



## 1. INFORMACION GENERAL

### SILABO

Curso :	<b>TALLER DE PROGRAMACIÓN I</b>
Código :	II0205
Ciclo:	Segundo
Profesores:	Javier Añanos C. Oswaldo Bravo N. Silvia Campos. Cesar Cebreros D. Carlos Franco. Julio Valverde Ch.

## 2 SUMILLA.

### 2.1 Naturaleza de la Asignatura

Es un curso de carácter práctico.

### 2.2 Síntesis del Contenido

Aplicación de la metodología de programación orientada a objetos (POO). Componentes de un programa, Clases y Objetos. Estructuras de Control. Estructuras de Selección. Estructuras Repetitivas (simples, múltiples, iterativas, anidadas). Programación modular. Métodos. Definiciones de argumentos. Tipos por valor y tipos por referencia. Constructores y Destruyores. Arreglos (vectores, matrices). Cadenas de Caracteres. Manejo de Errores de Excepción. Punteros. Manejo de archivos. Crear y leer Datos desde Archivos. Formas de Acceso para entrada y salida de datos. Recursividad (Conceptos básicos).

### 2.3 Objetivos Generales.

- Aplicar las características fundamentales de la programación orientada a objetos (POO).
- Desarrollar programas con clases y métodos básicos.
- Entrenar al alumno en el uso de conceptos y técnicas necesarias para la solución de problemas utilizando una herramienta de programación visual.
- Lograr que los alumnos adquieran un grado de dominio en el trabajo de análisis y abstracción.
- 

## 3. OBJETIVOS ESPECIFICOS.

Al final del curso el alumno será capaz de:

1. Elaborar aplicaciones utilizando el paradigma de la POO.
2. Realizar trabajos ordenados y auto-documentados en la solución de problemas.

#### **4. PROGRAMACION DE CONTENIDOS.**

##### **PRIMERA SEMANA.**

TEORÍA : Etapas del desarrollo de programas y familiarización con el ambiente de programación elegido. Presentación y sintaxis de la herramienta de programación visual a utilizar en el curso.

TALLER : Entorno de programación. Parte Nro. 1

TALLER : Entorno de programación. Parte Nro.2

##### **SEGUNDA SEMANA.**

TEORÍA : Métodos y Estilos de Programación. Representación de los algoritmos, definición, componentes y características.

TALLER : Representación de Algoritmos, características y componentes. Parte Nro.1

TALLER : Aplicación de algoritmos en la programación. Parte Nro. 2

##### **TERCERA SEMANA.**

TEORÍA : Componentes de un programa. Datos variables. Datos constantes. Tipos de datos. Operadores y tipos de operadores. Expresiones y tipos de Expresiones. Revisión de sintaxis.

TALLER : Programación básica con datos variables y constantes. Parte 1

TALLER : Programación básica con operadores y expresiones. Parte 2

##### **CUARTA SEMANA.**

TEORÍA : Programación condicional simple, condicional doble, condicional múltiple. Presentación y sintaxis de lenguaje para “Programación con bifurcación”.

TALLER : Programación con bifurcación. Parte 1.

TALLER : Programación con bifurcación. Parte 2

##### **QUINTA SEMANA.**

TEORÍA : Programación iterativa I. Control de flujo. Bucles anidados.

TALLER : Programación iterativa. Parte 1.

TALLER : Programación iterativa. Parte 2.

##### **SEXTA SEMANA.**

TEORÍA : Programación iterativa II. Estructura para, Anidación de la estructura para. optimización de bucles.

TALLER : Programación iterativa. Parte 1.

TALLER : Programación iterativa. Parte 2.

##### **SEPTIMA SEMANA.**

TEORÍA : Programación modular. Implementación de Métodos: Paso por valor y paso por referencia.

TALLER : Programación modular. Parte 1.

TALLER : Programación modular. Parte 2.

## **OCTAVA SEMANA.**

EXAMEN PARCIAL.

## **NOVENA SEMANA.**

TEORÍA : Arreglos: Vectores y matrices. Operaciones con arreglos.

TALLER : Arreglos unidimensionales (Vectores).

TALLER : Arreglos bidimensionales (Matrices).

## **DECIMA SEMANA.**

TEORÍA : Manejo de Cadenas

TALLER : Tratamiento de cadenas de caracteres.

TALLER : Operaciones con cadenas de caracteres.

## **DECIMA PRIMERA SEMANA.**

TEORÍA : Manejo de Errores de Excepción

TALLER : Aplicación y alcance de Errores de Excepción. Parte 1.

TALLER : Aplicación y alcance de Errores de Excepción. Parte 2.

## **DECIMA SEGUNDA SEMANA.**

TEORÍA : Punteros

TALLER : Aplicaciones de punteros. Parte 1.

TALLER : Aplicaciones de punteros. Parte 2.

