

PROGRAMA DE POSTGRADO: COMPUTO APLICADO

CURSO: ESTRUCTURAS DE DATOS Y PROGRAMACIÓN

PROFESOR TITULAR: JUAN MANUEL GONZALEZ CAMACHO/ PAULINO PÉREZ RODRÍGUEZ

COLABORADOR (ES): \_\_\_\_\_

CORREO ELECTRÓNICO: [jmgc@colpos.mx](mailto:jmgc@colpos.mx); [perpdgo@colpos.mx](mailto:perpdgo@colpos.mx)

TELÉFONO: 58045900 EDIFICIO/PLANTA/NÚMERO EDUARDO CASAS DÍAS  
EXT. 1431 1ER PISO

CLAVE DEL CURSO: COA 618

PRE-REQUISITOS: \_\_\_\_\_

TIPO DE CURSO:

PERIODO:

- Teórico  
 Práctico  
 Teórico-Práctico

- Primavera  
 Verano  
 Otoño  
 No aplica

SE IMPARTE A :

MODALIDAD:

- Maestría en Ciencias  
 Doctorado en Ciencias  
 Maestría Tecnológica

- Presencial  
 No presencial  
 Mixto

HORAS CLASE:

CREDITOS: 3

Presenciales 48  
 Extra clase 144  
 Total 192

Nota: Un crédito equivale a 64 horas totales (presenciales y extra clases)

**OBJETIVO GENERAL DEL CURSO**

El alumno debe conocer las estructuras de datos básicas para elegir la más conveniente en cada situación en la que debe almacenar y organizar datos en memoria o en archivos de disco. Asimismo podrá elaborar los programas para implementar las estructuras, aplicando los algoritmos más eficientes para cada estructura. Comparará el desempeño de operaciones típicas como el ordenamiento y la combinación de datos usando diversos algoritmos y estructuras. Aprenderá a decidir si usará algoritmos iterativos o recursivos dependiendo de la situación. Manejará los conceptos básicos de programación orientada a objetos, especialmente el uso de clases cuando sean convenientes.

HORAS ESTIMADAS	TEMAS Y SUBTEMAS	OBJETIVOS DE LOS TEMAS
6	1. CONCEPTOS GENERALES DE ESTRUCTURAS Y ALGORITMOS.	LOS ALUMNOS COMPRENDERÁN LA NECESIDAD DE USAR ESTRUCTURAS, SU UTILIDAD Y EL MOTIVO DE INTRODUCIR DIVERSAS ESTRUCTURAS.
6	2. PROGRAMACIÓN DE ALGORITMOS.	ALGORITMOS, FORMULACIÓN Y PROGRAMACIÓN. ESTIMACIONES DE DURACIÓN Y EFICIENCIA.
9	3. ORDENAMIENTO DE DATOS. COMBINACIONES DE LISTAS CON LAS OPERACIONES BOOLEANAS.	SE INTRODUCEN LISTAS, LISTAS ENLAZADAS, DOBLEMENTE ENLAZADAS, CIRCULARES, COLAS Y PILAS. BÚSQUEDAS BINARIAS Y SU APLICACIÓN. IMPLEMENTACIÓN DE LISTAS EN ARREGLOS Y CON MEMORIA OBTENIDA PARA CADA ELEMENTO.
6	USO DE COLAS PEPS Y UEPS (FIFO Y LIFO). USO DE PILAS.	EL ALUMNO APRENDE A DECIR LA ESTRUCTURA QUE CONVIENE A CADA SITUACIÓN.
3	4. ARCHIVOS EN DISCO	ARCHIVOS PLANOS, ACCESO ALEATORIO, ARCHIVOS TIPO EDICIÓN.
6	5. EL CONCEPTO DE ÁRBOL. ÁRBOLES BINARIOS. CONCEPTO E IMPLEMENTACIÓN DE BALANCEO DE ÁRBOLES.	EL ALUMNO APRENDERÁ A USAR UN ÁRBOL BINARIO. IMPLEMENTARÁ LOS ÁRBOLES E INCLUIRÁ CRITERIOS DE BALANCEO.
6	6. ÁRBOLES B Y B+. ÁRBOLES DE BÚSQUEDAS.	EL ALUMNO ELABORARÁ PROGRAMAS PARA IMPLEMENTAR ESTAS ESTRUCTURAS CON LOS CRITERIOS DE APROVECHAMIENTO DE ESPACIO Y BALANCEO CONVENIENTES.
6	7. OTRAS ESTRUCTURAS	MAPAS DE BITS (BIT-MAPS). COMBINACIONES DE ESTRUCTURAS (ESTRUCTURAS HÍBRIDAS).

