

FORMULARIO PARA LA PRESENTACION DE PROGRAMAS ANALITICOS Y DE EXAMEN

1. IDENTIFICACION

1.1. FACULTAD DE CIENCIAS EXACTAS Y NATURALES Y AGRIMENSURA

1.2. DEPARTAMENTO: **Informática.**

1.3. AREA: **Programación**

1.4. ASIGNATURA: **Algoritmos y Estructuras de Datos II**

1.5. CARRERAS: **Licenciatura en Sistemas de Información (Plan 2009)**

1.6. Año en que se dicta: **1° - 2do Cuatrimestre**

1.7. PROFESOR RESPONSABLE:

Apellido y Nombres: **VALLEJOS, OSCAR ADOLFO. Prof. Adjunto (Ord) A/C.**

Máximo Título Alcanzado: **Magister en Informática y Computación**

1.8. MODALIDAD: **Cuatrimestral**

1.9. CARGA HORARIA TOTAL: **128 (8 horas semanales)**

1.10. CARGA HORARIA SEMANAL TEORICA/PRACTICA: **8 (2.30 teóricas, 4 prácticas en aula y 1.30 practica en laboratorio)**

1.11. Obligatoriedad y optividad. **Obligatoria.**

2. DESCRIPCION

Fundamentación

La ciencia sobre el estudio de algoritmos incluye a diferentes e importantes áreas de investigación y docencia y considera la construcción, la expresión, la validación, el análisis y el testeo de programas. Conforme crece el parque de ordenadores, también aumentan sus prestaciones, y los cálculos más difíciles de efectuar se convierten en rutinas y como consecuencia de ello es necesario abordar el estudio de esta ciencia, habida cuenta de los prodigiosos adelantos, como por ejemplo, en el aspecto de velocidad de cálculos de los ordenadores más convencionales.

La codificación en un lenguaje de un programa de cierta complejidad se debe realizar efectuando oportunas descripciones que especifiquen de manera clara y concreta los datos que intervienen y el tratamiento de los mismos para el logro de los objetivos, logrando de esta manera, diseños de algoritmos eficientes.

Se enmarcan, a continuación, una serie de conceptos para esta asignatura en continuidad con la Asignatura Algoritmos y Estructuras de Datos I, ambas del primer año de la carrera Licenciatura en Sistemas de Información:

1. Criterios que sirven de sustento a la conceptualización y práctica que el alumno incorporara (definiciones, terminología, etc.)
2. Técnicas orientadas a la resolución de problemas utilizando estructuras de datos compuestas y enlazadas.
3. La eficiencia en la utilización de los recursos tiempo y memoria seleccionando estructuras de datos y su tratamiento utilizando herramientas poderosas que dan soluciones claras a problemas complejos. (Ej.: arreglos, pilas, colas, ordenamiento, búsqueda, recursividad).
4. Extender el concepto de tipo de dato definido por el usuario como una caracterización de elementos del mundo real, tendiendo al encapsulamiento de la representación y al comportamiento dentro de un tipo abstracto de datos.
5. Profundizar conceptos relacionados con el desarrollo y la evaluación de los algoritmos, precisando las diferentes implementaciones que resuelven correctamente un problema, que pueden ser muy distintos en cuanto a su eficiencia, adentrándonos de esta manera a la Complejidad Computacional.
6. Operaciones con estructuras de datos de almacenamiento.
7. Incentivar el uso de procesos confiables para el desarrollo de programas, las prácticas comunes y el análisis de cómo evolucionan.

Metodología

Los contenidos previstos se impartirán a través del dictado de clases teóricas, prácticas y de laboratorio.

Las primeras brindarán los conceptos necesarios que serán integrados con las actividades prácticas, con el propósito que los alumnos consoliden el pensamiento lógico y la capacidad de abstracción que se requiere para resolver problemas complejos utilizando distintos tipos de estructuras de los datos, incluyendo técnicas para optimización de algoritmos.

En las clases, el docente actuará como facilitador, atendiendo las consultas de los alumnos y estimulándolo al desarrollo de actividades de estudio independiente.

El objetivo de la asignatura es llegar a que el alumno pueda especificar el algoritmo para la solución de un problema complejo, su representación utilizando algunas herramientas vistas en el curso y posterior codificación mediante un lenguaje de programación estructurado. Finalmente, comprobar, depurar y probar la solución propuesta.

