

FACULTAD DE INGENIERÍA

CENTRO DE INVESTIGACIÓN Y ESTUDIOS DE POSGRADO



Nombre de la materia: TALLER DE PROGRAMACIÓN
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
Básica obligatoria
No. de créditos aprobados:
Fecha última de Revisión Curricular: Septiembre 2012

JUSTIFICACIÓN DEL CURSO

Esta materia está dirigida a desarrollar el razonamiento lógico para la resolución de problemas, proporcionar una base firme para el conocimiento de lenguajes de programación acordes a la programación estructurada e introducir el uso del Lenguaje C en la solución de problemas relacionados con la ingeniería.

Adicionalmente 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.

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 estructurada.

Además se busca el conocer y aplicar los conceptos principales de la programación orientada a objetos, y la creación de bibliotecas de clases.

CONTENIDO TEMÁTICO

I PROGRAMACIÓN ESTRUCTURADA

1. INTRODUCCIÓN A LA PROGRAMACIÓN 1 hr

Objetivo: Que el alumno conozca los conceptos básicos relacionados con la computadora, plataforma utilizada para la realización de programas.

- 1.1. Categorías de computadoras
- 1.2. Partes de una computadora
- 1.3. Tipos de software
- 1.4. Introducción a la lógica booleana
- 1.5. Diagramas de flujo
- 1.6. Algoritmos

2. INTRODUCCIÓN AL LENGUAJE C 1 hr

Objetivo: Que el alumno conozca la estructura básica de un programa en C.

- 2.1. Historia de C
- 2.2. Estructura de un programa en C
- 2.3. Proceso de compilación y ligado

3. CONCEPTOS BÁSICOS 2 hrs

Objetivo: Que el alumno conozca los elementos básicos del lenguaje de programación C.

- 3.1. Tipos de datos
- 3.2. Variables
- 3.3. Constantes
- 3.4. Operadores y expresiones
- 3.5. Cadenas de caracteres

3.6. Entrada/salida por consola

4. ESTRUCTURAS DE CONTROL DE FLUJO 3 hrs

Objetivo: Que el alumno sea capaz de aplicar las estructuras de control de flujo en la solución de problemas de ingeniería y problemas en general.

- 4.1. Estructuras de selección
- 4.2. Estructuras de repetición
- 4.3. Sentencias break y continue

5. ARREGLOS 2 hrs

Objetivo: Que el alumno sea capaz de almacenar datos en arreglos, con el fin de que sean utilizados en la solución de problemas.

- 5.1. Introducción a los arreglos
- 5.2. Arreglos unidimensionales
- 5.3. Arreglos bidimensionales

6. FUNCIONES 2 hrs

Objetivo: Que el alumno sea capaz de estructurar programas utilizando funciones, con el fin de facilitar su mantenimiento e incrementar la reutilización de código.

- 6.1. Formato
- 6.2. Definición y llamada
- 6.3. Paso de argumentos por valor
- 6.4. Ámbito de variables
- 6.5. Funciones que regresan valores
- 6.6. Arreglos como parámetros

7. MEMORIA DINÁMICA 2 hrs

Objetivo: Que el alumno sea capaz de utilizar apuntadores y memoria dinámica para el manejo eficiente de la memoria.

- 7.1. Concepto de apuntador
- 7.2. Aritmética de apuntadores
- 7.3. Manejo de memoria dinámica
- 7.4. Paso de argumentos por referencia
- 7.5. Arreglos y apuntadores

8. ESTRUCTURAS 1 hr

Objetivo: Que el alumno conozca el uso de estructuras para el almacenamiento de datos relacionados entre sí.

- 8.1. Estructuras
- 8.2. Estructuras como parámetros
- 8.3. Arreglos de estructuras
- 8.4. Apuntadores a estructuras

9. RECURSIÓN 1 hr

Objetivo: Que el alumno conozca el concepto de recursión, así como sus ventajas y desventajas.

- 9.1. Concepto
- 9.2. Ejemplos clásicos

II PROGRAMACIÓN ORIENTADA A OBJETOS

10. INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA A OBJETOS (POO) 2 hrs.

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

- 10.1. Ventajas de la POO
- 10.2. Objeto
- 10.3. Clase
- 10.4. Método
- 10.5. Mensaje
- 10.6. Herencia
- 10.7. Abstracción
- 10.8. Encapsulamiento
- 10.9. Polimorfismo

11. REPRESENTACIÓN EN UML 2 hrs

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

- 11.1. Diagramas de Estructura
- 11.2. Diagramas de Comportamiento
- 11.3. Diagramas de Interacción

12. POO CON C++ 6 hrs

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

- 12.1. Migración de C a C++
- 12.2. Conceptos de POO en C++
 - 12.2.1. Creación de clases
 - 12.2.2. Constructores y destructores
 - 12.2.3. Control de acceso a miembros
 - 12.2.4. Creación de objetos
- 12.3. Manejo de objetos con memoria dinámica
- 12.4. Sobrecarga
- 12.5. Herencia
- 12.6. Polimorfismo
- 12.7. Genericidad
- 12.8. Creación de bibliotecas de clases

13. EXCEPCIONES 2 hrs

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

- 13.1. Introducción a las excepciones
- 13.2. Sentencias try, throw y catch

- 13.3. Manejo de excepciones
- 13.4. Recuperación de errores usando excepciones

III PROGRAMACIÓN VISUAL

14. PLATAFORMA DE DESARROLLO 2 hrs

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

- 14.1. Introducción a la plataforma de desarrollo
- 14.2. Conceptos de programación orientada a eventos
- 14.3. Creación de proyectos con interfaz gráfica

15. DESARROLLO DE APLICACIONES CON GUI

3 hrs

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

- 15.1. Controles principales
- 15.2. Manejo de eventos
- 15.3. Manejo de errores usando biblioteca de excepciones
- 15.4. Ejemplos de aplicación
- 15.5. Proyecto

METODOLOGÍA

La primera parte de cada tema se enfoca a que el alumno desarrolle el razonamiento lógico mediante el análisis de problemas sencillos. Se hace énfasis en el proceso de resolución de problemas con programación, realizando inicialmente el análisis del problema, para posteriormente proceder a la elaboración de un diagrama de bloques o algoritmo, a partir del cual se implemente la solución en lenguaje C.

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.

En la segunda 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 tercera 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.

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 BASICA

Apuntes del curso

Deitel, Deitel. "Cómo programar en C/C++ y Java", 4ª Edición. Ed. Pearson.

BIBLIOGRAFÍA COMPLEMENTARIA

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

<http://cplusplus.com/>