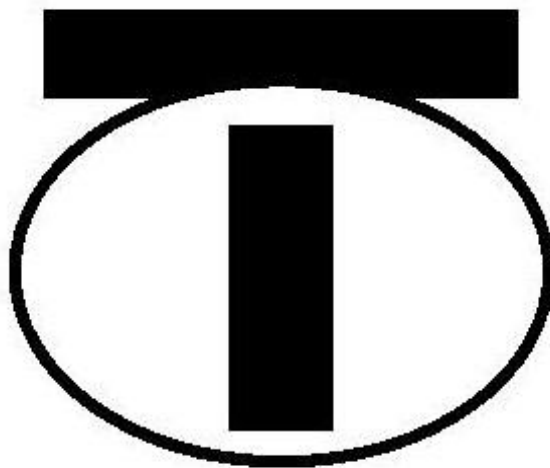




UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
COLEGIO DE CIENCIAS Y HUMANIDADES  
DEPARTAMENTO DE OPCIONES TÉCNICAS



Programa de la Opción Técnica  
**Sistemas computacionales:**  
*Desarrollo de software*



## **1 Concepción de la Opción Técnica programación.**

### **1.1 Antecedentes del Programa de la Opción Técnica Programación.**

En el año de 1979, el Consejo Académico de Opciones Técnicas aprueba el programa de la Opción Técnica de Programación: Lenguaje Cobol, la que comienza a impartirse en el semestre 1980-1 hasta el semestre 1987-1.

En el semestre 1987-1, comienza la aplicación del Plan Piloto de la Opción Técnica en Computación para la Administración, por considerarse que este programa estaba más acorde con el desarrollo reciente de la computación y con las necesidades de formación de recursos humanos detectados en el área.

En el semestre 1989-1 se aplica el programa denominado Opción Técnica en Computación con acuerdo del Director de la Unidad Académica del Ciclo de bachillerato del Colegio de Ciencias y Humanidades.

Este programa se estructuro una vez terminado y analizado el Plan Piloto iniciado en 1987-1, por la Profa. Evelina Chivley.

El programa que se llevo en la Opción Técnica en Computación durante agosto de 1994 a 1997, fue el resultado de la revisión efectuada por Concepción Rojas Hernández y Margarita Huerta Martínez, en 1994.

A partir de 1998 se aplica el nuevo programa para la Opción Técnica en Computación, la cual cambia también de nombre a "Sistemas Computacionales". El nuevo programa es el resultado de una revisión hecha por el conjunto de profesores que imparten la opción al programa anterior, por lo que se actualizaron algunos temas y se aumentaron otros, esto con el afán de ponerse al día y adecuarse al nuevo plan de estudios del colegio y así continuar con la labor que se venía realizando con los alumnos del colegio,

## **1.2 Modificación, conservación o creación de la Opción Técnica Programación.**

La computación juega un papel muy importante en la formación de los alumnos para su desarrollo académico, por lo que las Opciones Técnicas contemplan dos aspectos muy importantes el propedéutico, que le servirá para la continuación de sus estudios superiores y el terminal, el cual lo capacitara para incorporarse al campo laboral.

Los avances tecnológicos y en especial en el área de computación son muy rápidos, lo que ha provocado la necesidad de adecuar el programa de la opción ya varias veces, siendo este, otro momento para ello. La opción estará compuesta de dos fases, incorporando al programa el análisis y diseño de sistemas, además de una introducción a los lenguajes visuales. Se mantuvo el lenguaje de programación "C", con algunos reajustes a su secuencia y se elimino la parte referente a paqueterías (software de aplicación), ya que esta opción esta enfocada al desarrollo de sistemas por medio de un lenguaje de programación.

## **1.3 Contribución de los objetivos de la Opción Técnica Programación al perfil del egresado.**

En el proceso educativo, la enseñanza de las nuevas tecnologías contribuye a la formación de adolescente mediante el desarrollo de conocimientos, habilidades, destrezas, y actitudes en ámbitos que le permitirán desempeñarse como mejores alumnos, individuos y personas incorporadas en el campo laboral. Además de brindarles una visión de la que podría ser su futuro profesional.

