

1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura: Programación I
Carrera: Ingeniería Electrónica.
Clave de la asignatura: ECM-0431
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 Orizaba, del 25 al 29 de agosto del 2003.	Participantes de las academias de ingeniería electrónica de los Institutos Tecnológicos.	Reunión Nacional de Evaluación Curricular de la Carrera de Ingeniería Electrónica.
Institutos Tecnológicos de La Piedad y Nuevo Laredo, de septiembre a noviembre del 2003	Academias de Ingeniería Electrónica.	Análisis y enriquecimiento de las propuestas de los programas diseñados en la reunión nacional de evaluación
Instituto Tecnológico de Mexicali, del 23 al 27 de febrero 2004	Comité de consolidación de la carrera de Ingeniería Electrónica.	Definición de los programas de estudio de la carrera de Ingeniería Electrónica.

3.- UBICACIÓN DE LA ASIGNATURA

a). Relación con otras asignaturas del plan de estudio

Anteriores	
Asignaturas	Temas
Ninguna	

Posteriores	
Asignaturas	Tema
Programación II	- Programación en Lenguaje orientado a objetos
Instrumentación	- Tópicos de control asistido por computadora

b). Aportación de la asignatura al perfil del egresado

- Le permite programar en un lenguaje de alto nivel.
- Fomenta el análisis lógico.
- Desarrolla una metodología de trabajo.
- Adquiere habilidad para trabajo en equipo.

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

El alumno desarrollará algoritmos computacionales y programará en un lenguaje de programación estructurado.

5.- TEMARIO

Unidad	Temas	Subtemas
1	Introducción a la computación	1.1 Breve reseña de la computación 1.2 Impacto de las computadoras en la sociedad 1.3 Definiciones <ul style="list-style-type: none">• Lenguaje de Alto Nivel, bajo nivel, ensamblador y maquina.• Compilador• Traductor• Sistemas Numéricos• Arquitectura de computadoras• Sistemas operativos 1.4 Descripción de tipos de programación <ul style="list-style-type: none">1.4.1 Programación estructurada1.4.2 Programación orientada a objetos1.4.3 Programación de descripción de Hardware sistemas numéricos 1.5 Lenguajes de programación
2	Diseño de Algoritmos	2.1 Conceptos básicos 2.2 Metodología de la programación 2.3 Diagramas de flujo 2.4 Pseudocodigos

5.- TEMARIO (Continuación)

Unidad	Temas	Subtemas
3	Fundamentos del Lenguaje	3.1 Desarrollo histórico del lenguaje estructurado 3.2 Estructura general de un programa 3.3 Palabras reservadas 3.4 Representación de datos básicos 3.5 Declaración de variables, Constantes, Procedimientos y Funciones 3.6 Operaciones aritméticas, lógicas y relacionales. 3.7 Diseño, Edición, Compilación y Ejecución de programas
4	Estructuras de Decisión y Control	4.1 Estructura secuencial 4.2 Estructuras de Transferencia Incondicionales 4.3 Estructuras de Transferencia Condicionales 4.4 Estructuras Repetitivas (Bucles)
5	Arreglos y Archivos	5.1 Arreglos (Vectores y Matrices) 5.2 Manejo Archivos
6	Módulos	6.1 Diferencia entre Funciones y Procedimientos. 6.2 Activación / Invocación de Funciones y Procedimientos 6.3 Uso de Funciones y Procedimientos con Parámetros.
7	Graficación	7.1 Conceptos básicos de gráficas. 7.2 Introducción básica de manejo de gráficas. 7.3 Uso de librerías para la realización de graficas. 7.4 Graficación de un punto y líneas rectas 7.5 Graficación de Funciones Matemáticas en el plano cartesiano. 7.6 Introducción a la potencia grafica
8	Puertos	8.1 Conceptos generales de manejos de puertos 8.2 Configuración de los puertos en las computadoras 8.3 Uso de instrucciones para manejo de puertos

6.- APRENDIZAJES REQUERIDOS

- Uso básico de la computadora: Encendido y apagado, identificación de partes, manejo del teclado y mouse.
- Manejo básico de un sistema operativo: Identificación de archivos y directorios, creación, y borrado de archivos, creación y borrado de directorios. Exploración de archivos y directorios. Renombrar archivos y directorios. Comandos y ambiente gráfico del sistema operativo.