## **TALLER : DECIMA TERCERA SEMANA.**

TEORÍA : Archivos y flujos.

TALLER : Implementación para crear archivos . Parte 1.

TALLER : Acceso a archivos de datos. Parte 2.

## **DECIMA CUARTA SEMANA.**

TEORÍA : Operaciones con archivos.

TALLER : Grabación de archivos y datos Parte 1.

TALLER : Mantenimiento de archivos. Parte 2.

## **DECIMO QUINTA SEMANA.**

TEORÍA : Recursividad

TALLER : Aplicaciones. Parte 1.

TALLER : Aplicaciones. Parte 2.

## **DECIMA SEXTA SEMANA.**

Entrega de notas.

EXAMEN FINAL.

## **DECIMO SEPTIMA SEMANA.**

Entrega de notas Finales.

EXAMEN SUSTITUTORIO.

## **5. METODOLOGIA.**

### **5.1 Técnicas.**

- Análisis y síntesis en el desarrollo de programas para resolver ejercicios y solucionar problemas.

- Dialogo y exposición en la presentación de los talleres y de la sintaxis pertinente del lenguaje, con el apoyo de materiales y equipos disponibles.
- Tutoría.
- Taller guiado – construcción de programas- en ambiente de desarrollo de programas.
- Absolución de preguntas y reforzamientos continuos.

## 5.2 Material Didáctico.

1. Computador con software de presentación y video-proyector para presentación de talleres y sintaxis del lenguaje de programación seleccionado. Además de Equipos Multimedia.
2. Guías de talleres.
3. Un computador o estación de trabajo por alumno con un ambiente interactivo de desarrollo de programas en visual C#.NET

## 6. EVALUACION.

### 6.1 Criterios.

- Continua, basada en los conocimientos, habilidades, destrezas y conductas adquiridas, dentro del marco normativo de la Universidad.
- Formativa directa e indirecta. Directa, durante los diálogos y a través de las preguntas y respuestas verbales, e indirecta, vía tests.
- Cada taller tiene su propia evaluación acumulativa ( $T_i$ ); la cual es directa e indirecta: directa, durante el taller, e indirecta, a través de los informes y programas producto.
- La evaluación practica (PP) relacionada con las evaluaciones continuas tomadas en aula.
- La nota final del curso es el promedio aritmético redondeado, donde la nota de cada taller ( $T_i$ ) tiene peso simple y el promedio final de practica (PP) tiene peso doble.

La nota final resultará de aplicar la fórmula siguiente:

$$NF = \left( \sum_{i=1}^{T=10} T_i + 2 * PP \right) / 12$$

donde:

- $T_i$  representa el numero de talleres evaluados en laboratorios.
- PP representa el promedio final de práctica.
- El alumno, a su pedido, tiene derecho a sustituir una de sus evaluaciones practicas (PP), siempre y cuando su promedio final ( PP) sea mayor ó igual a 07.

## 7. RELACION DE LECTURAS

Portal de C# en español : ( <http://tdg.lsi.us.es/csharp>)

## 8. REFERENCIAS BIBLOGRAFICAS

1. Joyanes A. L. Fundamentos de programación. 2da Edición. 1996.McGraw Hill. España.

2. CAIRO, Oswaldo, (1995), Metodología de la Programación, Tomo I y II, Edit. Alfa y Omega
3. Villalobos S.J.A. Diseño y Manejo de Estructuras de datos en C. 1996. McGraw Hill Interamericana. Colombia.
4. Como programar en C#, H.M. Deitel, P.J Deitel Año 2002
5. Microsoft Visual C#, NET Edición de Aprendizaje ( McGraw Hill). Año 2002.
6. C# Para desarrolladores en JAVA. Jones (( McGraw Hill). Año 2003.
7. Visual C#.NET Francisco Charte. Editorial Anaya Multimedia.
8. GALVE, Javier y otros, (1993), Algoritmia, Edit. Addison-Wesley
9. AHO, Alfred y otros, (1988), Estructura de datos y algoritmos, Edit. Addison-Wesley
10. Inside C# de Tom Archer. Editorial Microsoft
11. A programmer's introduction to C# de Eric Gunnerson. Editorial Apress
12. C# Essentials de Beb Albahari, Peter Drayton y Brand Merril. Editorial O'Reilly
13. El lenguaje de programación .C# , Francisco Javier Ceballos Alfaomega-RAMA

### **PAGINAS WEB**

1. "C# Corner" (<http://www.c-sharpcorner.com>)
2. "C# Help" (<http://www.csharp-help.com>)
3. "C# Station" (<http://www.csharp-station.com>)
4. "Codehound C#" (<http://www.codehound.com/csharp>)
5. "csharpindex.com" (<http://www.csharpindex.com>)
6. "Developersdex" (<http://www.developersdex.com/csharp>)
7. ".NET Wire" (<http://www.dotnetwire.com>)