Las opciones técnicas además de lo antes mencionado, coadyuvan en el desarrollo de destrezas intelectuales, en la elaboración de sus formas de pensamiento y la adquisición de valores y normas que le permitirá evaluar más el conocimiento científico, desempeñándose con mayor seguridad y autoestima, realizar aprendizajes independientes e interrelacionar los diferentes campos del saber para aplicarlos en la sociedad en la cual se desempeñe.

## **2 Concepción de la Opción Técnica programación.**

### **2.1 Contenidos disciplinarios seleccionados para la enseñanza en el bachillerato.**

Con el uso de las herramientas computacionales en todos los campos del quehacer humano es indispensable que un bachiller adquiera los conocimientos que le permitan utilizar estas nuevas tecnologías en el manejo, análisis, procesamiento y automatización de la información, de la manera más adecuada y eficaz posible.

La Opción Técnica de Programación constituye la opción que permite al alumno formarse en términos laborales y académicos para poderse integrar en el campo laboral a la par que estudiar o en definitiva integrarse al campo laboral; o en otro caso, adquirir los conocimientos necesarios que le permitan decidir la carrera a estudiar a nivel superior, teniendo con esta Opción Técnica un complemento en su formación profesional. Todo lo anterior a través del estudio del análisis y desarrollo de sistemas, así como de su implementación en un lenguaje de programación de alto nivel, además de tener una introducción a la programación orientada a objetos mediante un lenguaje visual.

### **2.2 Profundidad del tratamiento.**

Los contenidos deberán ser abordados con un nivel medio de complejidad, considerando que no existe ninguna preparación previa, pero conforme el curso vaya avanzando el grado de complejidad deberá también avanzar hasta alcanzar casos que el alumno se pueda encontrar en el campo laboral.

Los tratamientos teóricos superfluos o demasiado extensos, deberán evitarse y deberán ser preferentemente prácticos. Se cuidará, sin embargo, que el estudiante entienda las relaciones entre los conceptos y la práctica, lo mismo que la estructuración de la programación.

Los problemas propuestos en clase, se recomienda; no en todos los casos; primeramente se ejemplifiquen por el profesor y que después el alumno plantee la solución a otros problemas similares, siempre con miras en lo posible a dar solución a problemas que ejemplifiquen casos reales.

### **2.3 Enfoque disciplinario.**

Respecto al enfoque desde el cual se abordará el estudio de estos contenidos en la Opción Técnica de Programación, es conveniente destacar que deberá mostrarse al alumno que el enfoque del programa consiste en la solución de problemas mediante su análisis, y el diseño de un sistema que me permita automatizarlo, valiéndose de una herramienta computacional en su implementación, sin que esta última sea la finalidad del curso, si no todo el proceso que le permitirá decidir cual es la herramienta computacional adecuada para su implementación, y así usar dichas herramientas con la certeza que es la adecuada para la implementación de su diseño.

## **3 Enfoque didáctico de la Opción Técnica programación.**

### **3.1 Formas de trabajo en el aula.**

- El profesor deberá propiciar la socialización del trabajo entre los estudiantes, motivándolos a la investigación y permitiéndoles la discusión en equipo y en grupo de las diversas ideas.
- Deberán diseñarse numerosas actividades para proporcionar a los alumnos una visión sobre la computadora y sus usos, a través de la adquisición de conocimientos básicos sobre las diferentes metodologías de programación, empleadas en la solución de diversos problemas.
- Se mostrara la programación como un medio donde se manejan los diferentes elementos que intervienen en un problema y las relaciones entre estos; es decir, la programación permite manejar un modelo matemático que representa el problema, para la planeación de su solución, así como el lenguaje formal, preciso y exacto, necesario para describir en la computadora las acciones a realizar para resolverlos. Es de particular importancia transmitir al alumno el panorama general de la resolución de problemas al utilizar la computadora y los lenguajes de programación como herramientas de apoyo.
- Para que los estudiantes participen de manera activa en las diferentes etapas de la elaboración de programa, es conveniente abordar la resolución de problemas cercanos a lo cotidiano, para posibilitar la aplicación de conocimientos y la interacción con la computadora y extender paulatinamente el campo de aplicación a problemas

mas complejos, de tipo científico, técnico, administrativo y social, de preferencia en relación con otras asignaturas del plan de estudios, de modo que los alumnos adquieran habilidades para analizar, modelar y resolver problemas por medio de la computadora, así como para estructurar sus razonamientos de manera lógica.

