

1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura: Tecnologías de la Información
Carrera: Ingeniería Bioquímica
Clave de la asignatura: BQM - 0534
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 Tuxtepec del 17 al 21 de Enero de 2005	Representantes de las academias de Ingeniería Bioquímica.	Reunión Nacional de Evaluación Curricular de la Carrera de Ingeniería Bioquímica.
Institutos Tecnológicos de Celaya, Morelia Abril del 2005	Academia de Ingeniería Bioquímica.	Análisis y enriquecimiento de las propuestas de los programas diseñados en la reunión nacional de evaluación
Instituto Tecnológico de Tepic del 25 al 29 de abril del 2005	Comité de Consolidación de la carrera de Ingeniería Bioquímica.	Definición de los programas de estudio de la carrera de Ingeniería Bioquímica.

3.- UBICACIÓN DE LA ASIGNATURA

a). Relación con otras asignaturas del plan de estudio

Anteriores		Posteriores	
Asignaturas	Temas	Asignaturas	Temas
		Todas las asignaturas de la carrera	Elaboración de todo tipo de trabajos y proyectos en computadora
		Métodos Numéricos	

b). Aportación de la asignatura al perfil del egresado

- Programar en un lenguaje de alto nivel, para la solución de problemas de la ingeniería bioquímica.
- Fomentar el análisis lógico.
- Favorecer el desarrollo de una metodología sistemática de trabajo.
- Adquirir habilidad para trabajo en equipo.

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

Conocerá y comprenderá los fundamentos y estado actual de la informática, utilizando tanto el software de aplicación como el desarrollo de algoritmos computacionales codificados en un lenguaje de alto nivel, para la solución de problemas y cálculos propios de la ingeniería bioquímica.

5.- TEMARIO

1	Introducción a la Informática.	<ul style="list-style-type: none">1.1 Importancia y perspectivas de la computación.1.2 Impacto de las computadoras en la sociedad.1.3 Hardware.<ul style="list-style-type: none">1.3.1 CPU.1.3.2 Periféricos.1.4 Software.<ul style="list-style-type: none">1.4.1 Sistemas Operativos.1.4.2 Lenguajes de Programación: Compiladores.<ul style="list-style-type: none">1.4.2.1 Alto nivel: Visual Basic, FORTRAN 90, Pascal, entre otros.1.4.2.2 Bajo nivel: Ensamblador. hexadecimal, binario.1.4.3 Programas de aplicación: Principales aplicaciones: procesadores de texto, hojas electrónicas, manejadores de bases de datos, diseñadores de presentaciones, graficadores, software científico y de ingeniería entre otros.1.5 Maneras de operación de la computadora: Interrelación usuario-computadora.1.6 Redes.1.7 Internet.<ul style="list-style-type: none">1.7.1 Características e importancia.1.7.2 Aplicaciones: ftp, telnet, www, entre otras.1.7.3 Búsqueda de información. Robots buscadores.1.7.4 Seguridad de la información.<ul style="list-style-type: none">1.7.4.1 Antivirus, Firewall.1.7.4.2 Spywares.
---	--------------------------------	---

5.- TEMARIO (Continuación)

2	Sistemas operativos y programas de Aplicación.	<ul style="list-style-type: none">2.1 Entorno Windows:<ul style="list-style-type: none">2.1.1 Archivos y carpetas.2.1.2 Manejo de información.2.2 Entorno UNIX (Linux).2.3 Procesadores de textos: Características y aplicaciones.2.4 Diseñadores de presentaciones: Características y aplicaciones. Estrategias de desarrollo de presentaciones.2.5 Hojas electrónicas.<ul style="list-style-type: none">2.5.1 Características y aplicaciones2.5.2 Funciones de librería2.5.3 Graficación de datos
3	El ambiente de desarrollo de aplicaciones.	<ul style="list-style-type: none">3.1 Desarrollo histórico de un lenguaje estructurado.3.2 Desarrollo y características de los lenguajes visuales.3.3 El ambiente de desarrollo de la aplicación.<ul style="list-style-type: none">3.3.1 Descripción de los menús y las opciones de ambiente de desarrollo de aplicaciones.3.4 Estructura general de una aplicación.<ul style="list-style-type: none">3.4.1 Interfase y código.<ul style="list-style-type: none">3.4.1.1 Proyecto.3.4.1.2 Formularios.3.4.1.3 Módulos de código.3.5 Eventos.3.6 Objetos y propiedades.3.7 Herramientas para crear la interfase de la aplicación.<ul style="list-style-type: none">3.7.1 Formularios y módulos de código.3.7.2 Cajas de texto.3.7.3 Etiquetas.3.7.4 Botones de comando.3.7.5 Botones de opción.3.7.6 Casillas de verificación.3.7.7 Cuadros de imagen.3.7.8 Otras herramientas.

