

UNIVERSIDAD NACIONAL DE TRUJILLO
Facultad de Ciencias Físicas y Matemáticas
Escuela Académico Profesional de Informática

SÍLABO

I. IDENTIFICACIÓN

1. Experiencia curricular : Estructura de Datos
2. Para estudiantes de la carrera : Informática
3. Calendario Académico : 2009
4. Semestre Académico : II
5. Extensión Horaria :
- Total de horas semanales : 6 h (Para cada grupo: A y B)
- Teoría : 2 h
- Práctica : 2 h
- Laboratorio : 2 h
- Total de horas en el semestre : 96 h
6. Creditaje : 4
7. Organización del tiempo anual semestral :

Tipo de Actividades	Total Horas	Unidades		
		I	II	III
7.1. Clases de enseñanza – aprendizaje	86	28	28	30
7.2. Sesiones de evaluación sumativa	10	2	2	6
7.3. Tiempo de holgura (imprevistos)	0	0	0	0
Total Horas	96	30	30	36

8. Departamento Académico y Facultad : Departamento de Informática
 Facultad de Ciencias Físicas y Matemáticas
9. Docentes : Dr(c). José M. Saavedra Rondo
 Ing. Teresa Jacqueline Bracamonte Nole
10. Fecha de Inicio : 18 de enero de 2010
11. Fecha de Término : 15 de mayo del 2010

II. FUNDAMENTOS Y DESCRIPCIÓN

Estructuras de datos es un curso fundamental en la formación de todo estudiante de Ciencias de la Computación o Informática, pues en este curso se estudian diversas formas de organizar los datos de modo que su acceso sea altamente eficiente. La elección de una estructura de datos para implementar un método en cualquiera de las áreas de la Ciencia de la computación afecta su aplicabilidad. Por tal motivo, el informático debe conocer profundamente las estructuras de datos clásicas y modernas de modo que su decisión sea la mejor. En este curso, estudiaremos estructuras clásicas como Pilas y Colas basadas en manejo dinámico de memoria, así como árboles de búsqueda y

técnicas de Hashing. Así mismo, revisaremos algunas estructuras novedosas como Union-Find, Splay Trees y Árboles aleatorizados.

Los grafos son modelos matemáticos muy utilizados en la solución computacional de problemas. Por este motivo, en este curso revisaremos algunos algoritmos clásicos para manipular grafos. De igual modo, estudiaremos algunos algoritmos utilizados en la compresión de texto como los algoritmos de Huffman y los basados en Lempel-Ziv.

La eficiencia de las estructuras utilizadas es un factor crítico. El análisis asintótico de un algoritmo es la clave para determinar la eficiencia de los mismos. Así, en las primeras semanas se discutirán algunas técnicas para determinar la complejidad de los algoritmos.

III. APRENDIZAJES ESPERADOS

Al finalizar el curso el alumno debe ser capaz de:

- Entender la importancia del curso en la Ciencia de la Computación
- Comprender el fundamento teórico de las principales estructuras de datos
- Evaluar la eficiencia de algoritmos diseñados para resolver problemas específicos
- Evaluar la eficiencia de las operaciones sobre las estructuras de datos
- Implementar adecuadamente las principales estructuras de datos
- Resolver problemas utilizando las estructuras de datos más apropiadas
- Desarrollar interés en esta área para poder realizar posteriormente investigaciones en este campo
- Entender que un software es de calidad solamente cuando se analiza el respectivo algoritmo y se lo implementa con la estructura de datos más adecuada

IV. PROGRAMACIÓN

UNIDAD Nº 1

1. Denominación: *Análisis de Algoritmos y Estructuras de Datos Básicas*

2. Inicio: 18/01/10

Término: 20/02/10

Nº de semanas: 5 semanas

3. Objetivos de Aprendizaje:

- Aplicar estructuras estáticas en la solución de problemas.
- Analizar diferentes algoritmos de ordenación.
- Identificar, superficialmente, la importancia del análisis de un algoritmo.
- Aplicar la recursividad en la solución de problemas computacionales.

- Diferenciar la implementación estática y dinámica de las estructuras de datos.
- Según sea el problema a estudiar, identificar qué tipo de estructura de datos se debe usar.

4. Desarrollo de la enseñanza – aprendizaje:

Semana	Actividades y/o contenido	MMEE
1 18/01/10 – 23/01/10	TEORÍA y PRACTICA: Introducción a las estructuras de datos, tipo abstracto de datos. Repaso de Arreglos Unidimensionales y estructuras. Casos de Estudio. LABORATORIO: Implementar en C o C++ operaciones sobre arreglos unidimensionales.	Clases preparadas por el docente Libro texto Listado de Ejercicios Software de Lenguaje de Programación
2 25/01/10 – 30/01/10	TEORÍA Y PRACTICA: Repaso de Arreglos Multidimensionales. Casos de Estudio. Introducción al Análisis de Algoritmos. LABORATORIO: Implementar en C o C++ operaciones sobre arreglos multidimensionales.	
3 01/02/10 – 06/02/10	TEORÍA Y PRACTICA: Técnicas de análisis de algoritmos para procedimientos iterativos y recursivos. LABORATORIO: Práctica de Análisis de Algoritmos.	
4 08/02/10 – 13/02/10	TEORÍA Y PRACTICA: Algoritmos de Ordenación: Selección, Inserción, QuickSort, MergeSort LABORATORIO: Implementación de algoritmos de ordenación.	
5 15/02/10 – 20/02/10	TEORÍA Y PRACTICA: Manejo de Memoria. Heap y Stack. Manejo de Punteros. Listas Enlazadas simples y dobles. Operaciones sobre Listas. LABORATORIO: Implementación de listas.	