- Muchos contenidos deberán tratarse a través de ejemplos. El profesor no debe pretender dar la teoría de todos los ejemplos presentados; sí, en cambio, debe proporcionar numerosas oportunidades para que el estudiante investigue y practique los procedimientos básicos y equilibre adecuadamente este aspecto con los desarrollos conceptuales. Para ello las nociones y procedimientos deberán introducirse mediante actividades y problemas que los doten de significado; las precisiones teóricas deberán establecerse cuando los alumnos dispongan de la experiencia y los ejemplos suficientes para garantizar su comprensión.
- Cuando en el curso sean requeridos temas vistos con anterioridad por los estudiantes, el profesor deberá tomar nota de lo olvidado y diseñar actividades para recuperar destrezas y avanzar en la adquisición de nuevos conocimientos.
- Es conveniente mantenerse dentro de los tiempos propuestos para el tratamiento de las diversas unidades, para no mermar la enseñanza de los contenidos finales de cada fase. Los tiempos, sin embargo, podrán ser ajustados por el profesor, según la complejidad de los contenidos y el grado de madurez y conocimientos del grupo. Es importante, para ello, no pretender agotar un tema desde un principio, pues un mismo contenido puede ser retomado y enriquecido aumentando la posibilidad de los alumnos para comprender y profundizar los conocimientos adquiridos y evitar las explicaciones teóricas superfluas, distinguiendo con claridad lo que es importante en cada tema.

#### **4 Sugerencias de evaluación.**

La evaluación es permanente porque además de asignar una calificación nos permite determinar el grado de habilidades, conocimientos y destrezas que el alumno ha adquirido durante el curso y a través de eso poder transformar nuestra labor docente.

Se recomienda una evaluación diagnóstica para determinar el grado de conocimientos, habilidades y destrezas con las que ingresa el alumno. Utilizando para ello exámenes escritos, lluvia de ideas, etc.

Durante el desarrollo del curso el profesor realizara la evaluación formativa mediante exámenes escritos, prácticas, investigaciones, participaciones en clase, asistencia.

En la evaluación sumativa se deben de considerar las evaluaciones diagnóstica y formativa, además de un proyecto que el alumno desarrollara en el segundo semestre de la opción, siendo el proyecto la elaboración de un sistema de computación que aborde el planteamiento y la solución de un problema donde utilice los conocimientos, habilidades y destrezas adquiridos a lo largo del curso. Este proyecto será evaluado por el profesor.

Se recomienda una evaluación periódica, tanto práctica como teórica, que tenga por objeto, por una parte, informar al alumno el grado de conocimientos alcanzados, permitiendo detectar errores en el aprendizaje y, por otra parte, asignar calificaciones. La evaluación estará basada en pruebas escritas, en los ejercicios en la computadora y trabajos objetivos acordes con la metodología del programa de estudios. Sin olvidar que el tiempo asignado en cada unidad incluye las actividades de evaluación.

#### **Perfil profesiográfico del docente.**

**Académicos:** Tener título superior al de bachiller en las licenciaturas de ingeniería en computo, informática, sistemas o matemáticas aplicadas y computación, ingeniería, actuaría y licenciatura en matemáticas o carreras afines cuando así lo determine el instructivo sobre agrupamiento de materias y otros reglamentos que expida el Consejo Técnico del Bachillerato a este respecto. Poseer sólidos conocimientos en la materia y sus fundamentos, así como tener una perspectiva del papel de la misma en el desarrollo de las ramas actuales de la computación.

**Profesionales:** Preferentemente tener experiencia en el campo laboral.

**Docente:** Para profesores de nuevo ingreso, haber acreditado dentro de los programas institucionales de formación de profesores los cursos instituidos como introducción al modelo educativo del colegio, a la didáctica del área y a los contenidos de la asignatura, de preferencia en forma de diplomado.

Para los profesores en ejercicio, acreditar el manejo de los contenidos y enfoques del programa de la opción, dentro del programa institucional de formación de profesores.

**Perfil del egresado:** El egresado de esta opción estará capacitado para desarrollar sistemas computacionales que den solución a problemas que implique la automatización de procesos, dominará un lenguaje de programación \* que le permitirá implementar en la computadora los sistemas desarrollados, además de contar con las bases de la programación en lenguajes visuales.