5.- TEMARIO (Continuación)

4	Introducción a la programación.	<ul style="list-style-type: none">4.1 Análisis de riesgos.<ul style="list-style-type: none">4.1.1 Establecimiento de un sistema de verificación 4.1 Metodología de la solución de problemas con la computadora.<ul style="list-style-type: none">4.1.1 Programación no estructurada4.1.2 Programación estructurada4.1.3 Programación modular4.1.4 Programación orientada a eventos4.1.5 Programación orientada a objetos.4.1.6 Programación por escenarios4.2 Algoritmos<ul style="list-style-type: none">4.2.1 Diagramas de bloque, flujo, simbología4.2.2 Pseudocódigo y codificación4.3 Elementos del lenguaje de programación<ul style="list-style-type: none">4.3.1 Palabras reservadas4.3.2 Tipos de datos básicos4.3.3 Declaración de variables, Constantes, Procedimientos y Funciones (subrutinas)4.3.4 Operaciones aritméticas, lógicas y relacionales.4.4 Uso del depurador (debbuger) para depurar y dar seguimiento a los programas.4.5 Diseño, edición, compilación y ejecución de aplicaciones.
---	---------------------------------	--

5.- TEMARIO (Continuación)

5	Funciones, procedimientos y estructuras de control.	5.1 Funciones y procedimientos. 5.1.1 Funciones estándar 5.2 Funciones definidas por el usuario. 5.2.1 Paso de parámetros por valor. 5.2.2 Paso de parámetros por referencia. 5.3 Estructuras de selección. 5.3.1 Selectiva simple. 5.3.2 Selectiva doble. 5.3.3 Selectiva anidada. 5.3.4 Selectiva múltiple. 5.4 Estructuras Repetitivas (Bucles). 5.4.1 Repetir mientras. 5.4.2 Repetir hasta. 5.4.3 Repetir desde. 5.5 Ejercicios de aplicación.
6	Arreglos, Archivos.	6.1 Arreglos. 6.1.1 Conceptos básicos. 6.1.2 Arreglos unidimensionales. 6.1.3 Arreglos bidimensionales. 6.1.4 Operaciones con vectores y matrices. 6.2 Archivos. 6.2.1 Archivos secuenciales. 6.2.2 Archivos de acceso directo. 6.2.3 Abrir, cerrar, acceder y guardar información en archivos. 6.3 Problemas de aplicación.

6.- APRENDIZAJES REQUERIDOS

- Conocimientos básicos acerca de las computadoras y del sistema operativo. Manejo de carpetas y de archivos.

7.- SUGERENCIAS DIDÁCTICAS

- Propiciar el uso del cañón electrónico para la enseñanza de:
 - El ambiente integrado de desarrollo del lenguaje seleccionado (IDE).
 - El depurador del IDE para mostrar el flujo de los programa en ejecución.
- Elaborar una guía de ejercicios para actividades extra clase
- Diseñar una guía de ejercicios para actividades extra clase
- Fomentar la participación y desempeño del alumno en el aula y el laboratorio
- Utilizar software que permita que el alumno interactúe con la computadora.
- Realizar proyectos de aplicación en las áreas de ingeniería bioquímica.
- Promover la participación de estudiantes como tutores

8.- SUGERENCIAS DE EVALUACIÓN

- Participación y desempeño del alumno en el aula y el laboratorio.
- Revisión de los programas elaborados.
- Cumplimiento de tareas y ejercicios.
- Trabajos de investigación.
- Prácticas.
- Desarrollo de un proyecto final que integre todas las unidades de aprendizaje.
- Exámenes escritos y frente a la computadora.

9.- UNIDADES DE APRENDIZAJE

UNIDAD 1.- Introducción a la Informática.

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
El estudiante comprenderá los conceptos básicos y las características de un sistema computacional, así como su evolución y su impacto en el desarrollo tecnológico.	<ul style="list-style-type: none">• Investigar en diversas fuentes sobre las tecnologías de la información.• Valorar los elementos que conforma un equipo computacional en software, hardware y Firmware.• Realizar una clasificación de los sistemas operativos, valorando sus actuales. Diferencias• Analizar y comprender los conceptos básicos de la paquetería de software.	1, 2, 3, 16

UNIDAD 2.- Sistemas operativos y programas de Aplicación.

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Manejará con soltura las funciones básicas del sistema operativo y los programas de aplicación de la ofimática.	<ul style="list-style-type: none">• Realizar una investigación en Internet y de otras fuentes acerca de la evolución de los sistemas operativos.• Elaborar una tabla comparativa con ventajas y desventajas de los lenguajes de bajo y alto nivel.	1,2,3,10,16

UNIDAD 3.- El ambiente de desarrollo de aplicaciones.

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
Conocerá el ambiente de desarrollo para la creación de aplicaciones y obtendrá la capacidad de crear interfases de aplicaciones de variada complejidad.	<ul style="list-style-type: none">• Buscar la información necesaria para instalar y configurar el lenguaje de programación a utilizar.• Elaborar un resumen con las opciones de cada menú del ambiente de desarrollo y su función.• Elaborar una tabla con los accesos rápidos a las funciones más importantes en el ambiente de desarrollo.• Acceder al sistema de ayuda del ambiente de desarrollo para resolver cualquier duda que se tenga sobre las herramientas o los comandos del lenguaje.• Crear las interfases para aplicaciones que indique el profesor.	14,15, 16

UNIDAD 4.- Introducción a la programación.

