

PROGRAMA DE ESTUDIOS POR COMPETENCIAS
Unidad de Aprendizaje: ALGORITMOS COMPUTACIONALES

I. IDENTIFICACIÓN DEL CURSO

ESPACIO ACADÉMICO: Facultad de Contaduría y Administración, CU UAEM Temascaltepec, Ecatepec, Valle de México, Texcoco, Valle de Teotihuacan y Atlacomulco								
Programa Educativo: Licenciatura en Informática Administrativa				Área de docencia: Desarrollo de Software				
Aprobación por los H.H. Consejos Académico y de Gobierno:				Programa elaborado por: LIA Magali Lecuona Patiño ING Erika Hidalgo Martinez LIA Rocío Palma Lopez LIA. Tania Elena Valdes Gil		Programa reestructurado por: M. en T. I. Gisela Regina Baena Castro (CU Temascaltepec) M. en A. Ana Luisa Ramírez Roja (CU Ecatepec) L.I.A. Tayde Nancy Chora Portilla (CU Valle de México) L.C. María Dolores Arévalo Zenteno (CU Texcoco) M. en C. Leticia Arévalo Cedillo (CU Texcoco) L.S.C. Jaqueline Sánchez Espinoza (CU Teotihuacan) Ing. Carlos Eduardo Torres Reyes (CU Atlacomulco) M. en A.T.I. Raúl Reyes Hernández (Facultad) L.I.A Magali Leucona Patiño (FCA UAEM)		
Fecha de elaboración: Julio del 2005		Fecha de aprobación: 24 y 25 de Septiembre del 2007		Revisado por: ISC. Gabriela Gaviño Ortiz LIA Magali Leucona Patiño ING Erika Hidalgo Martinez				
Fecha de revisión: Febrero del 2006		Fecha de Aprobación: Marzo del 2011		Revisado por: LIA Magali Leucona Patiño Ernesto Torales Noguez				
Fecha de reestructuración: 04 de Junio del 2010								
Clave	Horas de teoría	Horas de práctica	Total de horas	Créditos	Tipo de Unidad de Aprendizaje	Carácter de la Unidad de Aprendizaje	Núcleo de formación	Modalidad

L16102	2	4	6	8	Curso Teórico Práctico	Obligatoria	Sustantivo	Presencial
Prerrequisitos (Conocimientos previos): Conocimientos de aritmética y lógica				Unidad de Aprendizaje Antecedente: Ninguna		Unidad de Aprendizaje Consecuente: Programación Estructurada		
Programas educativos en los que se imparte: Licenciatura en Informática Administrativa								

II. PRESENTACIÓN

Una de las principales funciones del Licenciado en Informática Administrativa consiste en el planteamiento de soluciones y optimización de problemas basados en la organización de recursos computacionales, tomando como base principal el desarrollo de software.

La presente unidad de aprendizaje crea conocimientos, genera las competencias básicas para la fase de desarrollo y las habilidades necesarias para plantear soluciones utilizando Algoritmos Computacionales.

Constituye una base importante para la informática, no sólo para desarrollar software, sino también fomenta el análisis y planteamiento de soluciones en la automatización de procesos.

Una de las habilidades del Licenciado en Informática Administrativa es la construcción de software que ayude a llevar a cabo los procesos diarios de las organizaciones en forma eficaz y eficiente, de igual forma, tiene los conocimientos que permiten dirigir, crear e implementar proyectos que combinen las transacciones electrónicas de datos y el uso de las tecnologías computacionales con los métodos, técnicas y herramientas de carácter administrativo y contable, para la instrumentación de soluciones informáticas de calidad, participando así en la búsqueda del éxito y competitividad de la empresa.

La solución de problemas basados en la elaboración de programas en algunos de los diversos lenguajes, es uno de los campos del Licenciado en Informática, motivo por el cual, esta unidad de aprendizaje pretende brindar los conocimientos básicos sobre las técnicas y métodos para una solución óptima basada en la estructuración adecuada de algoritmos, dando las bases sólidas que permitan el desarrollo posterior del alumno en este ámbito.

