



UNIVERSIDAD PRIVADA “ANTENOR ORREGO”

FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL DE INGENIERÍA DE SOFTWARE

SÍLABO – SEMESTRE 2010-2

I. DATOS INFORMATIVOS

1.1 Asignatura	Estructura de Datos y de la Información
1.2 Código	INSO - 105
1.3 Ciclo de estudios	III
1.4 Créditos	4
1.5 Total de horas semestrales	96
1.6 N° de horas por semana	Teoría: 2 Laboratorio: 4 Total: 6 horas
1.7 Fecha de inicio	16 de Agosto de 2010
1.8 Fecha de terminación	4 de Diciembre de 2010
1.9 Duración	16 semanas (14 dictadas + 2 Exámenes)
1.10 Prerrequisitos	INSO - 103
1.11 Profesores	Mag. Eduardo F. Alonso Pérez (coordinador) ealonsop@upao.edu.pe Ing. Carla Y. Yupanqui Rodríguez yupanqui_carla@hotmail.com

II. FUNDAMENTACIÓN

Las estructuras y algoritmos son las principales herramientas para construir un programa, por lo cual es importante disponer del conjunto más completo posible de ellas. Si el lenguaje no los tiene predefinidos, es necesario desarrollarlos manualmente con los medios disponibles. A través de esta asignatura se estudiarán los tipos de datos más utilizados en programación desde un enfoque aplicativo y abstracto. Se hace énfasis en el concepto de TDA mediante el cual cada tipo de dato es visto como un mecanismo de representación de información y el conjunto de operaciones que se le pueden aplicar.

Este tema es considerado por las Asociaciones Internacionales como la IEEE y la ACM, como una de las 9 áreas de interés en un currículo de Computación.

III. SUMILLA

La asignatura de Estructura de Datos y de la Información corresponde a las asignaturas del área de “Fundamento Especializado Tecnológico” y es de naturaleza teórico-practico. Permite a los estudiantes de la carrera profesional de Ingeniería de Software complementar su formación en programación con aspectos que le servirán para la resolución de problemas de mediana complejidad, aplicando para ello diversas estructuras de datos. Se estudian los tipos de datos fundamentales, su representación, implementación y aplicaciones (listas, pilas, colas, árboles y grafos)

IV. COMPETENCIAS

- Modela la realidad a través de tipos de datos que se ajusten al problema.
- Diseña y programa en lenguaje Java, desde un enfoque orientado a objetos, los tipos de datos más utilizados (Listas, Pilas, Colas, Árboles y Grafos), eligiendo para ello diversas formas de representación de la información e implementación de las operaciones.
- Desarrolla aplicaciones de mediana complejidad utilizando los tipos de datos estudiados.

Actitudes

- Muestra una actitud analítica y reflexiva frente a situaciones problemáticas de estructuras de datos.
- Muestra disposición al trabajo en equipo y responsabilidad en la resolución de los ejercicios.

- Muestra interés en conocer las aplicaciones de las listas, pilas, colas y árboles en general.
- Presenta entusiasmo e interés en comprender los grafos así como aplicaciones de los mismos para resolver problemas reales.
- Reconoce el aporte valioso de las estructuras en el desarrollo de productos de software.

V. PROGRAMACIÓN POR UNIDADES DE APRENDIZAJE

UNIDAD 01. TIPOS DE DATOS ABSTRACTOS

Capacidades a desarrollarse

Comprende la evolución que ha tenido el concepto de Tipo de Datos desde los primeros lenguajes de programación hasta la actualidad.

Descubre la importancia que tienen los tipos de datos en la programación para la solución eficiente de problemas.

Comprende la importancia del concepto de Tipo de Dato Abstracto (TDA).

Desarrolla programas aplicando tipos de datos propios del lenguaje Java.

Duración: 1 semana (1)

N° de semana	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	CONTENIDOS ACTITUDINALES
1 (21/ago)	Evolución de los TDA. TDA y la POO. Ejemplo Motivador.	Desarrolla programas aplicando los tipos de datos propios del lenguaje Java	Comprende la importancia de los TDA para la programación

UNIDAD 02. TIPOS DE DATOS LINEALES (LISTAS)

Capacidades a desarrollarse

Comprende la utilidad de las LISTAS para la representación de colecciones de datos organizados linealmente.

Diseña e implementa el tipo de datos LISTA utilizando diversas estructuras de datos del lenguaje Java.

Aplica el tipo de datos LISTA para definir otros Tipos de Datos, restringiendo el acceso a los elementos.

Diseña e implementa los tipos de datos PILA y COLA a partir del tipo de datos LISTA.

