

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
COORDINACIÓN DE FORMACIÓN BÁSICA
COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA
PROGRAMA DE UNIDAD DE APRENDIZAJE

I. DATOS DE IDENTIFICACIÓN

1. Unidad Académica (s): Facultad de Ciencias
2. Programa (s) de estudio: Lic. en Ciencias Computacionales, Lic. en Matemáticas Aplicadas
3. Vigencia del plan:
4. Nombre de la Unidad de Aprendizaje: Programación Orientada a Objetos 5. Clave _____
6. HC: 2 HL: 2 HT: 1 HE: 2 CR 7
7. Etapa de formación a la que pertenece: Básica
8. Carácter de la Unidad de aprendizaje: Obligatoria X Optativa _____
9. Requisitos para cursar la unidad de aprendizaje:

Formuló: Omar Álvarez Xochihua

Vo. Bo. Dr. Alberto Leopoldo Moran y Solares

Fecha: Agosto de 2016

Cargo: Subdirector

II. PROPÓSITO GENERAL DE LA UNIDAD DE APRENDIZAJE

El paradigma de la programación orientado a objetos ofrece un enfoque distinto en la forma de diseño, desarrollo y mantenimiento del software, comparado con la programación estructurada o convencional. Este paradigma permite implementar de una forma más natural características de portabilidad, reusabilidad y mantenimiento del código, impactando en la reducción de ciclos de desarrollo y eliminando técnicas de codificación no intuitivas.

El propósito general de la asignatura es diseñar aplicaciones que transporten objetos reales a abstracciones de código, relacionando estos objetos por medio de asociaciones y/o generalizaciones, entre otras relaciones, e implementándolos y evaluándolos a través del uso de algún lenguaje orientado a objetos.

La asignatura está ubicada en la etapa básica y es de carácter obligatorio para la Lic. en Ciencias Computacionales y optativo para la Lic. en Matemáticas Aplicadas.

III. COMPETENCIA DE LA UNIDAD DE APRENDIZAJE

Desarrollar aplicaciones mediante el paradigma de programación orientada a objetos para automatizar procesos o representar problemas reales e hipotéticos en forma creativa y objetiva.

IV. EVIDENCIA (S) DE DESEMPEÑO

Desarrolla aplicaciones en un lenguaje de programación orientado a objetos, la cual deberá contener la documentación formal de las soluciones desarrolladas.

V. DESARROLLO POR UNIDADES

Competencia

Comparar el paradigma orientado a objetos con otras técnicas de programación, analizando sus características y sus aplicaciones típicas para identificar las ventajas de este nuevo modelo de programación con una actitud crítica.

Contenido

Duración 2 hrs.

I.- Introducción

1. Historia de la programación.
2. Tecnología y lenguajes
3. Técnicas de programación
 - a. Estructurada
 - b. Procedimental
 - c. Orientada a Objetos

Competencia

Describir la importancia de la programación orientada a objetos, definiendo sus principales elementos y características que le permitan identificar las ventajas de este paradigma de programación con una actitud crítica.

Contenido

Duración 2 hrs.

II.- Conceptos Orientados a Objetos

1. Objeto.
2. Clase.
3. Mensajes.
4. Ventajas de Orientación a Objetos.

Competencia

Diseñar soluciones a problemas hipotéticos, utilizando los conceptos del paradigma orientado a objetos que le permitan explotar los elementos de este paradigma de programación con una actitud analítica y creativa.

Contenido

Duración 4 hrs.

III.- Diseño Orientado a Objetos

1. Abstracción.
2. Encapsulamiento.
3. Polimorfismo.
4. Herencia.
5. Híbridos.
6. Relaciones.

Competencia

Modelar visualmente soluciones a problemas hipotéticos, utilizando los elementos básicos del Lenguaje de Modelado Unificado (UML) que le permitan generar escenarios gráficos de sus soluciones con una actitud analítica y creativa.

Contenido

hrs.

IV.- Modelado Orientado a Objetos

1. Introducción.
2. Elementos básicos de UML.
3. Diseño Estático: diagramas de clases.

Duración 4**Competencia**

Implementar los conceptos de programación orientación a objetos, mediante un lenguaje de programación que soporte este paradigma que le permita obtener la experiencia en el desarrollo de aplicaciones orientadas a objetos con una actitud analítica y creativa.

