

**GOBIERNO CONSTITUCIONAL DEL ESTADO LIBRE Y SOBERANO DE OAXACA
INSTITUTO ESTATAL DE EDUCACIÓN PÚBLICA DE OAXACA
COORDINACIÓN GENERAL DE PLANEACIÓN EDUCATIVA
COORDINACIÓN GENERAL DE EDUCACIÓN MEDIA SUPERIOR Y SUPERIOR**

PROGRAMA DE ESTUDIOS

NOMBRE DE LA ASIGNATURA	PARADIGMAS DE PROGRAMACIÓN I
-------------------------	------------------------------

CICLO CUARTO SEMESTRE	CLAVE DE LA ASIGNATURA PI-04	TOTAL DE HORAS 80
--------------------------	---------------------------------	----------------------

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA
Proporcionar los fundamentos de los paradigmas de programación; haciendo énfasis en el paradigma orientado a objetos para que sea capaz de diseñar e implementar programas en algún lenguaje de programación.

TEMAS Y SUBTEMAS
1. CONCEPTOS GENERALES SOBRE PARADIGMAS DE PROGRAMACIÓN 1.1. Introducción 1.1.1. Definición de paradigma 1.1.2. Lenguajes de programación 1.2. Paradigmas de programación 1.2.1. Imperativa 1.2.2. Funcional 1.2.3. Lógica 1.2.4. Orientada a objetos 1.2.5. Orientada a eventos 1.2.6. Otros paradigmas 2. PARADIGMA ORIENTADO A OBJETOS 2.1. Beneficios del paradigma orientado a objetos sobre otros paradigmas 2.2. Elementos primordiales 2.2.1. Clases y objetos 2.2.2. Abstracción 2.2.3. Encapsulamiento 2.2.4. Modularidad 2.2.5. Jerarquía y herencia 2.2.6. Polimorfismo 2.3. Ingeniería de software orientada a objetos 2.4. Diseño de clases empleando UML

3. INTRODUCCIÓN A LA TECNOLOGÍA DE UN LENGUAJE ORIENTADO A OBJETOS

- 3.1. Conceptos generales sobre el lenguaje
- 3.2. Características del lenguaje
- 3.3. Compilación y ejecución de programas

4. DATOS, OPERADORES Y SENTENCIAS DE CONTROL

- 4.1. Datos
 - 4.1.1. Tipos de datos
 - 4.1.2. Identificadores
 - 4.1.3. Arreglo unidimensional (vectores)
 - 4.1.4. Arreglo bidimensional
 - 4.1.5. Cadenas de caracteres
- 4.2. Operadores y expresiones
 - 4.2.1. Prioridad de operadores
 - 4.2.2. Evaluación de expresiones
 - 4.2.3. Promoción de tipos de dato
 - 4.2.4. Conversiones de tipos de datos
- 4.3. Estructuras condicionales y cíclicas
- 4.4. Definición de clases
 - 4.3.1. Atributos
 - 4.3.2. Métodos
 - 4.3.3. Instanciación
- 4.5. Modificadores de acceso
- 4.6. Composición
- 4.7. Ámbito de los atributos y métodos.
- 4.8. Auto-referencia (this)

5. MÉTODOS Y MENSAJES

- 5.1. El método como elemento de la comunicación
 - 5.1.1. Concepto de parámetro
 - 5.1.2. Parámetros de salida y de entrada
- 5.2. Declaración de métodos
- 5.3. Llamadas a métodos (mensajes)
- 5.4. Tipos de métodos
 - 5.4.1. Métodos constantes y estáticos
 - 5.4.2. Métodos normales y volátiles
- 5.5. Forma de pasar argumentos
- 5.6. Devolver un valor desde un método
- 5.7. Sobrecarga de métodos

6. CONSTRUCTORES

- 6.1. Conceptos de constructor

- 6.2. Declaración de métodos constructores
 - 6.2.1. Constructor sin parámetros
 - 6.2.2. Constructor con parámetros
 - 6.2.3. Sobrecarga de constructores
 - 6.2.4. Constructor copia

7. HERENCIA

- 7.1. Introducción a la herencia
- 7.2. Herencia simple
- 7.3. Herencia múltiple
- 7.4. Clase base y clase derivada
 - 7.4.1. Definición
 - 7.4.2. Declaración
 - 7.4.3. Uso de métodos de la clase base
- 7.5. Importancia de la encapsulación
- 7.6. Sobreescritura de métodos
- 7.7. Constructores y herencia
- 7.8. Introducción al polimorfismo

8. INTERFACES GRÁFICAS DE USUARIO (GUI)

- 8.1. Contenedores
- 8.2. Componentes
- 8.3. Administradores de diseño de GUI
- 8.4. Elementos gráficos
- 8.5. Cuadros de diálogo
- 8.6. Control de eventos

ACTIVIDADES DE APRENDIZAJE

Para cada unidad de aprendizaje se realizarán programas en los que el alumno practicará lo aprendido y demostrará sus conocimientos.

Se propone la realización de un proyecto final que sea aplicable a la vida real.

Revisar material existente en diferentes medios con el fin de enriquecer los conocimientos del alumno.

CRITERIOS Y PROCEDIMIENTOS DE EVALUACIÓN Y ACREDITACIÓN

Se realizan tres evaluaciones parciales y una evaluación ordinaria final de la asignatura.

Para las evaluaciones parciales, se deberá realizar un examen escrito y se podrá complementar la evaluación con exámenes prácticos, avances de proyectos, tareas, investigaciones y otras actividades académicas previamente aprobadas de acuerdo con la normatividad Universitaria. Queda a criterio del profesor la ponderación de todas las actividades.