Desarrolla programas utilizando los tipos estudiados

Duración: 5 semanas (de la 2 a la 6)

N° de semana	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	CONTENIDOS ACTITUDINALES
2 (28/ago)	Introducción al TDA Lista. Representación e implementación mediante Arreglos	Diseña e implementa el TDA Lista mediante Arreglos. Construye programas en JAVA que utilizan Listas	Comprende la utilidad de las listas para representar colecciones de datos. Comprende las ventajas y desventajas de los arreglos para implementar listas.
3 (4/sept)	TDA Lista. Representación e implementación mediante Nodos Simplemente Enlazados	Diseña e implementa el TDA Lista mediante Nodos Simplemente Enlazados. Construye programas en JAVA que utilizan Listas	Comprende la utilidad de las listas para representar colecciones de datos. Comprende las ventajas y desventajas de los nodos enlazados para implementar listas.
4 (11/sept)	Mejoras a la implementación de Listas con Nodos Enlazados	Diseña e implementa el TDA Lista mediante Nodos Simplemente Enlazados con algunas modificaciones que mejoran la eficiencia de	Comprende la utilidad de las listas para representar colecciones de datos. Comprende las ventajas y desventajas de los

		la implementación Construye programas en JAVA que utilizan Listas	nodos enlazados para implementar listas.
5 (18/sept)	Listas con restricciones de acceso. TDA Pila y Cola. Aplicaciones.	Diseña e implementa el TDA Pila y el TDA Cola como aplicaciones de Listas. Construye programas en JAVA que utilizan Pilas y Cola.	Comprende la utilidad de las Pilas y las Colas para representar y utilizar colecciones de datos con accesos restringidos Comprende la utilidad de las listas para implementar otros tipos de datos
6 (25/sept)	Aplicaciones. Listas Ordenadas. Colas de Prioridad	Diseña e implementa otros TDA a partir de los ya conocidos Construye programas en JAVA que utilizan los TDA Listas, Pilas y Colas.	Comprende la utilidad de los TDA para implementar otros tipos de datos

UNIDAD 03. TIPOS DE DATOS JERÁRQUICOS (ÁRBOLES)

Capacidades a desarrollarse

Comprende la utilidad del tipo de datos ARBOL para representar información organizada jerárquicamente.

Diseña e implementa en lenguaje Java el tipo de datos ARBOL.

Diseña e implementa Árboles Binarios y Árboles Binarios de Búsqueda (ABB)

Desarrolla aplicaciones utilizando Árboles.

Duración: 5 semanas (de la 7 a la 12*)

N° de semana	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	CONTENIDOS ACTITUDINALES
7 (2/oct)	Introducción al TDA Árbol. Recorridos. Recursividad. Representación e implementación mediante Listas	Diseña e implementa el TDA Árbol mediante Listas. Construye programas en JAVA que utilizan Árboles	Comprende la utilidad de los Árboles para representar y utilizar colecciones de datos organizados jerárquicamente Comprende la utilidad de las listas para implementar otros tipos de datos. Reconoce la importancia de la recursividad como una técnica de programación en la implementación y uso de árboles
8 (9/oct)	Aplicaciones	Construye programas en JAVA que utilizan Árboles	Comprende la utilidad de los Árboles para representar y utilizar colecciones de datos organizados jerárquicamente
10* (23/oct)	Árboles Binarios. Representación e implementación	Diseña e implementa el TDA Árbol Binario Construye programas que utilizan árboles binarios	Comprenda la utilidad de los árboles binarios como caso particular de los árboles
11 (30/oct)	Árboles Binarios de Búsqueda	Diseña e implementa el TDA Árbol Binario de	Comprende la utilidad de los árboles binarios para

		Búsqueda. Construye programas que utilizan árboles binarios de búsqueda.	mejorar la búsqueda de información en árboles binarios
12 (6/nov)	Aplicaciones. Conjuntos	Diseña e Implementa el TDA Conjunto como una aplicación de Árboles Binarios de Búsqueda	Comprende la utilidad de los conjuntos para representar información no repetida.

(*) Semana 9. Examen Parcial

UNIDAD 04. GRAFOS

Capacidades a desarrollarse

Comprende la utilidad del tipo de datos GRAFO para representar información relacionada.

Diseña e implementa en lenguaje Java el tipo de datos GRAFO.

Desarrolla aplicaciones utilizando Grafos.

Implementa los algoritmos de Ordenamiento Topológico y Búsqueda del camino más corto (Algoritmo de Dijkstra)

Duración: 3 semanas (de la 13 a la 15**)

N° de semana	CONTENIDOS CONCEPTUALES	CONTENIDOS PROCEDIMENTALES	CONTENIDOS ACTITUDINALES
13 (13/nov)	Introducción a los grafos. Representación mediante Matrices y Listas de Adyacencia. Implementación.	Diseña e implementa el tipo de datos Grafo. Construye programas en Java que utilizan grafos	Comprende la utilidad de los Grafos para representar y utilizar colecciones de datos relacionados (conectados) Comprende la utilidad de las listas para implementar otros tipos de datos. Reconoce las ventajas y desventajas de las diferentes implementaciones de grafos
14 (20/nov)	Grafos. Aplicaciones: -Ordenamiento Topológico -Costo Mínimo (Algoritmo de Dijkstra) -Árbol de expansión de costo mínimo (Algoritmo de Kruskal)	Implementa el algoritmo de ordenamiento topológico en Java, utilizando grafos, así como los algoritmos de costo mínimo y árbol de expansión mínimo.	Reconoce la utilidad de los grafos para resolver el problemas
15 (27/nov)	Aplicación Integradora de los tipos de datos estudiados	Implementa el algoritmo de búsqueda ciega que utiliza Listas, Pilas, Colas, Árboles y Grafos.	Reconoce la utilidad de los tipos de datos estudiados en el curso

