaje Vivencial (CAV).
2. Exposición y debate.
3. Trabajos grupales.

VII. - MEDIOS AUXILIARES

1. Tablero acrílico y pinceles.
2. Computador y proyector.
3. Herramienta de diseño de algoritmos.
4. Sala de máquinas para clases prácticas.

VIII. - EVALUACIÓN

1. Diagnóstica.
2. Formativa.
3. Sumativa.

IX. - BIBLIOGRAFÍA

1. Braunstein, Sílvia L. Introducción a la programación y a las estructuras de datos / Sílvia L. Braunstein, Alicia B. Gioia. -- Buenos Aires : Editorial Universitaria, 1987.
2. Joyanes Aguilar, Luis. Fundamentos de programación, algoritmos y estructuras de datos / Luis Joyanes Aguilar. -- Madrid : McGraw-Hill, 1990.
3. Rodríguez Almeida, Miguel Angel. Metodología de la programación a través de pseudocódigo / Miguel Angel Rodríguez Almeida. -- Madrid : McGraw-Hill, 1991.
4. Schriberm, Thomas J. Fundamentos de diagramas de flujo / Thomas J. Schriberm