7.- SUGERENCIAS DIDÁCTICAS

- Propiciar la búsqueda y selección de información de los temas del curso.
- Exposición verbal auxiliada de materiales de apoyo.
- Fomentar la solución de problemas específicos mediante algoritmos y pseudocódigos (en forma autónoma y dirigida)
- Fomentar la implementación de programas, en forma autónoma, mediante prácticas.
- Promover en el alumno la realización de reportes y exposiciones auxiliado por software de presentaciones.
- Fomentar la implementación de programas orientados al manejo y uso del hardware.

8.- SUGERENCIAS DE EVALUACIÓN

- Considerar la participación en las actividades programadas de la materia:
 - Tareas y Ejercicios
 - Participación/Exposiciones en clase
 - Prácticas en clase
 - Prácticas extraclase
- Exámenes escritos y frente a la computadora
- Proyecto Final que involucre el manejo de hardware

9.- UNIDADES DE APRENDIZAJE

Unidad 1: Introducción a la Computación

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno identificará los conceptos básicos de la computación actual, su impacto en la sociedad y los diferentes tipos de programación	1.1 Investigar los antecedentes de la computación actual, sus orígenes hasta llegar a las computadoras y software actuales. Expondrá sus resultados en forma grupal. 1.2 Analizar el impacto de las computadoras en la sociedad mediante lecturas propuestas por el maestro. 1.3 Investigar los conceptos básicos de programación como lo son: Lenguaje de alto y bajo nivel, ensamblador, lenguaje máquina, Compilador, Traductor, Sistemas numéricos, Arq. de computadoras y sistemas operativos. 1.4 Investigar los diferentes tipos de programación: Programación estructurada, Programación orientada a objetos y programación de descripción de hardware. Analizar sus características y sus diferentes aplicaciones. 1.5 Comparar los diferentes lenguajes de programación y sus aplicaciones.	1 2 3

Unidad 2: Diseño de Algoritmos

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno aplicará diagramas de flujo y pseudocódigos.	2.1 Identificar los conceptos básicos de diseño de algoritmos tales como la representación de los datos, estructuras, bucles, arreglos así como la utilidad de un buen diseño algorítmico. 2.2 Aplicar una metodología de programación para la resolución de ejercicios propuestos. 2.3 Utilizar diagramas de flujo para la resolución de ejercicios propuestos. 2.4 Aplicar pseudocódigos en la resolución de ejercicios propuestos.	1 2 3

Unidad 3: Fundamentos del Lenguaje

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno diseñará, editará, compilará y ejecutará programas básicos.	3.1 Investigar el desarrollo histórico del lenguaje estructurado y discutirá sus resultados en forma grupal.	
	3.2 Identificar la estructura general de un programa en lenguaje estructurado.	
	3.3 Identificar las palabras reservadas del lenguaje y su utilidad y la representación básica de datos.	1
	3.4 Aplicar la declaración de variables, Constantes, Procedimientos y Funciones en la programación estructurada.	6
	3.5 Utilizar las operaciones aritméticas, lógicas y relacionales en la programación de ejercicios prácticos propuestos.	7
	3.6 Diseñar, Editar, compilar y ejecutar programas de ejercicios propuestos.	8

Unidad 4: Estructuras de Decisión y Control

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno programará y aplicará las estructuras de decisión y control en programas.	4.1 Utilizar estructuras secuenciales en la programación de ejercicios propuestos.	1
	4.2 Utilizar estructuras de transferencias condicionales e incondicionales en la programación de ejercicios propuestos.	6
	4.3 Utilizar estructuras de repetitivas en la programación de ejercicios propuestos.	7

Unidad 5.- Arreglos y Archivos

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno aplicará arreglos y archivos en programas.	5.1 Utilizar los arreglos (vectores y matrices) en programación estructurada por medio de ejercicios propuestos.	1 6
	5.2 Identificar los diferentes tipos de archivos , programarlos y aplicarlos en la solución de problemas	7 8