5. Evaluación sumativa del Aprendizaje:

Semana	Técnica	Instrumento
1 – 5	Valoración de desempeño en horas de Laboratorio	Escala de Valoración Listado de ejercicios
5	Tarea 1: Evaluación de Algoritmos de Ordenación	Escala de Valoración
5	Examen Parcial de la unidad: Prueba escrita	Escala de Valoración

UNIDAD Nº 2

1. Denominación: *Análisis de Métodos de Ordenación*

2. Inicio: 22/02/10

Término: 27/03/10

Nº de semanas: 5 semanas

3. Objetivos de Aprendizaje:

- Comprender y aplicar estructuras dinámicas en la solución de problemas.
- Comprender y aplicar los TAD Pila y Cola.
- Comprender y aplicar árboles de búsqueda.
- Comprender y aplicar colas de prioridad.
- Comprender y aplicar técnicas de hashing.

4. Desarrollo de la enseñanza – aprendizaje:

Semana	Actividades y/o Contenido	MMEE
6 22/02/10 –27/02/10	TEORÍA y PRÁCTICA: Pilas y Colas. Aplicaciones en problemas reales. LABORATORIO: Implementación de pilas y colas en C o C++.	Clases preparadas por el docente Libro texto Listado de Ejercicios Software de Lenguaje de Programación
7 01/03/10 –06/03/10	TEORÍA Y PRACTICA: Árboles Binarios. Notación Postfija y evaluación de expresiones aritméticas. Árboles de Búsqueda Binaria LABORATORIO: Implementar en C o C++ la estructuras para representar árboles.	
8 08/03/10 –13/03/10	TEORÍA Y PRACTICA: Árboles Balanceados AVL, ABB aleatorizados, Splay Tree LABORATORIO: Implementar en C o C++ la estructuras para representar árboles balanceados.	
9 15/03/10 –20/03/10	TEORÍA Y PRACTICA: Hashing LABORATORIO: Implementar en C o C++ métodos de Hashing.	
10 22/03/10 –27/03/10	TEORÍA Y PRACTICA: Colas de Prioridad: Heaps. Ordenación con Heap. LABORATORIO: Implementar en C o C++ colas de prioridad.	

5. Evaluación sumativa del Aprendizaje:

Semana	Técnica	Instrumento
6 – 10	Valoración de desempeño en horas de Laboratorio	Escala de Valoración Listado de ejercicios
10	Tarea 2: Evaluación de Árboles de búsqueda	Escala de Valoración
10	Examen Parcial de la unidad: Prueba escrita	Escala de Valoración

UNIDAD Nº 3

- 1. Denominación:** *Análisis de Métodos de búsqueda.*
- 2. Inicio:** 29/03/10 **Término:** 15/05/10 **Duración:** 6 semanas
- 3. Objetivos de Aprendizaje:**
 - Comprender y aplicar algoritmos para búsqueda en texto.
 - Comprender y aplicar algoritmos para procesamiento de grafos.
 - Comprender y aplicar algoritmos de compresión.
- 4. Desarrollo de la enseñanza – aprendizaje:**

Semana	Actividades y/o Contenido	MMEE11
11 29/03/10 – 03/04/10	TEORÍA y PRÁCTICA: Treaps, Árboles B, Árboles digitales LABORATORIO: Implementación de pilas y colas.	Clases preparadas por el docente Libro texto Listado de Ejercicios Software de Lenguaje de Programación
12 05/04/10 – 10/04/10	TEORÍA Y PRÁCTICA: Búsqueda en Texto: Boyer Moore, Knuth Morris Pratt. LABORATORIO: Implementar en C o C++ los algoritmos de búsqueda en texto.	
13 12/04/10 – 17/04/10	TEORÍA Y PRÁCTICA: Grafos: Algoritmo Dijkstra, Algoritmo Kruskal. Uso de la Estructura Union Find LABORATORIO: Implementar en C o C++ algoritmos para grafos.	
14 19/04/10 – 24/04/10	TEORÍA: Algoritmos de Compresión de Texto: Huffman, Lempel-Ziv. LABORATORIO: Implementación del algoritmo de compresión de Huffman.	
15 26/04/10 – 01/05/10	TEORÍA: Búsquedas basadas en índices: Suffix Array.	

- Se considera una semana de holgura por Semana Santa.

