UNIVERSIDAD DE SONORA
Unidad Regional Centro
División de Ingeniería
Departamento de Ingeniería Industrial
LICENCIATURA INGENIERÍA EN MECATRÓNICA

Nombre de la Asignatura: PROGRAMACIÓN PARA INGENIEROS II					
Clave: 4648	**Créditos:** 7	**Horas totales:** 80	**Horas Teoría:** 2	**Horas Práctica:** 2	**Horas Semana:** 5
Modalidad: Presencial			**Eje de formación:** Profesionalizante		
Elaborado por: M.C. JORGE FRANCO ROMERO AGUILAR					
Antecedente: Programación para Ingenieros I			**Consecuente**:		
Carácter: Obligatoria			**Departamento de Servicio:** Ingeniería Industrial		

Propósito:

Aportar los conceptos, conocimientos y habilidades básicas de solución de problemas mediante computadoras aplicando el paradigma de programación orientado a objetos.
Esta asignatura permite el desarrollo de atributos de competencias y habilidades como: identificar las ideas clave en un texto o discurso oral e infiere conclusiones a partir de ellas, enfrentar las dificultades que se le presentan y es consciente de sus valores, fortalezas y debilidades, analizar críticamente los factores que influyen en su toma de decisiones, estructurar ideas y argumentos de manera clara, coherente y sintética, proponer maneras de solucionar un problema o desarrollar un proyecto en equipo, definiendo un curso de acción con pasos específicos, expresar ideas y conceptos mediante representaciones lingüísticas, matemáticas o gráficas. En el campo profesional las capacidades a desarrollar son diseñar, codificar y configurar piezas o componentes de software.

I. Contextualización

Introducción:

La asignatura desarrolla las competencias especializantes para desarrollar programas de computadora de propósito general bajo el paradigma de programación orientada a objetos utilizando un lenguaje de alto nivel como el C++ (C más más).

Para comenzar a estudiar cualquier lenguaje de programación se debe conocer cuáles son los conceptos que lo caracterizan, es decir, el tipo de programación que vamos a poder realizar con él. Como el C++ incorpora características nuevas respecto a lenguajes como Pascal o C, por tanto, en primer lugar, se da una descripción a los conceptos de este lenguaje, revisando los paradigmas de programación y con enfoque en la evolución desde la programación funcional a la programación orientada a objetos. Luego se estudia el lenguaje, primero se ven sus características funcionales y después se estudia las extensiones que dan soporte a la programación orientada a objetos.

En la unidad didáctica I. “Introducción al lenguaje de programación C++”. Recorrido por los principales

conceptos y rasgos del lenguaje de programación C++.

En la unidad didáctica II. "Paradigmas de programación". En esta unidad se presentan los diversos paradigmas de programación enfocándose en la orientación a objetos y los conceptos fundamentales de este.

En la unidad didáctica III. "Clases y objetos". Utilizar diagramas de clases para modelar clases y elaborar los códigos correspondientes utilizando el lenguaje de programación C++.En la unidad didáctica incluye "Herencia". Comprender el concepto de Herencia simple y múltiple en el diseño de clases y su implementación en lenguaje C++.

En la unidad didáctica IV. "Polimorfismo". Comprender el concepto de polimorfismo y sus diversas formas de implementación en lenguaje C++.

En la unidad didáctica V. "Encapsulación". Identificar la necesidad del uso de la encapsulación y aplicarla en el diseño de programas con elementos de seguridad.

En la unidad didáctica VI. "Principios de Adquisición de Datos". Conocer los conceptos y elementos básicos sobre señales, sensores y actuadores.

Perfil del(los) instructor(es):	Ingeniero en Sistemas Computacionales, Ingeniero en Electrónica o Eléctrico, Ingeniero en Sistemas de Información, Ingeniero en Mecatrónica, con experiencia en el desarrollo de aplicaciones orientados a objetos y conocimientos básicos de electrónica digital.