El Licenciado en Informática Administrativa debe poseer una personalidad crítica, creativa, humanista, y poseer una iniciativa para la solución de problemas, cooperativo y deseo constante de superación, todo esto lo forma como un individuo integral comprometido con la

sociedad.

III. LINEAMIENTOS DE LA UNIDAD DE APRENDIZAJE

DOCENTE	DISCENTE
- Asistir puntualmente a clase - Dar a conocer el programa de la unidad de aprendizaje - Realizar la planeación de clase en tiempo y forma - Conducirse con respeto dentro y fuera del aula de clase - Respetar los lineamientos y normas establecidas dentro de la Institución. - Fomentar la creación de programas de autoría propia - Entregar puntualmente la evaluación del alumno, en la fecha previamente establecida por el profesor - Fomentar el cambio de experiencias - Guiar a los alumnos para lograr las competencias establecidas - Tratar de cubrir el programa de estudios al 100% - Dar revisión de exámenes - Respetar las disposiciones del reglamento de evaluación y permanencia de estudios.	- Autoaprendizaje, para conocer toda el área de programación" - Contar con al menos el 80% de asistencias para presentar examen ordinario - Contar con al menos con el 60% de asistencias para presentar examen extraordinario - Contar con al menos 40% de asistencias para presentar examen de título de suficiencia - Entregar las evidencias de desempeño y de producto final de cada una de las unidades de competencia, de acuerdo a las fechas establecidas previamente - Guardar el debido respeto a los compañeros y al profesor. - Mostrar una actitud de trabajo en equipo y colaboración - Participar en forma dinámica en las sesiones de asignatura, cuestionando y aportando ideas

IV. PROPÓSITO DE LA UNIDAD DE APRENDIZAJE

El alumno:

Resolverá problemas mediante el desarrollo de algoritmos en cualquiera de sus formas de presentación: pseudocódigos, diagramas y programas

V. COMPETENCIAS GENÉRICAS

El alumno:

- Solucionar problemas reales utilizando la abstracción, para su implementación en la computadora.
- Desarrollar y sustentar la solución de problemas reales mediante la aplicación de algoritmos, considerando de manera crítica y reflexiva el punto de vista de sus compañeros.
- Recolectar y analizar los requerimientos de información.
- Resolver problemas con o sin un equipo electrónico.
- Utilizar ideas matemáticas y técnicas.
- Diseñar la solución a un problema utilizando diagramación y pseudocódigo en lenguajes de programación de alto nivel.
- Diseñar y plantear soluciones mediante el desarrollo de Algoritmos Computacionales.
- Identificar los procesos involucrados en el desarrollo de software.
- Comprender el esquema de funcionamiento de las computadoras y su forma de ejecutar algoritmos.

VI. ÁMBITOS DE DESEMPEÑO PROFESIONAL

- Consultorías en Tecnologías de la Información.
- Organizaciones dedicadas a la investigación científica.
- Departamentos de Desarrollo de Sistemas de organizaciones públicas y privadas
- Trabajo a Distancia
- Grupos de Investigación
- Educación
- Empresas públicas y privadas donde se desarrollen procesos de información automatizado con herramientas computacionales.

VII. ÁMBITOS DE APRENDIZAJE



- Salón de clases
- Laboratorio de Cómputo
- La red mundial Internet

VIII. NATURALEZA DE LA COMPETENCIA

(Inicial, entrenamiento, complejidad creciente, ámbito diferenciado)

Inicial, entrenamiento

IX. ESTRUCTURA DE LA UNIDAD DE APRENDIZAJE

Fundamentos:

1. Inducción, e implementación de soluciones en computadoras:

Describir el funcionamiento de una computadora, el proceso de solución de problemas mediante algoritmos y el ciclo de vida del software.