Contenido

V.- Programación Orientada a Objetos

1. Creación de clases
2. Encapsulamiento
3. Polimorfismo
4. Herencia Simple
5. Herencia Múltiple
6. Paquetes
7. Reutilización de código

Duración 10 hrs.**Competencia**

Aplicar los conceptos avanzados de la programación orientada a objetos, mediante la implementación de un sistema de cómputo que integre los elementos y técnicas vistas en clase para la solución de un problema de la vida real con una actitud analítica y creativa.

Contenido

VI.- Aplicaciones

1. Interfaz de usuario.
2. Colecciones de objetos.
3. Flujos.
4. Excepciones.
5. Hilos.

Duración 10 hrs.

VI. ESTRUCTURA DE LAS PRÁCTICAS

No. de Práctica	Competencia(s)	Descripción	Material de Apoyo	Duración
1 Taller	Realizar una investigación sobre las técnicas de programación con base en las aplicaciones típicas en las que es utilizada esta técnica para conocer las ventajas y desventajas con actitud crítica y analítica.	El alumno conocerá la posición de la programación orientada a objetos dentro del ámbito de las técnicas de programación con base en las aplicaciones típicas en las que es utilizada esta técnica.	Libros Internet Apuntes Cuaderno Lápiz	3 hrs.
2 Taller	Realizar ejercicios sobre la definición de conceptos y su relación con la vida real, destacando las diferencias en utilizar estos conceptos en la vida diaria para conocer las principales características de la técnica con responsabilidad.	El alumno identificará las principales características de la técnica de orientación a objetos evaluando las causas por las que se vuelve importante el paradigma de la programación orientada a objetos.	Libros Internet Papel	3 hrs.
3 Taller	Desarrollar una solución a un problema hipotético planteado por el profesor, destacando los componentes necesarios en la solución así como un modelo para visualizar dicha solución, y exponer los modelos ante el grupo con responsabilidad.	El alumno utilizará los conceptos del desarrollo orientado a objetos para plantear una solución de problemas hipotéticos.	Libros Papel	10 hrs.
4	Realizar un modelo que pueda ser implementado mediante un sistema de cómputo en el cual se visualice el modelo desarrollado para establecer la solución de un problema con actitud crítica y responsable.	El alumno dominará los elementos básicos de una herramienta de modelado, para diagramar los conceptos de orientación a objetos.	Libros Computadora Internet Papel	10 hrs.

5	Desarrollar un sistema de cómputo utilizando un lenguaje de programación en el cual se visualice el modelo desarrollado para establecer la solución de un problema con creatividad e innovación.	El alumno dominará una técnica para implementar los conceptos de orientación a objetos en un lenguaje de programación.	Libros Computadora Internet Papel	10 hrs.
6	Plantear un problema real observado, para desarrollar una solución, e implementarla a través de un lenguaje de programación que soporte el paradigma orientado a objetos con creatividad y responsabilidad.	El alumno identificará problemas reales y desarrollará una solución del mismo desde la perspectiva de orientación a objetos.	Libros Computadora Internet Papel	12 hrs.

VII. METODOLOGÍA DE TRABAJO

Aprendizaje participativo

Durante el desarrollo de la unidad de aprendizaje, el docente fomentará la participación activa de los estudiantes en actividades individuales y grupales. Mediante la discusión de las temáticas vistas en las clases teóricas y la asignación de ejercicios y prácticas en las sesiones de taller y laboratorio. En las actividades grupales el docente asignará un problema a cada equipo, el cual primeramente debe ser analizado en forma individual, posteriormente se discutirá y definirá una solución en equipo previa implementación de la misma. La participación del maestro en la aplicación de esta metodología es de mediador.

Prácticas de laboratorio

En las sesiones de laboratorio, el estudiante llevará a la práctica los conocimientos teóricos vistos en clase y los ejercicios realizados durante el taller, al mismo tiempo que desarrolle habilidades con el manejo de al menos un lenguaje de programación orientado a objetos y un ambiente integrado de desarrollo.