(**) Semana 16. Examen Final

VI. ESTRATEGIAS METODOLÓGICAS

Esta asignatura se desarrollará en base dos aspectos:

Teórico:

a) Exposición en clases de los contenidos temáticos de cada unidad y la importancia de estos en la resolución de problemas computacionales.

b) Lecturas complementarias para ampliar los temas dados en las clases teóricas.

Práctico (Laboratorio):

- a) Resolución de ejercicios tipos por el profesor de laboratorio.
- b) Resolución de ejercicios parcialmente resueltos.
- c) Resolución de ejercicios propuestos.

VII. MATERIALES EDUCATIVOS Y OTROS RECURSOS DIDÁCTICOS

Los materiales que se emplearán serán los siguientes:

- a) Materiales educativos interactivos
Materiales impresos: texto guía, material de clases, hojas de prácticas de salón y laboratorio.
Direcciones electrónicas para recabar información sobre contenidos planteados.
- b) Materiales educativos para la exposición
Pizarra acrílica, plumones, mota, transparencia, proyector multimedia.

VIII. INDICADORES, TÉCNICAS E INSTRUMENTOS DE EVALUACIÓN

- La evaluación del curso comprende dos exámenes: Parciales (EP) y otro Final (EF)
- Las actividades prácticas de laboratorio serán desarrolladas en forma conjunta por el profesor y el estudiante. Se llevará a cabo una práctica calificada de laboratorio al término de cada unidad (PCL). La solución de los ejercicios propuestos en cada laboratorio acumula puntaje para el PCL.
- La nota promocional (NP) se define del modo siguiente:
$$NP = (EP + EF + 2 * PCL) / 4$$
 y será un valor entero que se obtiene de aproximar al entero superior las notas de parte fraccionaria ≥ 0.5 y al entero inferior al resto de las notas.
- La nota de calificación está en la escala de 0 a 20.
- El alumno aprueba el curso si $NP \geq 11$
- Habrá un examen de aplazados para los estudiantes que estén desaprobados y cuya nota promocional sea mayor ó igual a 07.
- El alumno que falte a cualquier evaluación tendrá cero (00) de nota.
- Este sistema de evaluación no permite la existencia de exámenes rezagados por ningún motivo, salvo justificación solicitada por escrito ante las autoridades respectivas.
- La asistencia a clases teoría y de laboratorio es obligatoria, más de 30% de inasistencia inhabilita al alumno a tener promedio final (IN).

IX. PROGRAMA DE TUTORÍA Y CONSEJERÍA

Consiste en orientar y ayudar a los alumnos a resolver problemas académicos y personales con el aporte del docente debido a su experiencia; por lo tanto se tendrá en cuenta lo siguiente:

- a) Todas las dificultades que se susciben en el desarrollo de la asignatura en forma individual o grupal serán consultadas con el docente.
- b) Se concentrará los días de consulta y de entrevista de tal manera que se elaborara un horario concordante con la disponibilidad horaria con que cuenta el docente.
- c) El docente tendrá la responsabilidad de dar una orientación permanente y continua a los estudiantes para motivarlos y estimularlos a través de herramientas pedagógicas a fin de lograr los aprendizajes deseados.

X. BIBLIOGRAFÍA

- Aho Alfred V., Hopcroft John E. y Ullman Jeffrey D. Estructuras de datos y algoritmos. Addison-Wesley Iberoamericana, 1988
- Allen Weiss, Mark. Estructuras de datos en Java. Compatible con Java 2 Addison-Wesley Iberoamericana. 2000.
- Alonso Pérez, Eduardo. Estructura de Datos: Un enfoque aplicativo. Chiclayo. Perú: <http://usuarios.lycos.es/ealonsop/eda.html>
- Berman, A. Michael. Data structures via C++: Objects by evolution. Oxford University Press.1997

- Joyanes Aguilar, Luis; Zahonero Martínez, Ignacio. Programación en java 2: algoritmos, estructuras de datos y programación orientada a objetos. Mc Graw-Hill Interamericana. 2002.
- Langsam, Yedidyah; Augenstein, Moshe J. Y Tenenbaum, Aaron M. Estructuras de datos con C y C++. Prentice Hall Hispanoamericana, 1997.
- Michael Main / Walter Savitch. Data Structures And Other Objects. 1998
- Sedgewick, Robert. Algoritmos en C++. Addison-Wesley/Diaz de Santos.1995
- S. Sahní. Data Structures Algorithms and Applications in Java. McGraw Hill. 2000.
- Villalobos S., Jorge A. Diseño y Manejo de estructuras de datos en C. Mc-Graw Hill. Interamericana,S.A. 1996.