



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO.

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN.



PROGRAMA DE ESTUDIO DE LA ASIGNATURA: PROGRAMACIÓN APLICADA 1053 DEL SÉPTIMO SEMESTRE  
DE LACARRERA DE INGENIERÍA QUÍMICA.

HORAS/SEMANA : 4 (2 TEÓRICAS/2 PRÁCTICAS)

CRÉDITOS : 6

ÓRGANO INTERNO QUE COORDINA EL PROGRAMA DE LA ASIGNATURA: DEPARTAMENTO DE MATEMÁTICAS

CAMPO : COMPLEMENTARIO. MODALIDAD: CURSO TEÓRICO-PRÁCTICO CARÁCTER DE LA ASIGNATURA: OPTATIVA

ASIGNATURA PRECEDENTE: PROGRAMACIÓN Y COMPUTACIÓN.

ASIGNATURA SUBSECUENTE: NINGUNA

## INTRODUCCIÓN

La simulación computacional de operaciones unitarias y procesos industriales, es una de las áreas de investigación de la Ingeniería Química más importante para el desarrollo tecnológico. Requiere de conocimientos de Ingeniería de Procesos, Diseño de Equipo, Diseño de Reactores, Fenómenos de Transporte, Físicoquímica, Dinámica y Control, Métodos Numéricos y principalmente, Programación. La Programación Orientada a Objetos es una herramienta que, utilizada adecuadamente, permite obtener algoritmos computacionales robustos y eficientes, además de ser más fácil darles mantenimiento. El presente curso de Programación Aplicada está dedicada a la Programación Orientada a Objetos con aplicaciones prácticas a la Ingeniería Química.

## OBJETIVO GENERAL DEL CURSO.

Proporcionar al alumno los conceptos y el conocimiento de un lenguaje de programación orientada a objetos y escribir algoritmos computacionales de utilidad a la Ingeniería Química.



DIRECCION GENERAL DE  
ADMINISTRACION ESCOLAR  
SUBDIRECCION DE  
CERTIFICACION Y CONTROL  
DOCUMENTAL  
DEPARTAMENTO DE PLANES  
Y PROGRAMAS DE ESTUDIO



**PROGRAMA :**

DIRECCION GENERAL DE  
 ADMINISTRACION ESCOLAR  
 SUBDIRECCION DE  
 CERTIFICACION Y CONTROL  
 DOCUMENTAL  
 DEPARTAMENTO DE PLANES  
 Y PROGRAMAS DE ESTUDIO

No de HORAS	TEMA :	No de HORAS	TEMA:
8	<p><b>UNIDAD I. INTRODUCCION A LA PROGRAMACION ORIENTADA A OBJETOS</b>  <b>OBJETIVO :</b> El alumno conocerá los conceptos y mecanismos básicos del paradigma de la programación orientada a objetos.  <b>CONTENIDO :</b> I.1. Introducción            I.2 Mecanismos básicos de la P.O.O.                I.2.1 Objetos                I.2.2 mensajes                I.2.3 Métodos                I.2.4 Clases, subclases y objetos.            I.3 Características de la P.O.O.                I.3.1 Abstracción.                I.3.2 Encapsulamiento                I.3.3 Herencia                I.3.4 polimorfismo                I.3.5 El paradigma de la P.O.O.                I.3.6 Diseño orientado a objetos.</p>	8	<p><b>UNIDAD II. CARACTERÍSTICAS DE C++</b>  <b>OBJETIVO :</b> Conocer las características básicas y avanzadas de la programación orientada a objetos.  <b>CONTENIDO :</b>            II.1 C++ y la P.O.O.            II.2 Diferencias entre C++ y C                II.2.1 Funciones. Nuevas palabras clave. Comentarios.                II.2.2 Operadores C++. Prioridad y orden de evaluación.                II.2.3 Conversión explícita del tipo de una expresión. Conversión del tipo Void                II.2.4 Tipos de datos: tipos predefinidos                II.2.5 Identificaciones y estructuras. Ambito referido a una estructura                II.2.6 Declaración de constantes: calificador Const. Flexibilidad en las declaraciones                II.2.7 El operador de resolución de ámbito                II.2.8 Enlace de tipo seguro            II.3 Entrada y salida estándar.            II.4 Argumentos por defecto en una función.            II.5 Funciones en línea.            II.6 Funciones sobrecargadas            II.7 Referencias            II.8 Paso de parámetros por referencia            II.9 Referencia como valor retornado            II.10 Los operadores New y Delete                II.10.1 Operador New                II.10.2 Puntero new handler</p>
5	<p><b>UNIDAD III. CLASES</b>  <b>OBJETIVO :</b> Proporcionar al estudiante las técnicas de programación orientada a objetos para el manejo de las diferentes clases de objetos.  <b>CONTENIDO :</b> III.1 Introducción y panorama general            III.2 Clases y miembros            III.3 Interfaces e implantaciones            III.4 Características secundarias de las clases            III.5 Construcción y destrucción.</p>	8	<p><b>UNIDAD VI. PATRONES</b>  <b>CONTENIDO :</b>            VI.1 Introducción            VI.2 Un patrón sencillo            VI.3 Patrones de Lista            VI.4 Patrones de función.            VI.5 Resolución de sobrecarga de función patrón            VI.6 Argumentos de patrón.            VI.7 Derivación y patrones            VI.8 Un arreglo asociativo            VI.9 Ejercicios.</p>
8	<p><b>UNIDAD IV. CLASES DERIVADAS</b>  <b>OBJETIVO :</b> Proporcionar al estudiante las técnicas de programación orientada a objetos para el manejo de las clases de objetos derivadas y abstractas.  <b>CONTENIDO :</b> IV.1 Introducción y panorama general            IV.2 Clases derivadas            IV.3 Clases abstractas            IV.4 Un programa completo            IV.5 Herencia múltiple            IV.6 Control de acceso            IV.7 Almacenamiento disponible.            IV.8 Ejercicios</p>	8	<p><b>UNIDAD VII. MANEJO DE EXCEPCIONES.</b>  <b>CONTENIDO:</b>            VII.1 Manejo de errores.            VII.2 Discriminación de excepciones.            VII.3 Asignación de nombres a las excepciones.            VII.4 Adquisición de recursos.            VII.5 Excepciones que no son errores.            VII.6 Especificaciones de interfaz.            VII.7 Excepciones no atrapadas.            VII.8 Alternativas para manejar errores.            VII.9 Ejercicios.</p>