---

#### LISTA DE PRÁCTICAS

---

El curso es teórico en cuanto a las estructuras mismas, pero práctico en cuanto a su uso. La programación de los algoritmos constituye una parte fundamental del curso. La valoración de la eficiencia de los algoritmos se enfatiza para que el alumno aprenda a tomar en cuenta este aspecto cuando use las estructuras. Se enfatiza el uso de lógica recursiva.

---

#### RECURSOS DIDÁCTICOS

---

Los alumnos deben contar con una computadora. El lenguaje utilizado en el curso es Visual Basic, pero cada profesor podrá modificar este criterio.

Las clases se imparten con proyector y otros dispositivos que use el profesor en cada caso.

Los alumnos no utilizan sus computadoras durante las clases teóricas, pero hay sesiones pactadas precisamente para que el profesor vea el trabajo en vivo de los alumnos, y para aclaración de dudas que puedan tener los alumnos.

---

---

## NORMAS Y PROCEDIMIENTOS DE EVALUACIÓN

---

### Normas de evaluación

Al alumno se lo califica de acuerdo al grado de asimilación de las estructuras, las diferencias entre ellas en cuanto a cuando conviene utilizar cada una de ellas. Un elemento importante es la habilidad para elaborar programas para cada una de las estructuras y de situaciones en las que se utilizan.

La evaluación toma en cuenta la diversidad de conocimientos técnicos que tienen los alumnos, que puede ser amplia. Sin embargo, si un alumno no está en condiciones de efectuar alguna de las tareas o tiene dificultades con los conceptos derivadas de falta de conocimientos previos, se auxiliará al alumno para poder eliminar las deficiencias o dificultades.

### Procedimiento de evaluación

La calificación final se basa en “incrementos de habilidades y conocimientos”. Dada la diversidad de conocimientos técnicos, no es posible comparar resultados solamente. Se califican individualmente las tareas, incluyendo la entrega puntual de las mismas. Se estimula y califica la creatividad y el deseo de mejorar del alumno.

Se aplican dos exámenes: uno permite determinar las habilidades en cuanto a la programación de algoritmos para ordenamientos y combinación de listas, diversos tipos de listas, colas y pilas, y búsquedas binarias en listas ordenadas.

El segundo es sobre árboles binarios y árboles B y B+.

Si el profesor lo considera conveniente, asignará un proyecto final especialmente con el fin de permitir a los alumnos con calificaciones bajas en los exámenes a mejorar sus calificaciones. Este proyecto se califica tomando en cuenta precisamente el desarrollo del alumno, comparando los resultados con su desempeño anterior (y no necesariamente el de los otros alumnos del curso).

La calificación final no necesariamente se obtendrá con una combinación lineal de los resultados de los exámenes, tareas y el proyecto final. Si el profesor lo considera conveniente, podrá hacer un examen individual (en general oral, pero cuando es conveniente, con algunas componentes concretas). Se discute con el alumno el progreso que determinó el profesor y el propio alumno en cuanto a los que asimiló y cómo define un sistema y sus componentes.

---

BIBLIOGRAFÍA IMPRESA O ELECTRÓNICA (AUTOR, AÑO, TÍTULO, EDITORIAL, FECHA, EDICIÓN)

---

*Okasaki, C.. 1998. Purely Functional Data Structures. The Press syndicate of the University of Cambridge. UK. 1998.*

*Weiss, P.A. 1997 Data structures and algorithm analysis in C. 2nd edition. Addison Wesley, 1997*

*Weiss, P.A. 2010 Data structures and algorithm analysis in Java. 4th edition. Addison Wesley, 2010-*

*Comer, D. 1979- "The Ubiquitous B-Tree", Computing Surveys 11 (2): 123–137. 1979*

*Cormen, T; Leiserson, Charles; Rivest, Ronald; Stein, Clifford (2001). Introduction to Algorithms (Second ed.), MIT Press and McGraw-Hill, pp. 434–454, ISBN 0-262-03293-7. Chapter 18: B-Trees. 3<sup>rd</sup> Printing, 2002.*

*Folk, Michael J.; Zoellick, Bill (1992), File Structures (2nd ed.), Addison-Wesley,1992.*

*Knuth, D. (1998), Sorting and Searching, The Art of Computer Programming, Volume 3 (Second ed.), Addison-Wesley,1998.*

*Bayer, R. 1971. "Binary B-Trees for Virtual Memory", Proceedings of 1971 ACM-SIGFIDET Workshop on Data Description, Access and Control, San Diego, California. November 11–12, 1971.*

---