#### **Competencias desarrolladas en el alumno:**

- Clasificar los \* problemas en aquellos cuya solución pueda ser automatizada utilizando la computadora o alguna otra herramienta.
- Recabar la información necesaria referente a un problema que le permita definir adecuadamente los datos de entrada y salida, para el planteamiento de su solución.
- Utilizar las técnicas como la programación estructurada y modular en el desarrollo de programas de cómputo, que permitan plantear la solución adecuada a un problema.
- Extrapolar la solución de un problema a un modelo que le permita la sistematización del mismo.
- Discernir las diferentes herramientas como son software, calculadora, papel, etc., que se utilizan en la automatización de los sistemas, para emplear la más adecuada.
- Efectuar la instalación del software necesario para el uso del lenguaje de programación en la computadora.
- Configurar el software necesario para el uso del lenguaje de programación para trabajar de acuerdo a requerimientos específicos.
- Emplear un lenguaje de programación \*, para automatizar(programar) los sistemas en una computadora.
- Utilizar los medios de almacenaje (discos duros, disquete), para extraer y guardar datos utilizados en los programas.
- Desarrollar aplicaciones en ambientes gráficos desde ambiente externo.
- Aplicar algunas técnicas de ordenación y clasificación de datos
- Ser capaz de diseñar sistemas de manejo de archivos que implique el uso de los procedimientos típicos de altas, bajas, cambios, consultas y reportes.
- Utilizar un lenguaje visual para darle una interfaz más amigable a los programas.



- Diseñar aplicaciones por eventos para un ambiente gráfico.

### **OBJETIVO DE LA OPCION**

Esta opción fue modificada con la intención de poder satisfacer el vertiginoso avance de las nuevas tecnologías computacionales y las necesidades y requerimientos del campo laboral y de los alumnos. Se modifica a 2 semestres después de estarse impartiendo en 3 semestres, pues se veía que el tiempo estaba siendo un factor importante en los valores de egreso.

**UNIDAD UNO**  
**ANÁLISIS Y DISEÑO DE SISTEMAS**

**Objetivo de la Unidad:** Comprenderá los conceptos referentes al análisis y diseño de sistemas para aplicarlos en el desarrollo y mantenimiento de sistemas.

| HORAS     | TEMÁTICA   | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-<br>APRENDIZAJE   |
|-----------|--|---|--|
| <b>18</b> | 1) Cambios al sistema razones y resistencias<br>a) Motivos para el cambio.<br>b) Enfrentar el cambio.<br>c) Resistencia ante el cambio de un sistema.<br><br>2) Definición del problema<br>a) El estudio de definición del problema.<br>b) El escrito del estudio del sistema.<br><br>3) Análisis del sistema<br>a) Recopilación de información. | El alumno: <ul style="list-style-type: none"> <li>• Explicará por que cambian los sistemas de información y por que la gente se resiste al cambio.</li> <li>• Describirá los procedimientos que pueden seguir para la definición del problema.</li> <li>• Describirá como se recolectan los datos y después como se analizan durante la etapa de análisis del sistema.</li> </ul> | <ul style="list-style-type: none"> <li>• Mostrar el por que de la resistencia al cambio.</li> <li>• Seleccionar un problema y aplicar los pasos indicados para su solución.</li> <li>• Utilizar las herramientas del diseño de sistemas como algoritmos, diagramas de flujo, etc., para documentar el problema.</li> <li>• Después de terminado el sistema implantarlo y revisar si</li> </ul> |