A lo largo del dictado de la asignatura se utilizarán estrategias e instrumentos de evaluación que permitan realizar un seguimiento al proceso de aprendizaje individual y grupal.

Durante las clases se observará sistemáticamente el avance de los alumnos, fundamentalmente, sus producciones y animándolos a participar activamente en el desarrollo de los ejercicios.

Los criterios de evaluación son los siguientes:

- capacidad para resolver problemas utilizando las técnicas y herramientas vistas en el curso;
- capacidad de abstracción y de modelización;
- utilización de los términos técnicos específicos;
- tratamiento lógico;
- relación de los distintos conceptos.

Se promueve la participación y acompañamiento del alumno a través de la plataforma educativa virtual. Dicha plataforma se presenta como complemento del proceso de aprendizaje y sirve de apoyo a las actividades de los docentes de la asignatura, como así también, se encuentra a disposición de los alumnos la planificación de las actividades, programa analítico, material de consulta, etc.

Como parte de las estrategias de enseñanza y aprendizaje, los docentes de cada una de las comisiones de la asignatura llevan un registro de desempeño de cada alumno a los efectos de la verificación y seguimiento del proceso de aprendizaje.

Articulación con otras asignaturas

Este espacio curricular profundiza los conceptos iniciados en la asignatura Algoritmos y Estructuras de Datos I, en las que se imparten las estructuras de datos simples y la manera de manipular las mismas, enmarcado en el paradigma procedural.

Posteriormente, las estructuras de datos compuestas presentadas en la asignatura actual y la consolidación del pensamiento lógico, son componentes necesarios para el proceso de aprendizaje y constituyen la base sobre la que se construyen los contenidos de las siguientes asignaturas del área de Programación.

Articulación Vertical:

El tema 1 articula con la asignatura Algoritmo y estructuras de datos I.

Los temas del 2 al 9 se desarrollan sustentados en los conceptos ya visto por el alumno en Algoritmo y Estructura de Datos I.

La complementariedad adquirida en estos temas es el sustento a los problemas que serán presentados en las asignaturas posteriores del plan de estudio, que desarrollan distintos paradigmas de programación (Taller de Programación I y Taller de Programación II, Programación Orientada a Objetos).

Específicamente, en el tema 5 se desarrolla el concepto de abstracción y modelización de objetos del mundo real para definir estructuras de datos abstractos, siendo esta temática abordada y profundizada posteriormente por el alumno en la asignatura Programación Orientada a Objetos.

Articulación Horizontal:

Los temas 8 y 9 articulan con la asignatura Lógica y Matemática Computacional, específicamente en los temas de recorrido de grafos y Eficiencia Algorítmica en la notación asintótica y ecuaciones de recurrencia.

2.1. OBJETIVO(S) GENERALES:

Se pretende que el alumno logre:

1. Modelar la resolución de problemas dividiéndolos en módulos.
2. Adquiera el razonamiento lógico necesario para la resolución de problemas en el contexto del paradigma procedural o programación estructurada.
3. Construya algoritmos, utilizando las siguientes técnicas para la resolución de problemas:
 - a. Estructuras de control básicas (secuencia, decisión, selección e iteración) y tipos de datos simples.
 - b. Estructuras de datos compuestas (arreglos, archivos, pilas, colas, listas, grafos).
 - c. Estructuras de datos abstractos (TAD)
 - d. Estrategias de desarrollo de algoritmos (recursividad, optimización, eficiencia, etc.)
4. Analizar, depurar y evaluar algoritmos.
5. Adquiera los conocimientos introductorios a la programación orientados a objetos.
6. Mejorar, a través de la lectura, el vocabulario, ortografía y expresión oral y escrita de los alumnos.

2.2. TIPO/S DE ACTIVIDAD/ES: (marque con una cruz)

Clases Teóricas y Clases Prácticas (en aula y laboratorio)

2.2.1. Técnicas o Estrategias didácticas:

Clases de Teoría

Se desarrolla a través de clases expositivas del profesor, fomentando la lectura previa de materiales y libros de textos puestos a disposición del alumnado.

En dicha exposición se fomenta en los alumnos una actitud reflexiva y crítica a través de la lectura de temas propios de la asignatura.

Desde este espacio, e interactuando con las clases prácticas, el alumno reflexiona y descubre las relaciones entre los diversos conceptos formándose una mentalidad como afrontar los problemas y proponer un método para la solución de los mismos.