2. Elementos para el desarrollo de Algoritmos:

Identificar los elementos básicos en el planteamiento de algoritmos para solucionar problemas: tipos de datos, variables, operadores y expresiones.

3. Representación de Algoritmos:

Definir las formas de representación de algoritmos: pseudocódigo, diagramación y programación. Identificando la sintaxis en cada una de ellas.

Desarrollo de Habilidades:

4. Estructuras de Control

De Secuencia:

Identificar y aplicar las sentencias de "Asignación", "Entradas y Salidas de datos" y el "Orden de ejecución de las sentencias" diseñando soluciones a problemas reales para su automatización mediante el planteamiento de algoritmos.

De Selección:

Emplear las sentencias de selección *“Simple”*, *“Anidada”* y *“Múltiple”* en el planteamiento de soluciones a problemas condicionales, analizando sus diferentes formas de uso y estructuración para el diseño de algoritmos.

De Iteración o repetición:

Aplicar las sentencias repetitivas *“Mientras”*, *“Para”* y *“Repita”* además las estrategias de *“Contador”* y *“Acumulador”* para proponer soluciones a problemas de naturaleza repetitiva; valorando los métodos de control de ciclos, *“Centinela”* o *“Bandera”* y *“Contador”*

5. Conversión de Algoritmos en Programas de Cómputo:

Identificar los elementos, la estructura y sintaxis de un lenguaje de programación específico, aplicando el uso de variables y estructuras de control de secuencia, selección e iteración para la codificación y conversión de algoritmos en programas de cómputo.

