

## PROGRAMA SINÓPTICO POR COMPETENCIAS

### I. DATOS DE IDENTIFICACIÓN

<b>PROGRAMA ACADÉMICO:</b>	Ingeniería en Sistemas Computacionales		
<b>NOMBRE:</b>	Programación Orientada a Objetos	<b>CLAVE:</b> AED-1286	
<b>TIPO DE CURSO:</b>	Obligatorio		
<b>HORAS: (T.P.C.)</b>	<b>TEÓRICAS:</b> 2	<b>PRÁCTICAS:</b> 3	<b>CRÉDITOS ACADÉMICOS:</b> 5
<b>SEMESTRE:</b>	Segundo (2º)		
<b>FECHA DE ELABORACIÓN:</b>	13/ 09 /2013		
<b>ELABORADO POR:</b>	SNIT		

### II. COMPETENCIAS A DESARROLLAR:

Aplica la programación orientada a objetos para resolver problemas reales y de ingeniería.

### III. CONTENIDOS:

<b>UNIDAD I:</b> Introducción al paradigma de la programación orientada a objetos	
<b>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</b> Comprende y aplica los conceptos del paradigma de programación orientada a objetos para modelar situaciones de la vida real.	<b>CONTENIDO:</b> 1.1 Elementos del modelo de objetos: clases, objetos, abstracción, modularidad, encapsulamiento, herencia y polimorfismo 1.2 Lenguaje de modelado unificado: diagrama de clases
<b>UNIDAD II:</b> Clases y objetos	
<b>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</b> Aplica los conceptos de clases y objetos en el desarrollo de programas para solución de problemas de acuerdo al paradigma orientado a objetos.	<b>CONTENIDO:</b> 2.1 Declaración de clases: atributos, métodos, encapsulamiento 2.2 Instanciación de una clase 2.3 Referencia al objeto actual 2.4 Métodos: declaración, mensajes, paso de parámetros, retorno de valores 2.5 Constructores y destructores declaración, uso y aplicaciones 2.6 Sobrecarga de métodos 2.7 Sobrecarga de operadores: Concepto y utilidad, operadores unarios y binarios
<b>UNIDAD III:</b> Herencia	
<b>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</b> Identifica y aplica relaciones de herencia en clases derivadas para reutilizar los miembros de una clase base en el desarrollo de aplicaciones.	<b>CONTENIDO:</b> 3.1 Definición: clase base, clase derivada 3.2 Clasificación: herencia simple, herencia múltiple 3.3 Reutilización de miembros heredados 3.4 Referencia al objeto de la clase base 3.5 Constructores y destructores en clases derivadas 3.6 Redefinición de métodos en clases derivadas

<b>UNIDAD IV: Polimorfismo</b>	
<b>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</b> Aplica el concepto de polimorfismo para la definición de clases abstractas e interfaces que permitan reutilización de código.	<b>CONTENIDO:</b> 4.1 Definición 4.2 Clases abstractas: definición, métodos abstractos, implementación de clases abstractas, modelado de clases abstractas 4.3 Interfaces: definición, implementación de interfaces, herencia de interfaces 4.4 Variables polimórficas (plantillas): definición, uso y aplicaciones 4.5 Reutilización de código

<b>UNIDAD V: Excepciones</b>	
<b>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</b> Comprende y aplica las condiciones apropiadas para evitar los errores que pueden interrumpir el flujo normal de ejecución de un programa a través del manejo de excepciones.	<b>CONTENIDO:</b> 5.1 Definición 5.2 Tipos de excepciones 5.3 Propagación de excepciones 5.4 Gestión de excepciones: manejo de excepciones, lanzamiento de excepciones 5.5 Creación y manejo de excepciones definidas por el usuario

<b>UNIDAD VI: Flujos y Archivos</b>	
<b>COMPETENCIA ESPECÍFICA DE LA UNIDAD:</b> Comprende y aplica la clasificación de archivos y operaciones básicas sobre éstos para manipular su información.	<b>CONTENIDO:</b> 6.1 Definición 6.2 Clasificación: Archivos de texto y binarios 6.3 Operaciones básicas y tipos de acceso 6.4 Manejo de objetos persistentes

#### IV. FORMA DE EVALUACIÓN:

Las técnicas, herramientas y/o instrumentos sugeridos que permiten obtener el producto del desarrollo las actividades de aprendizaje: prácticas, mapas conceptuales, mapas mentales, resúmenes, reportes de prácticas. Las técnicas, herramientas y/o instrumentos sugeridos que me permite constatar el logro o desempeño de las competencias del estudiante: matriz de valoración, listas de cotejo, listas de verificación, guías de observación.