| HORAS | TEMÁTICA   | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-<br>APRENDIZAJE   |
|-------|--|---|--|
|       | <p>b) Análisis del problema.<br/>c) El informe del análisis.</p> <p>4) Diseño del sistema</p> <p>a) determinación de opciones.<br/>b) herramientas y técnicas de diseño.<br/>c) Algoritmos, diagramas de flujo y pseudocódigos.<br/>d) Selección de una opción.<br/>e) Pruebas de escritorio.<br/>f) El informe de un diseño.</p> <p>5) Implementación del sistema</p> <p>a) Consecuencias de anteriores instalaciones.<br/>b) Instalación del equipo.<br/>c) Conversión y cambio del sistema.</p> <p>6) Revisión posterior a la implementación.</p> | <ul style="list-style-type: none"> <li>• Analizará algunos problemas que puedan afectar las decisiones del diseño del sistema.</li> <li>• Resumirá los tipos de especificaciones del sistema que se producen durante la etapa de diseño.</li> <li>• Comprenderá los factores que se deben tomar en cuenta al adquirir un nuevo equipo.</li> <li>• Practicará algunas de las actividades propias de la implementación y sustitución del sistema</li> </ul> | <p>existen o no correcciones.</p> <ul style="list-style-type: none"> <li>• Estructurar un calendario de mantenimiento.</li> <li>• Diseñar los manuales de usuario.</li> <li>• Analizar diferentes problemas haciendo notar que algunos pueden no tener solución, única solución o múltiples soluciones y cual es la mejor.</li> <li>• Diseñar formatos de pantalla y de recopilación de datos</li> </ul> |

**UNIDAD DOS**  
**ESTRUCTURAS BÁSICAS DEL LENGUAJE “C”**

**Objetivo de la Unidad:** Utilizará el entorno integrado de desarrollo, así como las estructuras básicas que le permitirán iniciarse en la elaboración de programas mediante el uso de lenguaje “C”.

| HORAS     | TEMÁTICA  | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE   |
|-----------|---|---|--|
| <b>18</b> | 1) El entorno integrado de desarrollo<br>a) Instalación.<br>b) Arranque.<br>c) Editor.<br>d) Almacenamiento en archivo.<br>e) Compilador.<br>f) Recuperación de archivo.<br>g) Salir.<br><br>2) Conceptos fundamentales<br>a) Variables y Constantes.<br>b) Palabras reservadas.<br>c) Identificadores.<br>d) Estructura básica de un programa. | El alumno:<br><br>• Utilizará el entorno integrado de desarrollo de lenguaje “C”.<br><br>• Señalará los conceptos fundamentales de la programación en lenguaje “C”.<br><br>• Empleará la estructura de los programas en lenguaje “C”.<br><br>• Utilizará los tipos de datos básicas.<br><br>• Aplicará el concepto de constantes, variables y datos enumerados. | • Comparar el entorno integrado de desarrollo de “C” con los entornos de paqueterías de uso común como puede ser Word.<br><br>• Enfatizar cada una de las etapas, planeación, creación de algoritmo, codificación, edición, almacenamiento, compilación y ejecución de un programa.<br><br>• Identificar los elementos básicos de sintaxis y semántica del lenguaje “C”.<br><br>• Identificar las diferentes zonas |

| HORAS | TEMÁTICA   | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-------|--|---|---|
|       | <p>3) Tipos de datos y tamaño</p> <ul style="list-style-type: none"> <li>a) Sistemas numéricos. <ul style="list-style-type: none"> <li>i. Binario.</li> <li>ii. Hexadecimal.</li> </ul> </li> <li>b) int.</li> <li>c) float.</li> <li>d) fhar.</li> <li>e) Apuntadores.</li> </ul> <p>4) Declaración de datos</p> <ul style="list-style-type: none"> <li>a) Constantes.</li> <li>b) Variables.</li> <li>c) Enumerados.</li> </ul> <p>5) Operadores</p> <ul style="list-style-type: none"> <li>a) Aritméticos.</li> <li>b) Relacionales.</li> <li>c) Lógicos.</li> <li>d) Conversiones.</li> <li>e) Lógicos para el manejo de bits.</li> <li>f) Asignación.</li> <li>g) Prioridad y orden de evaluación.</li> </ul> | <ul style="list-style-type: none"> <li>• Utilizará los operadores del lenguaje así como su prioridad y orden de evaluación.</li> <li>• Utilizará las funciones de entrada y salida estándar para crear sus primeros programas.</li> </ul> | <p>de un programa en "C".</p> <ul style="list-style-type: none"> <li>• Mediante la resolución de problemas, introducir los conceptos fundamentales para la programación.</li> <li>• Aplicar todos los conceptos teóricos que se expongan al alumno, mediante la solución de problemas.</li> <li>• Retomar los ejercicios de creación de sistemas para su programación.</li> </ul> |

| HORAS | TEMÁTICA  | OBJETIVOS EDUCATIVOS | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE |
|-------|---|----------------------|--------------------------------------|
|       | 6) Rutinas de entrada y salida<br>a) Funciones de entrada estándar.<br>b) Funciones de salida estándar. |                      |                                      |