Clase Práctica (aula y laboratorio)

La resolución de ejercicios de tipo práctico, para afianzar los conceptos teóricos, es realizada por el alumno en forma grupal y/o en forma individual y se lo estimula para adquirir capacidad de la resolución de problemas. Teniendo en cuenta la génesis del tipo de asignatura, lo ideal es fomentar el tratamiento lógico de los problemas por cada uno de los alumnos, por lo que habitualmente la resolución de los problemas si bien pueden ser encarados grupalmente, una vez entendidos el mismo se hace hincapié para que la solución sea aportada utilizando el intelecto individual.

Semanalmente se realizan clases de laboratorio en las que los alumnos codifican en un lenguaje de programación las soluciones por ellos propuestas. A partir de dicha codificación y posterior ejecución, pueden verificar y validar su funcionamiento.

El objetivo de las clases prácticas (en aula y laboratorio) es consolidar en el alumno destrezas que le permitan presentar alternativas válidas de solución a los problemas que se plantean.

En cada clase se realiza un breve introducción de los conceptos involucrados (conceptos ya vistos en las clases de teoría) necesario para la resolución de los problemas que se presentan. Luego los alumnos resuelven las guías de ejercicios.

El equipo docente acompaña en el proceso de aprendizaje de los alumnos, aclarando las dudas y resolviendo las dificultades.

A lo largo del desarrollo de las clases de práctica los ejercicios de las guías son resueltos en la pizarra y/o en el ordenador por los alumnos. Esto propicia una verificación grupal de los mismos y la correspondiente unificación de criterios en la solución, presentada por algún docente del equipo.

Aula Virtual

El aula virtual, permite la interacción entre todos los participantes (docentes y alumnos). Se pueden exponer criterios, intercambiar experiencias, trabajar de forma colaborativa, opinar sobre las evaluaciones, tener acceso a recursos de diferentes tipos.

Este espacio, implementado como apoyo del proceso de aprendizaje, propende a reforzar la colaboración de cada uno de los alumnos. El alumno aporta ideas desde su proceso de aprendizaje y estas son consideradas por el equipo docente para reformular y retroalimentar los métodos,

herramientas y actualizaciones de contenidos a los efectos de que sea aprovechada por toda la comunidad virtual del presente cursado y de los posteriores.

Se intenta que el conocimiento se expanda a través de las conexiones que nos permite el aula virtual y en el que el aprendizaje consiste en la habilidad de construir ese conocimiento por el alumno mismo.

El alumno encontrará material de estudio, como también, series de trabajos prácticos, la codificación en lenguajes de programación de una gama de ejercicios prácticos, bibliografía, sitios en la red relacionados con algoritmos, etc., que le permitirá expandir en forma muy completa su visión en la aplicación de algoritmos de distintos tipos de complejidad.

También se le brinda información adicional, como ser, guías de trabajos prácticos adicionales, con una amplia variedad de ejercicios, con el objeto de estimular la capacidad de resolución de problemas.

Se estimulará el uso de la biblioteca para la búsqueda de información y aprendizaje.

2.3. REGIMEN DE PROMOCION:

2.3.1 Condiciones para promocionar la materia:

Los alumnos que cubran los requerimientos que se detallan más abajo podrán acceder a un tercer parcial de los contenidos teóricos de la asignatura.

Los alumnos que aprueben este tercer parcial promocionaran la asignatura. La nota de promoción será aquella que resulte de promediar la obtenida en la instancia práctica (promedio de las notas obtenidas en los parciales) y la obtenida en el tercer parcial.

Este tercer parcial no tiene instancia de recuperación. Es decir, aquellos alumnos que no aprueben el tercer parcial quedaran en condiciones de REGULARES en la asignatura.

Requerimientos para acceder al tercer parcial teórico:

- 75%, o más, de asistencia a las clases de teoría
- haber regularizado la materia con nota promedio mayor o igual a 7:
 - 75% de asistencia a las clases prácticas (aula y laboratorio)
 - Aprobación de los exámenes parciales (primer y segundo parcial) con nota promedio mayor o igual a 7. En cualquier instancia (parciales, recuperatorios, extraordinarios). Se promedian únicamente las instancias aprobadas.
 - Presentación y aprobación de un ejercicio integrador de las clases prácticas de laboratorio.

2.3.2 Condiciones para regularizar la materia:

- 75% de asistencia a las clases prácticas (aula y laboratorio)
- Aprobación de dos exámenes parciales (primer y segundo parcial) con nota mayor o igual a 6.
- Presentación y aprobación de un ejercicio integrador de las clases prácticas de laboratorio.

