

Universidad de Puerto Rico en Aguadilla  
**Departamento de Matemáticas**

PRONTUARIO

Profesor : \_\_\_\_\_ Nombre Estudiante : \_\_\_\_\_  
 Oficina : \_\_\_\_\_ Sección : \_\_\_\_\_  
 Horas de Oficina : \_\_\_\_\_

I. Título del curso : **Introducción a la Programación Moderna de Computadoras**

II. Codificación : **MATE 3205**

III. Texto : *Visual Basic 4 in 12 Easy Lessons*  
 Starter Kit (con CD-ROM incluido)

IV. Número de horas/crédito : Cuatro créditos. Cuatro horas contacto semanales para un total de sesenta horas en el semestre. Dos horas semanales en conferencia y otras dos en el laboratorio.

V. Requisito previo : Ninguno

VI. Descripción del curso :

Este curso ofrecerá una introducción a la literacia de computadoras desde la perspectiva de la programación. Los estudiantes matriculados serán usuarios y futuros programadores. Se estudiarán temas: del funcionamiento físico y lógico de los sistemas, sistemas operativos, los API y los conceptos de la programación moderna. Se escogerá un sistema operativo y un lenguaje de programación como herramientas para el laboratorio.

VII. Objetivos Generales.

Al finalizar el curso el estudiante estará preparado para:

- A. Describir el funcionamiento de los componentes físicos de una computadora. ( periferales de entrada y salida, memoria primaria y secundaria, CPU).
- B. Describir la estructura funcional básica de un sistema operativo.
- C. Demostrar dominio en los conceptos básicos de la programación moderna.
- D. Escribir programas sencillos usando los conceptos de programación.
- E. Escribir programas sencillos usando los servicios del sistema operativo (API).
- F. Desarrollar las destrezas del trabajo en equipo.

VIII. Objetivos específicos y distribución de tiempo.

Lección	Sección y Tópico	Como resultado de las experiencias en el curso los estudiantes serán capaces de:	Actividad en el laboratorio.
	Prefacio.	Determinar los requisitos generales del curso y sus objetivos; cómo la calificación final es determinada y las estrategias instruccionales usadas en el curso.	
1		Enumerar las características de los sistemas operativos y algunas de las diferencias básicas entre ellos. (DOS, Windows, Win32, Windows NT, Linux). Funcionamiento del sistema operativo. Descripción de los	Buscar un listado de los sistemas operativos y sus usos específicos. Investigar el apoyo de las compañías a los mismos a través del Internet. ( <i>Bug Lists, Service</i>

Lección	Sección y Tópico	Como resultado de las experiencias en el curso los estudiantes serán capaces de:	Actividad en el laboratorio.
		componentes del "Hardware".	<i>Packages, Drivers Updates</i> etc.)
2		Establecer las diferencias entre las dos perspectivas de uso de los sistemas operativos: Programadores y Usuarios. Describir el funcionamiento del sistema operativo como administrador del "Hardware". ( <i>CPU, RAM, Discos, Periferales de entrada y salida</i> )	Buscar la información en el sistema operativo destinada para el uso de los programadores. Buscar un listado en el Internet de lugares de intercambio de información para programadores.
3		Conocer el uso específico de los API's y enumerar las categorías de los diferentes servicios. API: "Application Program Interface"	Instalación de un lenguaje de programación. Visita al "Web Site" de la compañía para buscar la información más reciente del lenguaje.
4		Servicios API's y las estructuras de datos del sistema para el acceso a ellos.	Investigar los API de otros sistemas operativos.
5	Unidad 1	Enumerar los Lenguajes Modernos de Programación, diferencias y similitudes entre ellos.	Buscar un listado de los lenguajes modernos y discutir su uso particular.
6	Unidad 2	Usar el ambiente del Sistema de programación IDE, (Integrated Developing Environment). Escribir, almacenar y compilar un programa.	Crear el primer programa usando el lenguaje. Buscar si existen lugares en el Internet donde se pueda encontrar código de programación.
7,8	Unidad 3 y Unidad 4	Establecer las diferencias entre la Programación con herramientas visuales y la programación clásica. Explicar de forma general los nuevos conceptos dentro de la programación (clases, propiedades, métodos, eventos, herencia, encapsulación y polimorfismo).	Realizar proyecto #1. Buscar en el Internet lugares donde se publican nuevos controles de programación visual.
8,9	Unidades 5,6 y 7	Explicar y Aplicar los conceptos de la programación visual. Enumerar los controles, controles visuales, objetos y clases más comunes junto con sus propiedades.	Realizar el proyecto #2.
10-12	Unidades 9,10 y 11	Aplicar los conceptos básicos de los métodos y eventos.	Realizar el proyecto #3.
13-15	Unidades 15,16 y 17	Aplicar los elementos básicos de la programación visual para la creación de programas.	Realizar el proyecto #4.
16,17	Windows API Help	Uso de servicios API para: el acceso a controles visuales.	Buscar el directorio de servicios API. Buscar los "Bugs" publicados de los API.
18	Windows API Help	Uso de servicios API para: el acceso a los archivos.	Contruir un programa haciendo uso de los API.
19	Windows API Help	Contruir un programa haciendo uso de los API.	Contruir un programa haciendo uso de los API.
20	Windows API Help	Usando herramientas para rastrear y encontrar errores en la programación.	Correr varios programas haciendo uso de un "Debugger", marcando variables, eventos y

Lección	Sección y Tópico	Como resultado de las experiencias en el curso los estudiantes serán capaces de:	Actividad en el laboratorio.
			definiendo condiciones

IX. Estrategias Instruccionales.

Para el logro de los objetivos, se utilizarán los siguientes métodos o técnicas de enseñanza:

- A. conferencias y talleres.
- B. Uso de computadora con los programados : Lenguaje de Computadoras, Sistema Operativo y comunicación con Internet.
- C. proyectos de aplicación en el laboratorio de computadora.
- D. la aplicación de las técnicas de aprendizaje cooperativo para realizar proyectos.

IX. Evaluación.

Se administrará 1 examen escrito, se asignarán al menos 3 proyectos especiales para trabajar en el laboratorio de computadoras y un examen final comprensivo. El valor relativo de cada nota queda a discreción del profesor y dependerá del grado de dificultad de cada tarea.

X. Sistema de calificación.

Se utilizará el siguiente sistema de calificación cuantificable:

100–90	A	Sobresaliente
89–80	B	Notable
79–65	C	Aprobado
64– 60	D	Deficiente
59– 0	F	No aprobado

XI. Bibliografía.

1. Perry, Greg, *Visual Basic Made Easy* (SAMS Publishing, 1996). ISBN 0-672-31059-7
2. Kamin, Samuel N. y Reingold, Eduard M., *Programming with Class* (McGraw Hill, 1996). ISBN: 0-07-051833-5

Creado por Julio E. Rodríguez Vargas  
 Editado por Wilfredo Martínez Parrilla  
 septiembre del 1997.