Para la evaluación ordinaria final, se deberá realizar un examen escrito y se podrá complementar la evaluación con proyectos, exposiciones, tareas e investigaciones realizadas a lo largo del semestre. Queda a criterio del profesor la ponderación de todas las actividades.

Para la calificación final de la asignatura, se establece la ponderación de las evaluaciones parciales y ordinaria final con base en la normatividad de la Universidad.

BIBLIOGRAFÍA (TIPO, TÍTULO, AUTOR, EDITORIAL Y AÑO)

Básica:

- Cómo programar en C++. Deitel, Harvey M.; Deitel, Paul J. Prentice Hall. 2003, 4ª Edición.
- Cómo programar en Java. Deitel, Harvey M.; Deitel, Paul J. Prentice Hall. 2004, 5ª Edición.
- El lenguaje unificado de modelado. Booch, Grady; Rumbaugh, James; Jacobson, Ivar. Addison Wesley. 1999, 1ª Edición.
- Fundamentos de Java. Schildt, Herbert. McGraw-Hill. 2007, 3ª Edición.
- Fundamentos de programación en Java 2. Schildt, Hebert. McGraw-Hill. 2001, 1ª Edición.
- Fundamentos de programación: algoritmos, estructura de datos y objetos. Joyanes Aguilar, Luis. McGraw-Hill. 2008, 1ª Edición.
- Ingeniería de software orientada a objetos con UML, Java e internet. Weitzenfeld, Alfredo. Thomson. 2005.
- Java 2: curso de programación. Ceballos Sierra, Francisco Javier. Alfaomega, 2006.
- Java 2: manual del usuario y tutorial. Froufe Quintas, Agustín. Alfaomega Ra-Ma. 2006, 4ª Edición.
- Piensa en Java. Eckel, Bruce. Pearson. 2007.
- Programador certificado Java 2: curso práctico. Martín Sierra, Antonio J. Alfaomega Ra-Ma. 2008, 2ª Edición.
- UML para programadores Java. Martin, Robert C. Pearson. 2004.

Consulta:

- A practical introduction to data structures and algorithm analysis (C++ Edition). Shaffer, Clifford A. Prentice Hall. 2000, 2ª Edición.
- Algoritmos en C++. Sedgewick, Robert. Pearson. 2000, 1ª Edición.
- Análisis y diseño orientado a objetos con aplicaciones. Booch, Grady. Pearson. 1996, 2ª Edición.
- Análisis y diseño orientado a objetos de sistemas usando UML. Bennet, Simon; Farmer, Ray; Mcrobb, Steve. McGraw-Hill. 2006.
- Aplique Turbo C++. Schildt, Herbert. McGraw-Hill. 1991, 1ª Edición.
- Aprenda y practique Java. Jamsa, Kris. Oxford University Press. 2000.
- C# manual de programación. Joyanes Aguilar, Luis. McGraw-Hill. 2002.
- C++ guía de autoenseñanza. Schildt, Herbert. McGraw-Hill. 1995.
- Comunicaciones y bases de datos con java a través de ejemplos. Bobadilla Sancho, Jesús; Sancho Hernández, Adela. Alfaomega. 2003.
- Core Java 2: volumen I, fundamentos. Horstmann, Cay S.; Cornell, Gary. Pearson. 2006, 7ª Edición.
- Introducción a la programación orientada a objetos con Java. Wu, C. Thomas. McGraw-Hill. 2001.
- Java 2 fundamentos, volumen I. Horstmann, Cay S.; Cornell, Gary. Prentice Hall. 2003.
- Java 2: interfaces gráficas y aplicaciones para Internet. Ceballos Sierra, Francisco Javier. Alfaomega. 2006, 2ª Edición.
- Java 2: manual de programación. Joyanes Aguilar, Luis; Fernández Azuela, Matilde. McGraw-Hill. 2001.
- Java con programación orientada a objetos y aplicaciones en la WWW. Wang, Paul S. Thomson. 2000, 1ª Edición.
- Java: fundamentos de programación. Bishop, Judy M. Addison Wesley. 1999.
- Object oriented programming: an evolutionary approach. Cox, Brad J. Addison Wesley. 1991, 2ª Edición.
- Professional Java: JDK. Richardson, W. Clay; Avondolio, Donald; Vitale, Joe; Schrager, Scot; Mitchell, Mark W.; Scanlon, Jeff. Wrox. 2005.
- Programación Algol. Mc Cracken, Daniel D. Limusa. 1983.
- Programación en 3D con Java 3D. Pratdepadua, Joan J. Alfaomega Ra-Ma. 2003.
- Programación en Java 2: algoritmos, estructuras de datos y programación orientada a objetos. Joyanes Aguilar, Luis; Zahonero Martínez, Ignacio. McGraw-Hill. 2002.
- Programación en Java 5.0. Cohoon, James P. McGraw-Hill. 2006.
- Programación multithread en Java. Lewis, Bill; Berg, Daniel. Pearson. 2003.
- SCJP Sun certified programmer for Java 6 exam 310-065. Sierra, Katherine; Bates, Bert. McGraw-Hill. 2008, 1ª Edición.

- SQL y Java: guía para sqlj, jdbc y tecnologías relacionadas. Melton, Jim. Alfaomega. 2002.
- Thinking in Java. Eckel, Bruce. Prentice Hall. 2006, 4ª Edición.
- Turbo C/C ++: manual de referencia. Schildt, Herbert. McGraw-Hill. 1992, 1ª Edición.
- UML gota a gota. Fowler, Martin; Scott, Kendall. Addison Wesley. 1999.

PERFIL PROFESIONAL DEL DOCENTE

Licenciatura en Informática, Ingeniería en Ciencias Computacionales o afines, con grado de Maestría y preferentemente de Doctorado en Informática, Ciencias Computacionales o afines. Con experiencia profesional y docente de un año.