Los exámenes parciales cuentan con un recuperatorio cada uno, y un extraordinario para recuperar alguno de los dos.

Para las instancias de exámenes parciales se elaborarán exámenes de complejidad similar a los problemas planteados en los trabajos prácticos, que permitan determinar la comprensión de los alumnos respecto a los conceptos evaluados.

En cada instancia evaluativa, se explicaran los criterios de evaluación, los cuales se corresponden con los objetivos de aprendizaje.

2.3.3 Condiciones para aprobar la materia con examen final

- Los alumnos regulares rinden un examen oral ó escrito sobre los contenidos del programa vigente.
- Los alumnos libres deben aprobar un examen práctico (en el ordenador) para luego rendir el examen teórico.

2.4 PROGRAMA ANALITICO

2.4.1. CONTENIDOS POR UNIDAD:

Unidad 1: Sustentación de conceptos básicos

Tema I. Conceptos de sustentación: estructuras de datos simples. Funciones y Procedimientos. Archivos. Algoritmos. Concepto y Características. Concepto de programa. Escritura de algoritmos/programas. Programación Estructurada. Funciones y Procedimientos. Búsqueda de máximos y mínimos. Tratamiento de arreglos. Archivos: Operaciones simples.

Unidad 2: Estructuras de Datos

Tema II. Estructuras de datos Compuestas. Declaración de listas. Operaciones sobre listas. Declaración de pilas. Operaciones sobre pilas. Declaración de colas. Operaciones sobre colas. Cola circulares. Doble cola.

Tema III. Estructuras de Datos compuestos enlazados. Listas enlazadas. Punteros. Procesamiento de listas enlazadas. Listas circulares. Listas doblemente enlazadas. Árboles. Árboles binarios. Árbol binario de búsqueda. Grafos. Buscando el camino entre dos vértices del grafo. Tablas Hash. Colas de Prioridades.

Tema IV: Archivos. Operaciones más complejas Gestión de Archivos: ABM y Consultas en archivos de organización secuencial y directa. Corte de control. Unión de archivos.

Tema V: Datos Abstractos Introducción a los tipos de datos abstractos (TAD). Abstracción de datos. Concepto sobre tipos de datos. Módulos, interfaz e implementación. Encapsulamiento de datos. Diferencia entre tipo de dato y tipo abstracto de datos. Ventajas del uso de TAD. Formas de abstracción. Requerimiento y diseño de un TAD.

Unidad 3: Diseño y Análisis de Algoritmos

Tema VI: Ordenación y Búsqueda. Introducción a la ordenación. Clasificación de los algoritmos de ordenación. Métodos Directos. Métodos Logarítmicos. Intercalación. Ordenación por montículos. Problema de la Búsqueda Estática. Búsqueda Secuencial. Búsqueda Binaria. Búsqueda Interpolada. Cola de prioridades.

Tema VII: Recursividad. Naturaleza. Definición. Implementación. Recursividad directa e indirecta. Recursividad Infinita. Ventajas y desventajas. Comparación con la iteración.

Tema VIII: Estrategias de Algoritmos Complejos. Introducción a los algoritmos complejos. Estrategias de solución. Divide y Vencerás. Algoritmos Voraces. Algoritmo de recorrido de grafos. Algoritmo de Prim, Kruskal y Dijkstra. Programación Dinámica. Principio de Optimalidad. Vuelta atrás. Ramificación y Poda. Algoritmos Evolutivos. Ejemplos de aplicación.

Tema IX: Complejidad en los algoritmos. Recursos que requiere un algoritmo. Operación elemental. Principio de Invarianza. Análisis del Caso Mejor, Peor y Medio. Eficiencia Algorítmica. Notación asintótica. Ecuaciones de recurrencia. Optimización. Introducción a los NP Completitud. Clase P y NP. Pruebas y depuración de Algoritmos.