II. Competencias a lograr

Competencias genéricas a desarrollar:

- **Capacidad de aprender y actualizarse permanentemente.** Articula saberes de diversos campos y establece relaciones entre ellos y su vida cotidiana.
- **Trabajo colaborativo.** Asume una actitud constructiva, congruente con los conocimientos y habilidades con los que cuenta dentro de distintos equipos de trabajo.
- **Capacidad para la toma de decisiones.** Evalúa y sopesa información importante para identificar los aspectos relevantes. Define la prioridad para la solución del problema en términos de impacto y urgencia.
- **Capacidad para realizar investigación básica y aplicada.** Identifica los sistemas y reglas o principios medulares que subyacen a una serie de fenómenos.
- **Competencia Digital.** Aplica herramientas digitales para el pensamiento reflexivo, la creatividad y la innovación.

Competencias específicas:

- HABILIDAD PARA DESARROLLAR HARDWARE Y/O SOFTWARE PARA ESPECIFICACIONES ESPECÍFICAS DE CONTROL Y AUTOMATIZACIÓN
  - Desarrollar una lógica de pensamiento estructurada, para la comprensión del funcionamiento de los diferentes lenguajes de programación de alto nivel, indispensables para el desarrollo de tecnología informática.
- HABILIDAD PARA DESARROLLAR SISTEMAS DE CONTROL Y AUTOMATIZACIÓN

Aprender técnicas de sistemas computacionales aplicadas al desarrollo de sistemas mecatrónicos

automáticos controlados por computadora a través de interfaces electrónicas, incluyendo interfaces para la interacción hombre máquina.

Objetivo General:

Diseñar y codificar en el lenguaje C++ programas orientados a objetos para computadoras de propósito general para desarrollar aplicaciones de ingeniería. Programar utilizando diferentes algoritmos hasta obtener el programa ejecutable; utilizando los conceptos de programación, asimismo, aprender las estructuras básicas: sentencias condicionales, sentencias de repetición, diseño de funciones y procedimientos, todos necesarios en uso del lenguaje C++.

Objetivos Específicos:

1. Desarrollar algoritmos de solución a problemas diversos.
2. Aplicar las buenas prácticas en el desarrollo de programas.
3. Compilar, depurar y ejecutar programas.
4. Aplicar los principios del paradigma de programación orientados a objetos para desarrollar programas en el lenguaje C++.
5. Identificar los principios y conceptos sobre encapsulación, que pondrán en práctica en asignaturas posteriores.
6. Identificar los principios y conceptos sobre adquisición de datos que pondrán en práctica en asignaturas posteriores.

Unidades Didácticas:

Unidad Didáctica I - INTRODUCCIÓN AL LENGUAJE DE PROGRAMACIÓN C++.
Unidad Didáctica II - PARADIGMAS DE PROGRAMACIÓN.
Unidad Didáctica III - CLASES Y OBJETOS.
Unidad Didáctica IV – POLIMORFISMO.
Unidad Didáctica V – ENCAPSULACIÓN.
Unidad Didáctica VI - PRINCIPIOS DE ADQUISICIÓN DE DATOS.

III. Didáctica del programa

Unidades Didácticas:

Unidad didáctica I. Introducción al lenguaje de programación C++. Se hace una revisión de los principales conceptos y rasgos del lenguaje de programación C++.

- Historia del Lenguaje C++ y su relación con C.
- Elementos de un programa en C++.

Unidad Didáctica II. Paradigmas de programación. En esta unidad se presentan los diversos paradigmas de programación enfocándose en la orientación a objetos y los conceptos fundamentales de este.

- Identificar los diferentes paradigmas de programación a lo largo de la historia.
- Conocer los pilares de la programación orientada a objetos.

- Abstracción.
- Herencia.
- Encapsulación.
- Polimorfismo.

Unidad Didáctica III. Clases y Objetos. Utilizar diagramas de clases para modelar clases y elaborar los códigos correspondientes utilizando el lenguaje de programación C++.

- Utilizar diagramas de clases (UML) para representar clases definiendo atributos y métodos (miembros).
- Construir clases utilizando el lenguaje de programación C++.

Unidad didáctica IV. Polimorfismo. Aplicar el concepto de polimorfismo e implementarlos utilizando el lenguaje C++. Comprender el concepto de polimorfismo y sus diversas formas de implementación en lenguaje C++.Comprender el concepto de Herencia simple y múltiple en el diseño de clases y su implementación en lenguaje C++.

- Sobrecarga de constructores.
- Sobrecarga de métodos.
- Sobrecarga de operadores.
- Sobreposición de métodos.

