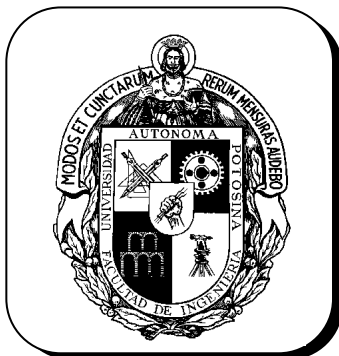


FACULTAD DE INGENIERÍA

CENTRO DE INVESTIGACIÓN Y ESTUDIOS DE POSGRADO



Nombre de la materia: TALLER DE PROGRAMACIÓN
ORIENTADA A OBJETOS

Clave Facultad:

Clave U.A.S.L.P.:

No. de créditos: 4

Horas/Clase/Semana: 2

Horas totales/Semestre: 32

Horas/Práctica (y/o Laboratorio):

Prácticas complementarias:

Trabajo extra clase Horas/Semana: 2

Carrera/Tipo de materia: Posgrado en Ingeniería Mecánica
Optativa común

No. de créditos aprobados:

Fecha última de Revisión Curricular: Marzo 2020

JUSTIFICACIÓN DEL CURSO

Esta materia está dirigida a desarrollar el razonamiento lógico para la resolución de problemas relacionados con la ingeniería, suponiendo que el alumno tenga el conocimiento previo de lenguajes de programación acordes a la programación estructurada, en particular sobre el uso del Lenguaje C.

En este curso, se introducirá al alumno en la programación orientada a objetos, aplicando

metodologías y técnicas que permitan un mejor manejo de los datos y una mejor organización del código. Además, se introduce herramientas para el desarrollo de la interfaz gráfica (GUI) que permite un desarrollo más profesional para uso con más comodidad y/o por personas que no requieren o tengan acceso al conocimiento de la programación.

OBJETIVO DEL CURSO

Desarrollar el razonamiento lógico necesario y aprender la sintaxis básica del Lenguaje de Programación C++. Con estas habilidades el alumno podrá estructurar y desarrollar soluciones a problemas básicos de ingeniería

con el apoyo de la computadora, utilizando el paradigma de programación orientada a objetos, y la creación de bibliotecas de clases.

CONTENIDO TEMÁTICO

I PROGRAMACIÓN ORIENTADA A OBJETOS

1. INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA A OBJETOS (POO) 5 hrs.

Objetivo: Que el alumno tenga una visión general de la Programación Orientada a Objetos.

- 1.1. Ventajas de la POO
- 1.2. Objeto
- 1.3. Clase
- 1.4. Método
- 1.5. Mensaje
- 1.6. Herencia
- 1.7. Abstracción
- 1.8. Encapsulamiento
- 1.9. Polimorfismo

2. REPRESENTACIÓN EN UML 5 hrs

Objetivo: Que el alumno sea capaz de modelar un sistema orientado a objetos simple.

- 2.1. Diagramas de Estructura
- 2.2. Diagramas de Comportamiento
- 2.3. Diagramas de Interacción

3. POO CON C++ 12 hrs

Objetivo: Que el alumno conozca las sentencias y funciones básicas equivalentes de C en C++, y sea capaz de crear programas con POO.

- 3.1. Migración de C a C++
- 3.2. Conceptos de POO en C++
 - 3.2.1. Creación de clases
 - 3.2.2. Constructores y destructores
 - 3.2.3. Control de acceso a miembros
 - 3.2.4. Creación de objetos

- 3.3. Manejo de objetos con memoria dinámica
- 3.4. Sobrecarga
- 3.5. Herencia
- 3.6. Polimorfismo
- 3.7. Genericidad
- 3.8. Creación de bibliotecas de clases

4. EXCEPCIONES 2 hrs

Objetivo: Que el alumno conozca el manejo de excepciones y su uso en la recuperación de errores.

- 4.1. Introducción a las excepciones
- 4.2. Sentencias try, throw y catch
- 4.3. Manejo de excepciones
- 4.4. Recuperación de errores usando excepciones

II PROGRAMACIÓN VISUAL

5. PLATAFORMA DE DESARROLLO 2 hrs

Objetivo: Que el alumno conozca un entorno de desarrollo visual, y sea capaz de crear proyectos.

- 5.1. Introducción a la plataforma de desarrollo
- 5.2. Conceptos de programación orientada a eventos
- 5.3. Creación de proyectos con interfaz gráfica

6. DESARROLLO DE APLICACIONES CON GUI 6 hrs

Objetivo: Que el alumno sea capaz de realizar proyectos de programación utilizando elementos de interfaz gráfica.

- 6.1. Controles principales
- 6.2. Manejo de eventos
- 6.3. Manejo de errores usando biblioteca de excepciones
- 6.4. Ejemplos de aplicación
- 6.5. Proyecto

METODOLOGÍA

En la primera parte del curso se presentan los conceptos básicos de Programación Orientada a Objetos. Se sugiere el uso de una versión simplificada de UML para representar las clases y objetos de manera gráfica, con el fin de agregar claridad a los conceptos.

En la segunda parte del curso se introducen conceptos de programación visual, para la generación de interfaces de usuario gráficas (GUI). Se muestran ejemplos de aplicación y se realiza un proyecto con el fin de reforzar los conceptos.

A lo largo del curso se hace énfasis en la importancia del diseño para evitar errores y en el análisis del código con el fin de detectar errores (prevención y corrección de errores).

La materia se imparte en un salón con pizarrón y cañón. En las sesiones se alternan actividades en las cuales se trabaja en el pizarrón y/o en papel, con actividades en las cuales se implementan en computadora algunos de los algoritmos desarrollados.

EVALUACIÓN

De cada tema visto en clase se realizarán actividades y se dejarán trabajos de aplicación relacionados con la Ingeniería y otras disciplinas, de tal manera que para aprobar la materia sea requisito haber presentado por lo

menos el 80% de estos trabajos. Los trabajos y actividades constituirán el 20% de cada calificación parcial y el 80% lo constituirá el examen parcial.

BIBLIOGRAFÍA

BIBLIOGRAFÍA BÁSICA

Apuntes del curso

Deitel & Deitel. "Cómo programar en C/C++ y Java", 6ª Edición. Ed. Pearson, 2008.

Joyanes Aguilar, Luis, Ignacio Zahonero Martínez, "Programación en C, C++, Java y UML", McGraw-Hill Education, 2ª edición, 2014.

BIBLIOGRAFÍA COMPLEMENTARIA

Kernighan & Ritchie. "El lenguaje de programación C", 2ª Edición, Prentice Hall, 1991.

<http://cplusplus.com/>