2.5.2. BIBLIOGRAFIA:

Específica:

1. ESTRUCTURA DE DATOS EN C. Luis Joyanes y Otros. 2007. Editorial MCGRAW-HILL. ISBN: 978-84-5645-9.
2. INTRODUCCION AL DISEÑO Y ANALISIS DE ALGORITMOS. UN ENFOQUE ESTRATEGICO. R.C.T. Lee; S.S. Tseng; R.C. Chang; Y.T. Tsai 2007. MCGRAW-HILL. ISBN: 978-970-10-6124-4.
3. ANALISIS Y DISEÑO DE ALGORITMOS: UN ENFOQUE TEORICO PRÁCTICO. Dr. Jose I. Pelaez. 2003. Editorial: UMA. ISBN: 84-7496-971-9.
4. ESTRUCTURAS DE DATOS EN JAVA. MARK Allen Weiss. 2006. Editorial: PEARSON. ISBN: 978-84-7829-035-2

5. PROGRAMACIÓN; Castor F. Herrmann, María E. Valesani.; 2001; Editorial: MOGLIAS.R.L..ISBN: 9874338326.

General:

6. ESTRUCTURA DE DATOS Libro de Problemas, Luis Joyanes Aguilar, Fernández Matilde, Rodríguez Luis; 1999; Editorial: MCGRAW-HILL. ISBN: 8448122984.
7. Fundamentos de programación. Algoritmos, estructuras de datos y objetos; Luis Joyanes Aguilar; 2003; Editorial: MCGRAW-HILL. ISBN: 8448136642.
8. ALGORITMOS + ESTRUCTURAS DE DATOS = PROGRAMAS; Niklaus Wirth; 2000; Editorial: C.I.E / DOSSAT. ISBN: 8421901729.
9. ESTRUCTURA DE DATOS; Cairó y Guardati; 2002; Editorial: MCGRAW-HILL. ISBN: 9701035348.

Material didáctico:

- Material de estudio y guías de trabajos prácticos para la asignatura Algoritmo y Estructura de Datos II. Cuerpo Docente de la asignatura Algoritmo y Estructura de Datos II. Área Programación. Dpto. Informática. FaCENA.

3. PROGRAMA DE EXAMEN:

Bolillas	Temas		
1	1	9	5
2	2	8	4
3	3	7	5
4	4	3	2
5	5	3	1
6	6	4	9
7	7	3	8
8	8	2	7
9	9	1	6

4. NOMINA DE TRABAJOS PRACTICOS

Serie de Trabajos Prácticos	
Nro.	Descripción
1	Repaso de estructuras simples: variables, arreglos, funciones y procedimientos.
2	Resolución de problemas utilizando estructuras compuestas. Listas. Pilas. Colas. Cola doble. (Implementadas con arreglos).
3	Resolución de problemas utilizando estructuras compuestas enlazadas. Listas enlazadas. Grafo. Arboles. (Implementadas con punteros).
4	Ejercicios de manejos más complejos de archivos.
5	Resolución de problemas aplicando estructuras de tipos abstractos de datos (TAD).
6	Resolución de problemas utilizando técnicas de ordenamiento, búsqueda y optimización.
7	Resolución de problemas recursivos.
8	Calculo de complejidad algorítmica. Eficiencia. Pruebas de algoritmos.

5. CRONOGRAMA DE ACTIVIDADES

Se detallan en la planificación que la asignatura eleva al inicio del dictado en cada ciclo lectivo.

6. Efectos sobre la formación integral del alumno

- Estará capacitado para identificar las distintas partes de un programa.
- Estará capacitado para construir los algoritmos para la obtención de una solución ante el problema que se plantee.
- Estará en condiciones de seleccionar y definir la estructura de datos más adecuada para la resolución del problema en cuestión.
- Una vez diseñada la solución, podrá optimizar el algoritmo para mejorar la eficiencia en la utilización de tiempo y de memoria del ordenador.

7. RECURSOS HUMANOS

Apellido y Nombres	Cargo	Área	Titulo (max)	Dedicación
Vallejos, Oscar A.	ADJ (A/C)	Programación	Mgter. En Informática y Computación	10 Hs
Valesani, Maria E.	JTP	Programación	Mgter. En Informática y Computación	10 Hs
Zacarias, Ramón A.	JTP	Programación	Experto en Estadística y Computación	10 Hs
Company, Ana M.	JTP	Programación	Licenciada en sistemas de Inf.	10 Hs.
Bougardth, Mariela	Aux. 1ra	Programación	Licenciada en Sistemas de Inf.	10 Hs.
Alfonzo, Pedro		Programación	Mgter en Informática y Computación	Mayor dedicación
Greiner, Cristina		Programación	Mgter. En informática y Computación	Mayor dedicación
Espindola, Cecilia		Programación	Lic. En Sistemas de Información	Colaboración