#### V. REFERENCIA BIBLIOGRÁFICA:

Budd, T. (2002). Object Oriented Programming. Third edition: Addison Wesley. 2 Ceballos F. (2003). Programación Orientada a Objetos con C++. 3ra. Edición: Editorial Ra-Ma 3 Ceballos J. (2012) Microsoft C# -Curso de Programación. España: Alfaomega 4 Clark D., (2013). Beginning C# Object-Oriented Programming, Apress. 5 Craig I. (2002). The Interpretation of Object-Oriented Programming Languages. 2nd Edition: Springer. London. 6 Dean J. y Dean R. (2009)

Introducción a la programación con Java.: McGraw Hill 7 Deitel P., Deitel H. (2013). Como programar en java. 9a. Edición. Pearson. 8 Doyle, B (2013) C# Programming: From Problem Analysis to Program Design. Cengage Learning 9 Groussard, T. (2009) Visual Basic.NET (VB.NET) - Programe con Visual Studio 2008. Espana: Eni Ediciones 10 Groussard, T. (2011) Recursos Informáticos C#4 Los fundamentos del lenguaje- desarrollar con visual estudio 2010. España: Eni Ediciones 11 Gutiérrez F., Duran F., Pimentel E. (2007). Programación Orientada a Objetos con JAVA: Ediciones Paraninfo, S.A. 12 Harvey M. (2008). Como programa en Java. México: Prentice Hall. 13 Holmes B., Joyce D. (2001). Object-Oriented Programming with JAVA. Jones and Barlett: Canada. Publishers Inc. 14 Joyanes L. (2011). Fundamentos de programación: Algoritmos, Estructuras de Datos y Objetos. 3ra. Edición. Mac-Graw Hill 15 Joyanes, L. y Zahonero, I. (2011) Programación en Java 6. España: McGraw Hill 16 Larman G. (2002). UML y Patrones 2/E: Pearson Educacion. 17 Marrer G. (2009). Fundamentals of Programming with Object Orientated Programming. Python Edition. Ebook Edition. ©TecNM mayo 2016

Página | 10 TECNOLÓGICO NACIONAL DE MÉXICO Secretaría Académica, de Investigación e Innovación Dirección de Docencia e Innovación Educativa 18 Muñoz C., Nino A. Vizcaino A. (2002). Introducción a la Programación con Orientación a Objetos: Pearson Educacion. 19 Schildth H. (2002). Fundamentos de programación en Java: Mac-Graw Hill 20 Solana Aroa (2009). Programación con C# 4.0. Madrid, España. 21 Velez Serrano, J.F., Peña Abril, A., Gortázar Bellas, F. Sánchez Calle, A. (2010) Diseñar y Programar, Todo Es Empezar: Una Introducción a la Programación Orientada a Objetos Usando UML Y JAVA (EBOOK). 22 VV.AA. (2004). C/C++ Y Java. Como programar (4ª Ed). México 23 VV.AA., (2006) Introducción a la Programación Orientada a Objetos: Universidad de Alicate. Servicio de Publicaciones. 24 Warnes D., Kolling M.(2007). Programación Orientada a Objetos con Java. 3ra Edición: PrenticeHall. 25 BlueJ, (2014). A free Java Development Environment designed for beginners. Disponible en internet en [www.bluej.org](http://www.bluej.org). Fecha de acceso: 13 de Febrero de 2014. 26 Greenfoot, (2014). Teach and learn java programming. Disponible en internet en [www.greenfoot.org](http://www.greenfoot.org). Fecha de acceso: 13 de Febrero de 2014. 27 Jeroo, (2014). Jeroo. Disponible en internet en <http://home.cc.gatech.edu/dorn/jeroo>. Fecha de acceso: 13 de Febrero de 2014 28 Oracle (2014). Java platform – Standard edition 7 API specification. Disponible en internet en <http://docs.oracle.com/javase/7/docs/api/>. Fecha de acceso: 13 de Febrero de 2014.