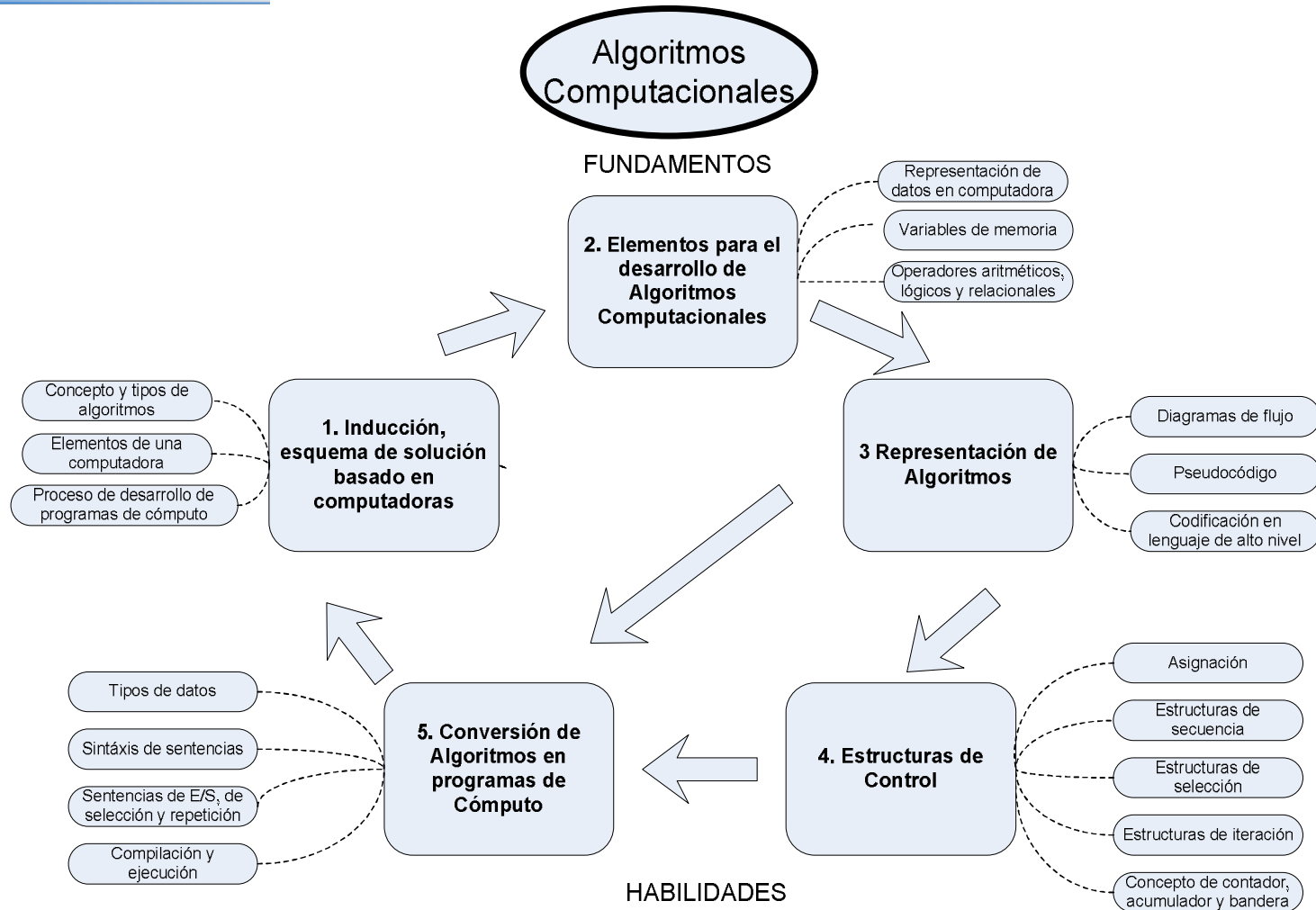
X. SECUENCIA DIDÁCTICA

No.	TAREA	HT	HP	TOTAL
I	Inducción, e implementación de soluciones en computadoras	6	4	10
II	Elementos para el desarrollo de algoritmos	4	2	6
III	Representación de algoritmos	6	6	12
IV	Estructuras de Control	14	20	34
V	Conversión de Algoritmos en Programas de Cómputo	14	20	34

T O T A L

96

MAPA CONCEPTUAL



XI. DESARROLLO DE LA UNIDAD DE APRENDIZAJE

UNIDAD DE COMPETENCIA I	ELEMENTOS DE COMPETENCIA		
	Conocimientos	Habilidades	Actitudes/ Valores
Inducción e implementación de soluciones en computadoras: Describir: El funcionamiento de una computadora El proceso de solución de problemas mediante algoritmos El ciclo de vida del software	-Tipos de Algoritmos. -Fases en la resolución de problemas. -Elementos de una computadora (memoria, procesador, E/S), características y su funcionamiento en la ejecución de algoritmos. -Proceso de desarrollo de programas de cómputo - Conceptos de lenguaje de máquina y de alto nivel	• Capacidad de análisis • Capacidad de Síntesis • Observación • Razonamiento Lógico • Comunicación oral y escrita • Investigación • Creatividad e innovación • Concentración • Aplicación del conocimiento	• Disposición de trabajo en equipo • Responsable • Participativo • Crítico • Perseverante • Dispuesto al cambio • Honestidad • Respeto
ESTRATEGIAS DIDÁCTICAS: -Mapas Mentales. -Técnica expositiva. -Discusión dirigida. -Estudios de caso. -Investigación bibliográfica.	RECURSOS REQUERIDOS Pizarrón. Computadora. Cañón (proyector) Internet Libros Bibliografía		TIEMPO DESTINADO 10 horas.
CRITERIOS DE DESEMPEÑO I	EVIDENCIAS		
	DESEMPEÑO	PRODUCTOS	
-Plantear Algoritmos en lenguaje natural -Describir los elementos de una computadora involucrados en la ejecución de algoritmos, y	-Plantear algoritmos para solucionar problemas. -Esquema de funcionamiento de una	-Algoritmos escritos en lenguaje natural.	