**UNIDAD TRES**  
**SENTENCIAS DE CONTROL**

**Objetivo de la Unidad:** Distinguirá las distintas formas de selección e iteración, y utilizará estas misma para la creación de programas indicando selectividad y repetición de sentencias.

| HORAS     | TEMÁTICA   | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-----------|--|---|---|
| <b>24</b> | 1) De selección<br>a) if - else.<br>b) switch.<br>c) Sentencia break.<br><br>2) De iteración<br>a) for.<br>b) while.<br>c) do. | El alumno: <ul style="list-style-type: none"> <li>• Identificará los distintos tipos de sentencias de selección e iteración.</li> <li>• Manipulará las sentencias de selección e iteración para la elaboración de programas.</li> </ul> | <ul style="list-style-type: none"> <li>• Desarrollar programas donde sólo se utilice la sentencia If – else y realizar versiones posteriores sólo utilizando la sentencia switch, marcando ventajas y desventajas de cada una de ellas.</li> <li>• Desarrollar programas donde</li> </ul> |

| HORAS | TEMÁTICA | OBJETIVOS EDUCATIVOS   | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-------|----------|--|---|
|       |          | <ul style="list-style-type: none"> <li>Utilizará de manera adecuada cada una de las sentencias de selección e iteración.</li> <li>Conocerá la importancia del uso de la sentencia break en las sentencias de selección e iteración.</li> </ul> | <p>sea necesario utilizar los distintos tipos de sentencias de iteración.</p> |

#### UNIDAD CUATRO FUNCIONES

**Objetivo de la Unidad:** Aplicará el concepto de modularidad para la elaboración de programas extensos o complejos.

| HORAS     | TEMÁTICA   | OBJETIVOS EDUCATIVOS   | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-----------|--|--|---|
| <b>30</b> | <p>1 Modularidad</p> <p>a) Concepto.</p> <p>b) Importancia.</p> <p>2 Funciones</p> | <p>El alumno:</p> <ul style="list-style-type: none"> <li>Describirá la importancia de estructurar programas en forma modular.</li> </ul> | <ul style="list-style-type: none"> <li>Trabajar con problemas que propicien la necesidad de utilizar el concepto de programación modular como una estrategia para la</li> </ul> |

| HORAS | TEMÁTICA  | OBJETIVOS EDUCATIVOS   | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-------|---|--|---|
|       | <p>a) Definición.</p> <p>    i) Cuerpo de la función.</p> <p>    ii) Valor retornado por una función.</p> <p>b) Llamada a una función.</p> <p>c) Declaración de una función.</p> <p>d) Paso de parámetros.</p> <p>3 Librerías</p> <p>    a) Definición.</p> <p>    b) Funciones estándar.</p> <p>4 Variables</p> <p>    a) Externas.</p> <p>    b) Estáticas.</p> <p>    c) Registro.</p> <p>5 Funciones Gráficas</p> <p>    a) Modos de video.</p> <p>    b) Inicialización.</p> <p>    c) Primitivas.</p> | <ul style="list-style-type: none"> <li>• Desarrollará programas utilizando el concepto de modularidad por medio de las funciones.</li> <li>• Utilizará paso de parámetros por valor o referencia en las funciones.</li> <li>• Utilizará las funciones predefinidas en las librerías de "C".</li> <li>• Identificará la diferencia de utilizar variables externas, estáticas y de registro.</li> <li>• Distinguirá entre los distintos modos de video en los que trabaja "C" y la inicialización de cada uno.</li> <li>• Realizará programas que corran en el modo gráfico de "C".</li> </ul> | <p>resolución de problemas complejos o extensos.</p> <ul style="list-style-type: none"> <li>• Utilizar las librerías, funciones estándar y diferentes tipos de variables que contienen las definiciones y declaraciones que se emplea en todo programa.</li> <li>• Elaborar programas utilizando el modo gráfico y sus principales comandos (primitivas), para la elaboración de desplegados gráficos.</li> </ul> |



**UNIDAD CINCO**  
**DATOS ESTRUCTURADOS**

**Objetivo de la Unidad:** Generará diferentes tipos de datos compuestos, para optimizar el manejo de datos y aplicará los métodos de ordenamiento y búsqueda.