Investigación Bibliográfica

Se sugiere solicitar investigación en diferentes fuentes bibliográficas sobre temas de actualidad o temáticas que serán discutidos posteriormente en clase. El propósito de estos trabajos es fomentar el autoaprendizaje y que el estudiante aprenda a realizar investigación en medios electrónicos (Internet), libros, y revistas sobre temas del área. Las fuentes serán tanto en el idioma inglés como español. Los reportes deberán contener las referencias que se utilizaron para la realización del trabajo y debe contar imprescindiblemente una conclusión personal acerca de la investigación. El maestro debe enfatizar a los estudiantes que los reportes escritos sean claros y bien redactados, recalcándoles también las faltas de ortografía.

Ejercicios y exámenes de conocimientos

El maestro deberá aplicar al menos 2 exámenes de conocimientos durante el periodo, que permitan identificar la obtención de competencias de los estudiantes. Los exámenes podrán ser de varios tipos, tales como: de preguntas abiertas, opción múltiple y solicitud de programas. Así como, la asignación de ejercicios para ser realizados en la sesión de taller o extra clase, de tal manera que refuercen los conocimientos aprendidos durante la clase teórica. Se solicitará la entrega oportuna y formal de tareas y trabajos de investigación.

Los alumnos, al final del periodo, realizan un proyecto de forma individual o en equipo de máximo tres personas, sobre problemas reales.

VIII. CRITERIOS DE EVALUACIÓN

Criterio de acreditación:

Aplicar al menos 2 exámenes parciales en tiempo y forma.

Cumplir con las prácticas y tareas extra-clase en tiempo y forma.

Cumplir con la presentación del proyecto final (desarrollo de la aplicación orientada a objetos) en tiempo y forma.

Cumplir con los requisitos de acreditación establecidos en el estatuto escolar vigente.

Evaluación:

La evaluación general de la asignatura consistirá de exámenes teórico-prácticos, tareas-reportes, prácticas de laboratorio y un proyecto final.

El proyecto final deberá realizarse en equipo de trabajo, la evaluación se dividirá en dos partes el 50% de la calificación será asignado al producto terminado y la segunda se asignará a un reporte escrito y a la exposición oral acerca del proyecto.

Los reportes por escrito será por equipo y los puntos a evaluar son, contenido, claridad y forma, así como ortografía y redacción; para la exposición oral los puntos a evaluar serán, dominio del tema, claridad y estructura. Los alumnos pueden ayudarse en la exposición mediante apoyos visuales tales como proyector de transparencias, acetatos u otros medios.

"Para la acreditación del curso se atenderá al Estatuto Escolar Vigente, artículos 70-71, por lo que el estudiante deberá contar un mínimo de 80% de asistencias en el periodo. Tener un mínimo aprobatorio de 60 en su calificación final".

Calificación:

Los porcentajes de evaluación propuestos serán los siguientes:

Exámenes	30%
Tareas/prácticas	30%
Proyecto final	40 %
Total	100%

IX. BIBLIOGRAFÍA

Básica	Complementaria
<ul style="list-style-type: none">• Deitel M., Harvey; Paul J., Deitel. (2014). "C++ Cómo programar". Pearson Educación, 9na. Edición.• Ramírez, Felipe. (2012). "Introducción a la programación: algoritmos y su implementación en VB.Net, C#, Java y C++". Alfaomega, 2da. Edición.• Stroustrup, B., (2013). The c++ programming language, 4th edition, Addison-Wesley.• Weisfeld, M., (2013). The object-oriented thought process, 4th ed. Addison-Wesley.	<ul style="list-style-type: none">• Booch, G., (2005). <i>The Unified Modeling Language User Guide</i>, 2da Edición, Addison-Wesley. (Clásico).• Schach, Stephen R., (2004). <i>An introduction to object-oriented systems análisis and design with UML and the unified process</i>, McGraw-Hill. (Clásico).• Unified Modelling Language (disponible en: http://www.uml.org, Agosto 2016)

X. PERFIL DEL DOCENTE

El docente de esta asignatura deberá contar con un título de licenciatura en Ciencias Computacionales o área afín; con experiencia en lógica programática, análisis, diseño e implementación de programas de computadora, conocimiento de lenguajes de programación basados en el paradigma orientado objetos, y tener experiencia docente.