sus características.	computadora que describe la ejecución de programas o algoritmos.	
----------------------	--	--

UNIDAD DE COMPETENCIA II	ELEMENTOS DE COMPETENCIA		
	Conocimientos	Habilidades	Actitudes/ Valores
Elementos para el desarrollo de Algoritmos: Identificar los elementos básicos en el planteamiento de algoritmos para solucionar problemas: tipos de datos, variables, operadores, expresiones.	1. Representar datos en una computadora (códigos y sistemas numéricos). 2. Definir y hacer uso del concepto de variables de memoria. 3. Definir y hacer uso de expresiones aritmética (operadores aritméticos y jerarquía) 4. Definir y hacer uso de expresiones relacionales (operadores relacionales y lógicos).	• Capacidad de análisis • Capacidad de Síntesis • Observación • Razonamiento Lógico • Comunicación oral y escrita • Investigación • Creatividad e innovación • Aplicación del conocimiento	• Disposición de trabajo en equipo • Responsable • Participativo • Crítico • Perseverante • Dispuesto al cambio • Honestidad • Respeto
ESTRATEGIAS DIDÁCTICAS: -Cuadros sinópticos -Técnica expositiva -Ejemplos y ejercicios resueltos -Diagramas de procesos de solución de problemas.	RECURSOS REQUERIDOS Pizarrón Computadora Internet Libris Cañón (proyector) Software		TIEMPO DESTINADO 6 horas

CRITERIOS DE DESEMPEÑO II	EVIDENCIAS	
	DESEMPEÑO	PRODUCTOS
Algoritmos en los que se resalta el uso correcto de variables y expresiones aritméticas y lógicas.	Resolver problemas matemáticos y lógicos de aplicación cotidiana	Algoritmos simples

UNIDAD DE COMPETENCIA III	ELEMENTOS DE COMPETENCIA		
	Conocimientos	Habilidades	Actitudes/ Valores
Representación de Algoritmos: Definir las formas de representación de algoritmos: pseudocódigo, diagramación y programación; identificando la sintaxis en cada una de ellas.	1. Elementos y sintaxis de los diagramas de flujo. Símbolos para E/S, asignación, selección simple y múltiple, Ciclos y estructuras de control. 2. Elementos y sintaxis para pseudo-código. 3. Elementos de codificación en diferentes lenguajes de programación de alto nivel	• Capacidad de análisis • Capacidad de Síntesis • Observación • Razonamiento Lógico • Comunicación oral y escrita • Investigación • Creatividad e innovación • Aplicación del conocimiento	• Disposición de trabajo en equipo • Responsable • Participativo • Crítico • Perseverante • Dispuesto al cambio • Honestidad • Respeto

ESTRATEGIAS DIDÁCTICAS: -Técnica expositiva. -Matrices de contraste -Cuadros sinópticos -Resumen de contenidos -Ejemplo prácticos en computadora -Discusión en grupo	RECURSOS REQUERIDOS Pizarrón Computadora Internet Libros Cañón (proyector) Software		TIEMPO DESTINADO 12 horas
CRITERIOS DE DESEMPEÑO III	EVIDENCIAS		
	DESEMPEÑO		PRODUCTOS
Algoritmos que contengan sentencias de E/S, Asignación, Selección y Repetición, en las diferentes formas de representación de algoritmos. Algoritmos codificados en lenguaje de alto nivel.	Algoritmos en diferentes formas de representación (Diagrama de flujo, pseudo-código, diagramas estructurados) Algoritmos en algún lenguaje de programación de alto nivel.		Algoritmos representados en diagrama de flujo y pseudo-código, codificados en lenguajes de programación de alto nivel.
UNIDAD DE COMPETENCIA IV	ELEMENTOS DE COMPETENCIA		
	Conocimientos	Habilidades	Actitudes/ Valores
Estructuras de Control: Secuencia: Identificar y aplicar las sentencias de Asignación Entradas y Salidas de datos y el Orden de ejecución de sentencias diseñando soluciones a problemas reales para su automatización mediante el planteamiento de algoritmos. Selección: Emplear las sentencias	1. Secuencia de sentencias 2. Asignación 3. Entradas. 4. Salidas. 5. Sentencia Si-Entonces y Si-Si no 6. Sentencia Caso 7. Estrategias de contador y	• Capacidad de análisis y síntesis • Observación • Razonamiento Lógico • Creatividad e Innovación • Aplicación del conocimiento • Comprensión y aplicación de las reglas de sintaxis	• Disposición de trabajo en equipo • Responsable • Participativo • Crítico • Perseverante • Dispuesto al cambio • Honestidad • Respeto • Autocrítica y auto corrección • Asertividad

de selección óSimpleô óAnidadaô y óMúltipleô en el planteamiento de soluciones a problemas condicionales, analizando sus diferentes formas de uso y estructuración para el diseño de algoritmos Iteración: Aplicar las sentencias repetitivas óMientrasô óParaô y óRepiteô en combinación con las estrategias de ócontadorô para proponer soluciones a problemas de naturaleza repetitiva; valorando los métodos de control de ciclos: ócentinela o banderaô y ócontadorô	acumulador 8. Sentencias repetitivas óParaô óMientrasô óRepiteô 9. Estrategias de control de ciclos (contador, dato bandera) 10. Sintaxis de todas las sentencias: secuenciales, de selección e iterativas	Desarrollo de planteamientos individuales que muestren la lógica personal	
ESTRATEGIAS DIDÁCTICAS: -Técnica expositiva -Discusión en grupo -Ejemplos prácticos -Resolución de problemas -Comprobación, Pruebas de escritorio	RECURSOS REQUERIDOS: Pizarrón Computadora Internet Libros Software (simulador, compilador) Cañón (proyector)	TIEMPO DESTINADO: 34 horas	
CRITERIOS DE DESEMPEÑO IV	EVIDENCIAS		
	DESEMPEÑO	PRODUCTOS	
Desarrollar algoritmos de secuenciación, de selección simple, anidada y múltiple, y de naturaleza iterativa, aplicando las sentencias óparaô ómientrasô y órepiteô con lógica	Planteamiento de soluciones a problemas de secuenciación, selección y repetición, escritos en cualquiera de las formas de representación de algoritmos.	Planteamientos de algoritmos de secuenciación, selección simple, anidada, múltiple, y algoritmos repetitivos simples, de ciclos anidados y repetición, tantos como sean	

asertiva, aplicando correctamente el concepto de variables, expresiones aritméticas en sentencias de E/S y Asignación usando la sintaxis correcta de cada una de ellas. Mostrar evidencias del dominio de los métodos para control de ciclos (contadores y dato bandera).		necesarios para demostrar la habilidad en el manejo de estas estructuras de control.
---	--	--

UNIDAD DE COMPETENCIA V	ELEMENTOS DE COMPETENCIA		
	Conocimientos	Habilidades	Actitudes/ Valores
Conversión de algoritmos en programas de cómputo: Identificar los elementos, la estructura y sintaxis de un lenguaje de programación específico, aplicando el uso de variables y estructuras de control de secuencia, selección e iteración para la codificación de algoritmos en programas de cómputo.	1. Estructura de un programa. 2. Tipos de datos básicos y sintaxis de declaración. 3. Sentencias de E/S de datos (sintaxis). 4. Sentencias selectivas 5. Sentencias repetitivas. 6. Compilación y ejecución.	• Capacidad de análisis y síntesis • Observación • Razonamiento Lógico • Comunicación oral y escrita • Investigación • Creatividad e innovación • Concentración • Aplicación del conocimiento • Comprensión y aplicación de las reglas de sintaxis • Desarrollo de planteamientos individuales que muestren la lógica personal	• Disposición de trabajo en equipo • Responsable • Participativo • Crítico • Perseverante • Dispuesto al cambio • Honestidad • Respeto • Autocrítica y auto corrección • Asertividad

ESTRATEGIAS DIDÁCTICAS: -Técnica expositiva -Ejemplos prácticos -Resolución de problemas -Discusión en grupo -Comprobación -Pruebas de escritorio	RECURSOS REQUERIDOS Pizarrón Computadora Internet Libros Cañón (proyector) Compilador del lenguaje de programación	TIEMPO DESTINADO 34 horas	
CRITERIOS DE DESEMPEÑO V	EVIDENCIAS		
	DESEMPEÑO	PRODUCTOS	
Codificar programas en un lenguaje de alto nivel (C), de los desarrollados en la unidad 4, en los que se aplique las sentencias de Asignación, E/S, selección simple y múltiple y todas las sentencias repetitivas. Los programas deben compilarse y ejecutarse correctamente.	Codificar programas de acuerdo a la sintaxis de un lenguaje de alto nivel (lenguaje C).	Programas codificados en un lenguaje de Alto nivel (lenguaje C).	

XII. EVALUACIÓN Y ACREDITACIÓN

EVALUACIÓN Y ACREDITACIÓN (CONTINUA INTEGRADORA)

Durante el semestre se aplicarán dos evaluaciones parciales.

Primer Evaluación Parcial (Unidades I, II, III y IV)		Segunda Evaluación Parcial (Unidades IV, y V)		Fechas de entrega.	Fechas de retroalimentación de productos y revisión de exámenes.
Conocimientos (Examen)	60 %	Conocimientos (Examen)	60 %	productos	Dar revisión siguiente clase
Portafolio de evidencias. Algoritmos simples Algoritmos representados en diagramas de flujo y pseudocódigo empleando los temas de las unidades I, II, III y parte de la IV	20 %	Portafolio de evidencias. Programas codificados en un lenguaje de alto nivel (lenguaje C) aplicando los temas de las unidades anteriores	20 %	A más tardar una semana antes de cada parcial.	Dar retroalimentación una o dos clases posteriores a fecha de entrega del producto
Tareas y participaciones.	20 %	Tareas y participaciones.	20 %		
Total	100%	Total	100 %		

N O T A: Los exámenes serán aplicadas en la fecha programada, según calendario.

EXAMEN (teórico práctico) ORDINARIO 100%,

EXAMEN (teórico práctico) EXTRAORDINARIO 100%,

EXAMEN (teórico práctico) TÍTULO DE SUFICIENCIA 100%

XIII. REFERENCIAS

BIBLIOGRAFÍA BÁSICA	BIBLIOGRAFÍA COMPLEMENTARIA
1. Correa, Guillermo (1992): <i>Desarrollo de Algoritmos y sus Aplicaciones en Basic, Pascal, Cobol y C</i> Mc Graw Hill 2. Joyanes Aguilar, Luis (2003): <i>Fundamentos de Programación</i> 3ra. Edición, Mc Graw Hill 3. Cairó, Osvaldo (2004): <i>Metodología de la Programación</i> 3ra. Edición, Alfaomega 4. Criado Clavero, María Asunción (2006): <i>Programación en Lenguajes Estructurados</i> Alfaomega 5. López Román, Leobardo (2003): <i>Programación Estructurada un Enfoque Algoritmico</i> Alfaomea, 6. Charte, Francisco (2001): <i>Introducción a la Programación</i> Anaya Multimedia	7. Joyanes Aguilar, Luis (2004): <i>Algoritmos y Estructuras de Datos, una perspectiva en C</i> Mc Graw Hill 8. Joyanes Aguilar, Luis (2003): <i>Fundamentos de Programación (Algoritmos, Estructuras de Datos y Objetos)</i> Mc Graw Hill