| HORAS     | TEMÁTICA   | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-----------|--|---|---|
| <b>22</b> | <p>1 Arreglos</p> <p>a) Cadenas de caracteres.</p> <p>b) Matrices.</p> <p>c) Arreglos multidimensionales.</p> <p>2 Estructuras</p> <p>a) Declaración.</p> <p>b) Manipulación.</p> <p>c) Clases.</p> <p>3 Ordenamiento y búsqueda</p> <p>a) Método de la burbuja.</p> <p>b) Métodos de búsqueda secuencial y binaria.</p> | <p>El alumno:</p> <ul style="list-style-type: none"> <li>• Empleará la compatibilidad de tipos de datos optimizando los recursos del hardware (memoria).</li> <li>• Desarrollará programas utilizando el concepto de arreglos.</li> <li>• Desarrollará programas utilizando el concepto de estructuras.</li> <li>• Desarrollará programas utilizando las técnicas de ordenación de datos, para obtener información rápidamente y bien clasificada.</li> </ul> | <ul style="list-style-type: none"> <li>• Recabar por medio de una lluvia de ideas que se entiende por arreglos.</li> <li>• Explicar el concepto de arreglos</li> <li>• Investigar el concepto de arreglos para reafirmar su conocimiento.</li> <li>• Ejemplificar el concepto de arreglos.</li> <li>• Elaborar programas que involucren el concepto de</li> </ul> |

|  |  |  |                          |
|--|--|--|--------------------------|
|  |  | <ul style="list-style-type: none"> <li>• Aplicarás los métodos de búsqueda en la recuperación de datos.</li> </ul> | ordenamiento o búsqueda. |
|--|--|--|--------------------------|

## UNIDAD SEIS ARCHIVOS

**Objetivo de la Unidad:** Manejará los mecanismos de creación y acceso a archivos de tipo texto y binario, de manera secuencial o aleatoria.

| HORAS     | TEMÁTICA  | OBJETIVOS EDUCATIVOS   | ESTRATEGIAS DE ENSEÑANZA-<br>APRENDIZAJE   |
|-----------|---|--|--|
| <b>36</b> | 1 Archivos<br>a) Tipo<br>i) Modo Texto.<br>ii) Modo Binario.<br>b) Apertura<br>i) fopen().<br>c) Cierre<br>i) fclose().<br><br>2 Escritura y Lectura<br>a) Entrada y salida | El alumno: <ul style="list-style-type: none"> <li>• Identificará los archivos de tipo texto y binario.</li> <li>• Utilizará la apertura y cierre de un archivo.</li> <li>• Manejará las funciones que le permitan realizar un flujo de archivo para entrada, salida y</li> </ul> | <ul style="list-style-type: none"> <li>• Recomendar una exposición de la idea central por medio de un modelo.</li> <li>• Seleccionar algún programa que muestre la creación, apertura y cierre de un archivo.</li> <li>• Con relación al apertura y cierre de archivos se sugiere cuidar que los alumnos:</li> </ul> |

| HORAS | TEMÁTICA  | OBJETIVOS EDUCATIVOS   | ESTRATEGIAS DE ENSEÑANZA-<br>APRENDIZAJE   |
|-------|---|--|--|
|       | <p>formateado.</p> <p>i) fscanf().</p> <p>b) Entrada y salida de carácter.</p> <p>i) fgetc().</p> <p>ii) fgets().</p> <p>iii) fputc().</p> <p>iv) fputs().</p> <p>c) Entrada y salida directa.</p> <p>i) fread().</p> <p>ii) fwrite().</p> <p>d) Detección del fin.</p> <p>i) feof().</p> <p>3 Acceso secuencia y aleatorio</p> <p>a) Funciones secuenciales.</p> <p>i) ftell().</p> <p>ii) rewind().</p> <p>b) Función aleatoria.</p> <p>i) fseek().</p> | <p>ambos.</p> <ul style="list-style-type: none"> <li>• Manejará las funciones básicas que le permitan escribir y leer datos en los archivos.</li> <li>• Identificará las condiciones de error que pueden causar el uso de archivos.</li> <li>• Aplicará en el manejo archivo los conceptos de acceso secuencial o aleatorio.</li> <li>• Realizará programas que involucren las actividades propias del manejo de archivos como son altas, bajas cambios consultas y reportes.</li> </ul> | <ul style="list-style-type: none"> <li>✓ Sepan abrir y cerrar un archivo.</li> <li>✓ Manejar el modo en el cual se quiere abrir el archivo.</li> <li>✓ Ver el contenido de un archivo.</li> <li>✓ Diferencien entre modo texto y binario.</li> </ul> <ul style="list-style-type: none"> <li>• Con relación al acceso secuencial y aleatorio se sugiere cuidar que los alumnos: <ul style="list-style-type: none"> <li>✓ Acceso secuencial y aleatorio de lectura y escritura.</li> <li>✓ Control sobre el indicador de posición de un flujo con la función fseek().</li> </ul> </li> <li>• El profesor mostrará las funciones de entrada y salida, de tipo carácter, directa y formateada.</li> <li>• Aplicar los conceptos de creación de archivos en la elaboración de programas.</li> </ul> |