Unidad 6: Módulos

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno aplicará funciones y procedimientos en programas.	6.1 Identificar la diferencia entre funciones y procedimientos participando en discusión grupal.	1 6
	6.2 Aplicar la activación / invocación de funciones y procedimientos en ejercicios propuestos.	7
	6.3 Utilizar funciones y procedimientos en programas.	8

Unidad 7: Graficación

Objetivo de la Unidad	Actividades de Aprendizaje	Bibliografía
El alumno utilizará los conceptos básicos de la graficación.	7.1 Investigar los conceptos básicos de gráficas en programación estructurada.	6
	7.2 Aplicar los conceptos básicos de gráficas en programas.	7
	7.3 Aplicar librerías para gráficas.	8
	7.4 Graficar puntos y líneas rectas.	8
	7.5 Graficar funciones matemáticas en el plano cartesiano.	9
	7.6 Elaborar un programa que active y desactive la potencia gráfica.	10

Unidad 8: Puertos

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El alumno diseñará e implementará programas con manejo de puertos.	8.1 investigar los conceptos generales de manejo de puertos en la programación estructurada.	4
	8.2 Analizar la configuración de los puertos paralelo y serial en la computadora.	
	8.3 Diseñar e implementar un proyecto que involucre el manejo de puertos en la computadora.	6

10. FUENTES DE INFORMACIÓN

1. G. Levine G.
Introducción a las computadoras y programación estructurada
Ed. Mc Graw Hill
2. Peter Norton
Introducción a la computación
Ed. Mc. Graw Hill
3. Robert Sedgewick
Algorithms in C++
Ed. Addison Wesley
4. M. Morris Mano
Arquitectura de computadoras
Ed. Prentice Hall
5. O. J. Dahl, E. W. Dijatra, C.A.R. Huare
Programación estructurada.
Ed. Tiempo contemporáneo
6. B. W. Kernighan y D. M. Richie
Lenguaje de programación C
7. Byron S. Gottfried
Programación en C.
Ed. Mc Graw Hill
8. Herbert Schildt
C manual de referencia
Ed. Mc Graw Hill

9. Jamsa
Lenguaje C bibliotecas de funciones
Ed. Mc Graw Hill
10. Lee Adams
Programación avanzada de gráficos en C para windows
Ed. Graw Hill

10 a .- SOFTWARE PROPUESTO

- C++
- C

11. PRÁCTICAS

- 1.- Elaboración de un programa que obtenga el promedio de n datos. El programa debe pedir el número de datos, después los datos. Realizar su programación.
- 2.- Elaboración de un programa de conversión de unidades de medida. Se deberán pedir los datos correspondientes y deberá mostrar el resultado de la conversión.
- 3.- Elaboración de un programa de almacenamiento de datos tipo agenda. Nombre, dirección, teléfono, email y un campo de comentarios.
- 4.- Del programa realizado en el número 3, elaboración de un programa que introduciendo opciones de búsqueda de datos: Por nombre o por teléfono.
- 5.- Elaboración de un programa que pida una secuencia de caracteres y que los despliegue en forma invertida como resultado.
- 6.- Elaboración de un programa que resuelva la ecuación general de segundo grado utilizando la formula general mediante programación de funciones.
- 7.- Realice, utilizando procedimientos, el control básico del mouse.
- 8.- Realice, utilizando procedimientos, el control básico de la impresora.
- 9.- Elaboración de un programa de manejo de gráficos que permita dibujar figuras geométricas.
- 10.- Elaboración de un programa de manejo de gráficos que permita dibujar funciones matemáticas.
- 11.- Elaboración de un programa que maneje menús.
- 12- Manejo de un grupo de leds conectados al puerto serial de la computadora que enciendan en forma secuencial, esta secuencia la dará el programa,
- 13.- Manejo de un grupo de leds conectados al puerto paralelo de la computadora que enciendan en forma semejante a un semáforo. Programe la secuencia de cambio de color.