<p><b>UNIDAD V. OPERADORES SOBRECARGADOS</b>  <b>CONTENIDO :</b>  V.1 Introducción.  V.2 Funciones de operadores  V.3 Conversión de tipo definida por el usuario  V.4 Literales.  V.5 Objetos grandes  V.6 Asignación e inicialización.  V.7 Subíndices.  V.8 Llamada a una función.  V.9 Indirección.  V.10 Incremento y decremento,  V.11 Una clase de cadena de caracteres.  V.12 Amigas y miembros  V.13 Advertencias.  V.14 Ejercicios.</p>	8	<p><b>UNIDAD VIII. FLUJOS (STREAMS)</b>  <b>CONTENIDO :</b>  VIII.1 Introducción.  VIII.2 Salidas  VIII.3 Entradas  VIII.4 Imposición de formato.  VIII.5 Archivos y flujos  VIII.6 Entradas/Salidas en C.  VIII.7 Ejercicios.</p>
--	---	--

**METODOLOGÍA DE LA ENSEÑANZA :**

Exposición oral  
Exposición audiovisual  
Elaboración de programas de cómputo orientados a objetos  
Prácticas de taller de cómputo

**MÉTODO DE EVALUACIÓN :**

Exámenes parciales  
Tareas y trabajos fuera del aula  
Desarrollo de programas

**REQUISITOS PARA CURSAR LA ASIGNATURA :**

Programación y computación

**PERFIL PROFESIOGRÁFICO DE QUIEN IMPARTE LA ASIGNATURA :**

Ingeniero Químico, Químicos y afines con conocimientos en el área de la computación.

**BIBLIOGRAFÍA BÁSICA :**

**Terribile, Mark A.**  
C++ Programming  
Mc Graw Hill. México, 1993

**Rau, Bindu R.**  
C++ and OOP Paradigm  
Mc Graw Hill. México, 1993

**Fasion, Ted**  
Borland C++ 3.1. Object oriented programming  
Edit. Osborne. Mc Graw Hill. México, 1992

**Spuler, David**  
C++ and C efficiency  
Prentice Hall. México, 1992

**Murray, L. And Papas, P.**  
Borland C++ Handbook  
Mc Graw Hill. México, 1993

**BIBLIOGRAFÍA COMPLEMENTARIA :**

**Murray, L. And Papas, P.**  
Microsoft C/C++ 7: The complete reference  
Mc Graw Hill. México, 1993

**Spuler, David**  
C++ and Object oriented programming.  
Prentice Hall. México, 1995



DIRECCION GENERAL DE  
ADMINISTRACION ESCOLAR  
SUBDIRECCION DE  
CERTIFICACION Y CONTROL  
DOCUMENTAL  
DEPARTAMENTO DE PLANES  
Y PROGRAMAS DE ESTUDIO