Unidad didáctica V. Encapsulación. Utilizar los modificadores de acceso.

- Public.
- Private.
- Protected.

Unidad didáctica VI. Principios de Adquisición de Datos. Identificar los principios para desarrollar software de adquisición de datos. Conocer los conceptos y elementos básicos sobre señales, sensores y actuadores.

- Señales, Sensores y acondicionamiento de señales.
- Sistemas de Adquisición de datos.
- Equipo de adquisición de datos para PC.
- Conversión analógica a digital.

Criterios de desempeño:

1. Desarrollo de análisis para resolución de problemas.
2. Vincular los problemas tratados en los libros en diferentes dispositivos para su programación.
3. Proponer varios métodos de solución de problemas:
    a. Realización de algoritmos propuestos en los ejemplos y los ejercicios del libro de texto.
    b. Realización de programas por parte de los alumnos supervisados y asistidos por el instructor.

c. Desarrollo de programas por parte de los alumnos que se realicen en extra clase.
4. Participación activa en clase y a los laboratorios, la asistencia es muy importante, estudiar en equipo.

Experiencias de Enseñanza / procesos y objetos de aprendizaje:

1. Exposición y presentación oral de los temas con soporte en el uso de las TIC´s.
2. Brindar asesoría y/o retroalimentación en el momento de desarrollo práctico de las actividades.
3. Apoyar el proceso de aprendizaje utilizando plataformas en línea.
4. Desarrollar las habilidades de solución de problemas por computadora.

Experiencias de aprendizaje:

1. Lecturas previas de los materiales para contenidos nuevos.
2. Investigación de artículos de divulgación científica, búsquedas en internet y visualización de videos. previamente seleccionados relacionados con los temas de la asignatura.
3. Realización de ejercicios para la solución de problemas.
4. Diseño e implementación de prototipos.
5. Análisis de la información para la construcción de modelos de solución.
6. Exposición de casos y trabajo en equipo.
7. Utilización de herramientas tecnológicas como auxiliares para la práctica de la programación de computadoras.
8. Realización de prácticas de laboratorio de computadoras.

Recursos didácticos y tecnológicos (material de apoyo):

1. Computadora.
2. Video Proyector.
3. Pintarrón.
4. Se sugiere el uso de una plataforma on-line para dar seguimiento del programa y publicar evaluaciones.
5. Software:
   a. Se sugieren programas como StarUML para elaborar diagramas de clases.
   b. Compilador de C++.
   c. Ambiente de desarrollo Integrado (IDE).
6. En la unidad correspondiente VI es necesario utilizar componentes de hardware para adquirir datos, se sugiere la plataforma Arduino y sensores para pruebas.

Bibliografía	***Básica/ Complementaria***
Bjarne Stroustrup. (1993). El C++ Lenguaje de Programación. México: Edit. Addison-Wesley/Díaz de Santos.	***Básica***
Deitel & Deitel (2010). C++ Como programar. México: Edit. Pearson Educación	***Complementaria***
Mesurement Computing.(2012). Data Acquisition, Handbook.	***Complementaria***

IV. Evaluación Formativa de las Competencias

#	Tipo (C,H,A)	Evidencias a evaluar	Criterios de evaluación	Técnicas e Instrumentos de Evaluación	Ponderación %
1	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos con relación a las unidades I y II	Examen escrito	20 %
2	H, A	Utilización de herramientas tecnológicas para la solución de problemas	Se evaluará la capacidad en relación al trabajo en equipo y análisis de ejemplos	Diseño y presentación de ejemplos	20 %
3	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación con las unidades III y IV	Examen escrito	20 %
4	H, A	Ejercicios realizados en clases	Se evaluarán los conocimiento, habilidades y actitudes en la realización de prácticas de laboratorio	Evidencias de práctica de laboratorio	15 %
5	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos con relación a las unidades V y VI	Examen escrito	15 %
6	H, A	Asistencia, participación en clase y trabajo en equipo	Se evaluarán las habilidades de comunicación, organización y actitudes de trabajo y compromiso del alumno	Participación en clases y asistencia	10 %
Total					**100 %**

C: Conocimientos H: Habilidades A: Actitudes