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
<p>Comprenderá el concepto de algoritmo y la terminología relacionada con los mismos.</p> <p>Identificará las características de las técnicas de diseño. Aplicará un lenguaje algorítmico gráfico o manuscrito.</p> <p>Comprenderá los conceptos básicos de la programación y escribirá expresiones aritméticas y lógicas en un lenguaje de programación</p>	<ul style="list-style-type: none">• Elaborar algoritmos de actividades cotidianas.• Organizar intercambios grupales acerca de conceptos de: identificador, constantes, variables y la proposición de asignación, entre otros.• Construir y evaluar expresiones matemáticas.• Compilar y ejecutar un programa modelo.	4,5,8,9,11,12

UNIDAD 5.- Funciones, procedimientos y estructuras de control.

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
<p>Comprenderá el uso de funciones y procedimientos en la programación modular.</p> <p>Comprenderá el uso y funcionamiento de las estructuras de selección y de repetición y las implementará en el desarrollo de aplicaciones.</p>	<ul style="list-style-type: none">• Realizar ejemplos que requieran funciones estándar.• Realizar ejemplos utilizando funciones definidas por el usuario• Realizar ejemplos utilizando paso de parámetros por valor y por referencia.• Usar el depurador del ambiente de desarrollo para seguir el paso de los parámetros en las funciones y los procedimientos.• Realizar una síntesis sobre el funcionamiento y aplicación de las estructuras selectivas.• Implementar el diseño de programas que requieran	5,9,14,15,16

	estructuras selectivas para probarlas en una aplicación.	
--	--	--

UNIDAD 6.- Arreglos, Archivos.

Objetivo Educativo	Actividades de Aprendizaje	Fuentes de Información
<p>Conocerá la representación interna de los arreglos unidimensionales y bidimensionales.</p> <p>Construirá modelos y desarrollará aplicaciones que requieran de éstos.</p> <p>Conocerá los diferentes tipos de registros y de archivos y los aplicará en el desarrollo de programas.</p>	<ul style="list-style-type: none"> • Desarrollar los programas de manipulación de los arreglos para realizar operaciones básicas • Distinguir los diferentes tipos de registros y archivos • Desarrollar los programas de manipulación de registros y archivos. • Desarrollar una aplicación tipo base de datos. 	5,9,14,15,16

10. FUENTES DE INFORMACIÓN

1. Jean Paul Tremblay & Richard B. Bunt. *Introducción a la ciencia de las computadoras (enfoque algorítmico)*. McGraw Hill. 1998.
2. Luis Joyanes Aguilar, Peña Tresancos Jaime, Vidal Fernández, María Carmen. *Introducción a la informática*. España: Mc. Graw Hill. 2004.
3. Levine, Guillermo. *Introducción a la computación y a la programación estructurada*. Ed. Mc. Graw Hill. 1989
4. Seymour E. Goodman, S. T. Hedetniemi. *Introduction to the design and analysis of algorithms*. Mc. Graw Hill. 1977
5. Joyanes Aguilar, Luis. *Fundamentos de programación, algoritmos y estructura de datos*. Mc. Graw Hill. 1996
6. Forsythe, Queman. Organik, Stenberg. *Lenguajes de diagramas de flujo*. Limusa.
7. O. J. Dahl, E. W. Dijatra, C.A.R. Hoare. *Programación estructurada*. Tiempo contemporáneo
8. Simon Mochon Cohen. *Actividades para el Autoaprendizaje Dirigido de: Word, Excel y Powerpoint*. McGraw-Hill
9. Baase, Sara . *Algoritmos Computacionales*. Prentice Hall. 2002.

10. Armando E. De Guisti. *Algoritmos, Datos y Programas*. Universidad Nacional de la Plata Prentice Hall. 2001
11. Ruth Maran, Paul Whitehead. *Aprenda Internet y la WWW Visualmente*. Trejos Hermanos Sucesores. 2002
12. Ramírez R, José Felipe. *Aprenda Visual Basic Practicando*. Pearson Education. 2001
13. *Manuales del usuario del lenguaje seleccionado*.
14. *Uso intensivo de Internet para obtener información relevante al curso*.
15. Ceballos, Fco. Javier. *Curso de programación de Visual Basic 6*. México :Alfaomega, 2000
16. Ceballos, F.J. De QBasic a Visual Basic. *Curso de Programación*. RAMA 1996.

11. PRÁCTICAS

Nota: El lenguaje de programación deberá definirse por el Instituto Tecnológico, aunque se sugiere emplear Visual Basic (en su forma completa o como *addendum* de Excel), Pascal o FORTRAN 90. Elaboración de un programa para resolver un sistema de ecuaciones lineales por el método de Gauss