**UNIDAD SIETE**  
**LENGUAJE VISUAL “INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA A EVENTOS”**

**Objetivo de la Unidad:** Se introducirá en el ambiente de la programación orientada a eventos

| HORAS     | TEMÁTICA  | OBJETIVOS EDUCATIVOS  | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE   |
|-----------|---|---|--|
| <b>32</b> | <p>1 El entorno del lenguaje visual</p> <p>a) Las características.</p> <p>b) Pantalla principal.</p> <p>    i) Barra de títulos.</p> <p>    ii) Barra de menús.</p> <p>    iii) Barra de herramientas.</p> <p>c) Ventana del proyecto.</p> <p>d) Menú de ayuda.</p> <p>2 Herramientas de programación y su funcionamiento</p> <p>a) La caja de herramientas.</p> <p>b) Creación de controles.</p> <p>    i) Propiedades del control.</p> <p>    ii) Procedimientos del control.</p> | <p>El alumno:</p> <ul style="list-style-type: none"> <li>• Obtendrá un panorama general de la organización de un proyecto en cualquier entorno propio de los lenguajes visuales.</li> <li>• Relacionará cada uno de los objetos con sus eventos correspondientes, de igual manera identificará propiedades exclusivas de acuerdo a la función del objeto en turno.</li> <li>• Solucionará problemas a través de elementos propios del lenguaje</li> </ul> | <ul style="list-style-type: none"> <li>• Mediante una exploración dirigida mostrar el entorno.</li> <li>• Identificar los elementos básicos de sintaxis y semántica.</li> <li>• Identificar las diferentes zonas de un programa en lenguaje visual.</li> <li>• Mediante la resolución de problemas, introducir los conceptos fundamentales del lenguaje visual.</li> </ul> |

| HORAS | TEMÁTICA  | OBJETIVOS EDUCATIVOS   | ESTRATEGIAS DE ENSEÑANZA-APRENDIZAJE  |
|-------|---|--|---|
| 3     | Programación de eventos<br>a) Diseño de interfaz.<br>b) Definición de variables y constantes.<br>c) Asignación de propiedades de un evento.<br>d) Control de flujo.<br>i) Sentencias de programación.<br>ii) Funciones.<br>e) Ejecución y depuración del código.<br>f) Depuración del formulario. | de programación, diseñado una interfaz legible que muestre resultados de manera sencilla | <ul style="list-style-type: none"> <li>• Retomar algunos ejercicios para realizar programas.</li> </ul> |

## **BIBLIOGRAFÍA:**

- Jhon G. Burch, Sistemas de información: Teoría y práctica, Editorial Limusa
- James A. Senn, Análisis y Diseño de Sistemas de información, McGraw Hill.
- Osvaldo Cairó, Metodología de la programación tomo I, AlfaOmega 1995.
- Francisco Javier Cevallos, Enciclopedia del lenguaje C, Alfaomega 2001
- Craig Arnush, Aprendiendo Borland C++ en 21 días, Prentice hall 1997.
- Luis Joyanes Aguilar, Curso de programación con turbo C++, McGraw Hill 1996.
- Gary Cornell, Manual del programador de Visual Basic 6.0, McGraw Hill, ed. 2001
- Marco Antonio Tiznado Santana, El camino Fácil a Visual Basic 5.0, McGraw Hill, ed. 2000