5. Evaluación sumativa del Aprendizaje:

Semana	Técnica	Instrumento
10 – 15	Valoración de desempeño en horas de Laboratorio	Escala de Valoración Listado de ejercicios
15	Tarea 3: Implementación de un intérprete de comandos aritméticos: MINIARIT	Escala de Valoración

15	Examen Parcial de la unidad: Prueba escrita	Escala de Valoración Lista de Cotejos
----	---	--

Semana	Técnica	Instrumento
16	Examen de Aplazados	Escala de Valoración

V. NORMAS DE EVALUACION

- La mínima nota promocional aprobatoria es 10.5; en una escala de calificación de 0 – 20
- Sólo la nota promocional es redondeada.
- La evaluación del curso comprenden una serie de valoraciones que se consideran en el cálculo de la nota de cada unidad. La nota de 1ra, 2da y 3ra unidad se define como

$$PU = \frac{NL + 2*EP + 1.5T}{4.5}$$

Donde:

NL: Valoración de desempeño en horas de Laboratorio.

EP: Examen Parcial de la unidad.

T: Nota de la Tarea.

- La nota promocional se define como

$$NP = \frac{PU1 + PU2 + PU3}{3}$$

- Los estudiantes que tengan una nota promocional desaprobatória y hayan cumplido con presentar los proyectos encomendados en cada unidad, rendirán un examen de aplazados.
 - **No se aceptarán presentaciones de laboratorio fuera de la fecha establecida.**
 - La evaluación de rezagados será tomada antes de evaluar la tercera unidad, habiendo solamente una única fecha de evaluación.
 - La asistencia tanto a clase teórica, práctica y de laboratorio es obligatoria. **Más del 30% de inasistencia inhabilitan al alumno a tener promedio final.**
 - Es deber del alumno, asistir a todas las evaluaciones y presentar todos los trabajos; en caso contrario su calificativo sera de cero (0) en la respectiva evaluación o trabajo, salvo justificaciones solicitadas por escrito ante las autoridades pertinentes.
 - El alumno solamente tendrá derecho a rezagar una evaluación parcial previa justificación (salud o viaje) ante las autoridades correspondientes.
- Importante:** La justificación deberá realizarse con documentos originales y durante los 2 días hábiles siguientes a la realización de la evaluación.

VI. CONSEJERIA

1. Propósito:

- Reforzar los contenidos vertidos durante las horas de clase.
- Proveer al alumno información extra acerca de los temas tratados.
- Aclarar las dificultades encontradas en el desarrollo de sus investigaciones.

2. Estrategia de prestación de servicios:

Las modalidades de asesoría son:

- Individual, a manera de resolución de consultas y análisis de resultados de evaluación.
- En grupos, brindando orientación durante sus investigaciones y desarrollo de prototipos.

3. Lugar y horario semanal para la consejería extra – clase:

La asesoría se realiza de acuerdo a previa coordinación con los alumnos que la soliciten, la que se realizará los días **Lunes de 11 a.m. – 1:00 p.m.** en la oficina Nro 13 de la Escuela de Informática.

VII. BIBLIOGRAFIA

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C* (Segunda Edición), 1997
2. Mark Allen Weiss, *Data Structures and Algorithm Analysis in Java* (Second Edition), 2007.
3. Duane A. Bailey, *Java Structures: Data Structures in Java for the Principled Programmer*, 2007.
4. Thomas H. Cormen, Charles E. Leiserson and Donald L. Rivest, *Introduction to Algorithms*, Second Edition, McGraw-Hill Book Company, Cambridge Massachusetts, 2001.
5. G. H. Gonnet e R. Baeza-Yates, *Handbook of Algorithms and Data Structures*, Addison-Wesley, 1991, (<http://www.dcc.uchile.cl/~rbaeza>).
6. Robert L. Kruse and Alexander J. Ryba, *Data Structures and Program Design in C++*, Prentice-Hall, 2000.
7. Robert Sedgewick, *Algorithms in Java*, Third Edition, Addison Wesley, 2002.
8. Nivio Ziviani, *Projeto de Algoritmos com Implementações em Pascal e C*, Editorial Pioneira Thompson, S.P. 2003.
9. Luis Joyanes Aguilar. *Fundamentos de Programación. Algoritmos, Estructuras de Datos y Objetos 4ta Ed.* McGrawHill, España, 2008.
10. Luis Joyanes e Ignacio Zahonero. *Programación en C: Metodología, Algoritmos y Estructura de Datos.* Ed. McGrawHill, 1ra Ed. Madrid, 2004.
11. Deitel y Deitel. *Cómo programar en C++*. Pearson Education. México, 2003.
12. Nell Dale. *C++ Plus Data Structures*. Jones and Bartlett Publishers, Third Edition. *Massachusetts, 2003*.
13. Parker, Alan. *Algorithms and Data Structures in C++*. CRC Press. Alemania, 1993.

Dr(c) José M. Saavedra Rondo

Profesor coordinador del Grupo A

Teresa J. Bracamonte Nole

Profesor coordinador del Grupo B

ADDENDA

Presentación

1. Docente

2. Fecha

3. Firma

Recepción

1. Nombres y Apellidos

2. Cargo

3. Fecha

4. Firma

Supervisión por Jefatura de Departamento Académico	
Logros	Sugerencias para la mejora

Jefe de Departamento: José G. Cruz Silva

Fecha:

Firma:
