

1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura: Algoritmos y lenguajes de programación
Carrera: Ingeniería Industrial
Clave de la asignatura: INM-0407
Horas teoría-horas práctica-créditos 3-2-8

2.- HISTORIA DEL PROGRAMA

Lugar y fecha de elaboración o revisión	Participantes	Observaciones (cambios y justificación)
Instituto Tecnológico de Celaya del 11 al 15 agosto 2003.	Representante de las academias de ingeniería industrial de los Institutos Tecnológicos.	Reunión Nacional de Evaluación Curricular de la Carrera de Ingeniería Industrial
Instituto Tecnológico de Reynosa del 2003	Academia de Ingeniería Industrial y Sistemas Computacionales	Análisis y enriquecimiento de las propuestas de los programas diseñados en la reunión nacional de evaluación
Instituto Tecnológico de La Laguna del 26 al 30 abril 2004	Comité de Consolidación de la carrera de Ingeniería Industrial.	Definición de los programas de estudio de la carrera de Ingeniería Industrial.

3.- UBICACIÓN DE LA ASIGNATURA

a). Relación con otras asignaturas del plan de estudio

Anteriores	
Asignaturas	Temas
Ninguna	

Posteriores	
Asignaturas	Temas
Investigación de operaciones I y II	
Simulación	

b). Aportación de la asignatura al perfil del egresado

- Diseña, administra y mejora sistemas de materiales.
- Diseña, implementa y controla los sistemas integrados de manufactura.
- Diseña, implementa y administra sistemas de mantenimiento.
- Planea y diseña la localización y distribución de instalaciones para la producción de bienes y servicios.
- Desarrolla y utiliza tecnologías de vanguardia en su área de competencia

4.- OBJETIVO(S) GENERAL(ES) DEL CURSO

El estudiante desarrollará la lógica algorítmica para aplicar los conocimientos de un lenguaje de programación estructurado en la resolución de problemas científicos y tecnológicos.

5.- TEMARIO

Unidad	Temas	Subtemas
1	Introducción a la computación	1.1 Nuevas Tecnologías de la información. 1.2 Elementos de un sistema computacional 1.2.1 Hardware 1.2.2 Software: de operaciones, de traducciones y aplicación. Firmware: definición, ventajas y desventajas 1.3 Clasificación de los sistemas operativos: la familia de Windows, Arquitectura Macintosh, Uníx y Linux. 1.4 Paquetería de software: Hoja de calculo y modelos de bases de datos

2	Desarrollo de lógica algorítmica	<p>2.1 Metodología para la solución de problemas.</p> <p>2.2 Metodología para el diseño de software: Top down, Bottom up, modular y programación estructurada.</p> <p>2.3 Definición de lenguajes algorítmicos</p> <p>2.3.1 Elementos y reglas de la representación grafica y manuscrita de los algoritmos (diagrama de flujo, diagrama N-S, diagrama estructurado y pseudocódigo.</p> <p>2.3.2 Implementación de algoritmos secuenciales (utilizando notación algebraica)</p> <p>2.4 Pruebas y depuración</p>
3	Conocimiento de programación y un lenguaje estructurado	<p>3.1 Introducción a la programación</p> <p>3.1.1 Definición de programa</p> <p>3.1.2 Definición de programación</p> <p>3.1.3 Definición de lenguaje de programación</p> <p>3.2 Introducción y orígenes del lenguaje</p> <p>3.3 Estructura básica de un programa</p> <p>3.4 Datos</p> <p>3.4.1 Tipos de datos</p> <p>3.4.2 identificadores.</p> <p>3.4.3 Almacenamiento, direccionamiento y representación en memoria.</p> <p>3.4.4 Proposición de asignación</p> <p>3.5 Operadores, operandos y expresiones.</p> <p>3.5.1 Prioridad de operadores, evaluación de expresiones</p> <p>3.6 Proceso de creación de un ejecutable</p>
4	Funciones	<p>4.1 Funciones</p> <p>4.1.1 Definición de funciones</p> <p>4.1.2 Funciones estándar</p> <p>4.2 Entrada y Salida de datos</p> <p>4.3 Funciones definidas por el usuario</p> <p>4.3.1 Pase por valor</p> <p>4.3.2 Pase por referencia</p> <p>4.4 Punteros</p> <p>4.4.1 Definición de punteros</p> <p>4.4.2 Paso de variable</p>

5	Estructuras de selectivas	5.1 Selectiva simple 5.2 Selectiva doble 5.3 Selectiva anidada 5.4 Selectiva múltiple
6	Estructuras de repetición	6.1 Repetir mientras 6.2. Repetir hasta 6.3 Repetir desde
7	Arreglos	7.1 Arreglo Unidimensionales 7.1.1 Conceptos básicos 7.1.2 Operaciones Aplicaciones 7.2 Arreglo Bidimensionales 7.3 Conceptos básicos 7.4 Operaciones 7.5 Aplicaciones 7.6 Registros 7.7 Archivos

6.- APRENDIZAJES REQUERIDOS

- Álgebra matricial elemental
- Conocimientos básicos de computación (manejo de la computadora)

7.- SUGERENCIAS DIDÁCTICAS

- Realizar el encuadre del grupo:
- Presentación del maestro y cada uno de los alumnos utilizando una técnica
- grupal
- Dar a conocer el objetivo, contenido y bibliografía del curso
- Definir con el grupo el porcentaje que le correspondería a cada uno de los conceptos a evaluar
- Solicitar y comentar las expectativas de los alumnos en relación al curso
- Diagnosticar el nivel de conocimiento del grupo mediante una prueba
- Utilizar equipo audiovisual
- Fomentar el trabajo en equipo
- Solicitar al alumno propuestas de problemas o casos a resolver que sean significativos para el alumno.
- Elaborar una guía de ejercicios para actividades extra clase
- Diseñar una guía de ejercicios para actividades extra clase
- Participación y desempeño del alumno en el aula y el laboratorio
- Fomentar el hábito de leer y traducir artículos en inglés
- Utilizar software que permita que el alumno interactúe con la computadora.
- Realizar proyectos de aplicación en las áreas de ingeniería industrial.
- Utilizar el lenguaje de programación estructurado.

8.- SUGERENCIAS DE EVALUACIÓN

- Ponderar tareas
- Participación y desempeño del alumno en el aula y el laboratorio.
- Dar seguimiento al desempeño integral del alumno en el desarrollo del programa
 - Exposición de temas
 - Dominio de los conceptos
 - Cumplimiento de tareas y ejercicios
 - Capacidad de la aplicación de los conocimientos en problemas reales
 - Asistencia
 - Participación
 - Participación en eventos académicos
 - Exámenes
 - Trabajos de investigación
 - Proyectos
 - Prácticas
 - Trabajo en equipo
 - Tránsito del conocimiento, entre otras
- Desarrollo de un proyecto final que integre todas las unidades de aprendizaje.
- Dar valor a la participación del alumno (mesas redondas y de debate).
- Integración del alumno en actividades de auto evaluación.
- Aplicar exámenes considerando que no sea el factor decisivo para la acreditación del curso.
- Cumplimiento de los objetivos y desempeño del alumno en las prácticas
- Se recomienda utilizar varias técnicas de evaluación con un criterio de evaluación específico para cada una de ellas. Los pesos que se le den a cada una de las técnicas se basará en la experiencia del docente.

9.- UNIDADES DE APRENDIZAJE

UNIDAD 1: DESARROLLO DE LA LÓGICA ALGORITMICA

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Comprenderá los conceptos básicos, propiedades y características de un sistema computacional así como sus tendencias	1.1 Investigar en Internet sobre las tecnologías de la información. 1.2 Valorar los elementos que conforma un equipo computacional en software, hardware y Firmware. 1.3 Realizar una clasificación de los sistemas operativos, valorando sus	4,6,12,13

actuales.	diferencias 1.4 Analizar y comprender los conceptos básicos de la paquetería de software	
-----------	---	--

UNIDAD 2: DESARROLLO DE LA LÓGICA ALGORITMICA

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Comprenderá el concepto de algoritmo y la terminología relacionada con los algoritmos. El alumno conocerá las características de las técnicas de diseño. Aplicara un lenguaje algorítmico gráfico o manuscrito.	2.1 Explicar las características lógicas de problemas susceptibles de ser computarizadas. 2.2 Elaborar algoritmo cotidiano. 2.3 Diseñara una solución de problema utilizando diferentes tipos de algoritmos.	1, 2, 3, 4, 5, 6, 7

UNIDAD 3: CONOCIMIENTO DE PROGRAMACIÓN Y UN LENGUAJE ESTRUCTURADO

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Conocerá la historia y características del lenguaje "C" El alumno comprenderá los conceptos básicos de la programación y escribirá expresiones aritméticas y lógicas en un lenguaje de programación	3.1 Buscar y seleccionar información histórica del lenguaje "C" 3.2 Comentar con el grupo los conceptos de: identificadores, constantes, variables y la proposición de asignación. 3.3 Construir y evaluar expresiones matemáticas 3.4 Buscar la información necesaria para instalar y configurar el lenguaje de programación a utilizar.	2, 6, 9, 10

UNIDAD 4: FUNCIONES

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Conocerá la estructura de una función y su implementación en un lenguaje de programación.	4.1 Compilar y ejecutar un programa modelo 4.2 Realizar ejemplos que requieran funciones estándar. 4.3 Realizar ejemplos utilizando funciones definidas por el usuario 4.4 Realizar ejemplos utilizando punteros	9, 10, 11

UNIDAD 5: ESTRUCTURAS DE SELECCION

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Comprenderá el uso y funcionamiento de las estructuras selectivas y las implementará en el desarrollo de aplicaciones.	5.1 Realizar una síntesis sobre el funcionamiento y aplicación de las estructuras secuenciales y selectivas. 5.2 Implementar el diseño de programas que requieran estructuras secuenciales y selectivas para probarlas en una aplicación.	9, 10, 11

UNIDAD 6: ESTRUCTURAS DE REPETICIÓN

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Comprenderá el uso y funcionamiento de las estructuras de repetición y las implementará en el desarrollo de aplicaciones.	6.1 Realizar una síntesis sobre el funcionamiento y aplicación de las estructuras de repetición 6.2 Implementar el diseño de programas que requieran estructuras de repetición para probarlas en una aplicación	9, 10, 11

UNIDAD 7: ARREGLOS

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Conocerá La representación interna de los arreglos unidimensionales y bidimensionales. Construirá modelos y desarrollar aplicaciones de software que requieran de estos. Conocerá los diferentes tipos de registros y de archivos	7.1 Desarrollar los programas de manipulación de los arreglos para realizar operaciones básicas 7.2 Distinguir los diferentes tipos de registros y archivos 7.3 Desarrollar los programas de manipulación de registros y archivos.	9, 10, 11

10. FUENTES DE INFORMACIÓN

1. Jean Paul Tremblay & Richard B. Bunt
Introducción a la ciencia de las computadoras (enfoque algorítmico)..
Ed. Mc.Graw Hill
2. Luis Joyanes Aguilar
Metodología de la programación
Ed. Mc. Graw Hill.
3. Luis Joyanes Aguilar
Problemas de metodología de la programación...
Ed. Mc. Graw Hill
4. Guillermo Levine
Introducción a la computación y a la programación estructurada.
Ed. Mc. Graw Hill
5. Goodman Hedetmiemi
Introduction to the design and analysis of algorithms
Ed. Mc Graw Hill
6. Luis Joyanes Aguilar
Fundamentos de programación, algoritmos y estructura de datos
Ed. Mc. Graw Hill.
7. Forsythe, Queman. Organik, Stenberg
Lenguajes de diagramas de flujo
Ed. Limusa.

8. Microsoft MS Dos
Ed. Guía de referencia para el usuario
9. Kernighan & Richie
Lenguaje "C"..
10. Schildt, Helbert
Programación de Lenguaje "C".
11. Stanley B. Lippman and Jasse Lajole
C++ Primer
12. Manuales del usuario de los sistemas operativos
13. Internet

11. PRÁCTICAS

Unidad I

Utilizando diagrama de flujo, diagrama N-S, diagrama estructurado y pseudocódigo, elaborar algoritmos.

Se pueden utilizar problemas presentados por el facilitador o utilizar problemas presentados por el alumno.

Unidad II

Elaborar ejercicios que impliquen el uso de operadores, operandos y expresiones.

Unidad III

Implementara aplicaciones que utilicen funciones con comportamientos que impliquen el uso de estructuras secuenciales y expresiones aritméticas.

Unidad IV

Implementar aplicaciones que utilicen funciones con comportamientos que impliquen el uso de estructuras selectivas y expresiones lógicas.

Unidad V

Implementar aplicaciones que utilicen funciones que impliquen el uso de estructuras repetitivas.

Unidad VI

El alumno desarrollará arreglos incluyendo todas las operaciones básicas que operan sobre un arreglo, tales como crear, insertar, eliminar, recorrer, buscar y modificar.