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

Nombre de la Asignatura: PROGRAMACIÓN PARA INGENIEROS I					
Clave: 4642	**Créditos:** 7	**Horas totales:** 80	**Horas Teoría:** 2	**Horas Práctica:** 2	**Horas Semana:** 5
Modalidad: Presencial			**Eje de formación:** Básico		
Elaborado por: JORGE FRANCO ROMERO AGUILAR, HÉCTOR MARIO RAMÍREZ PONCE DE LEÓN					
Antecedente: Ninguna			**Consecuente**: Programación para Ingenieros II, Análisis Numérico y Programación		
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 estructurado y utilizando algorítmicos para diseño y posteriormente un lenguaje de alto nivel para la creación de programas. 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. La asignatura busca también desarrollar capacidades profesionales como: diseñar, codificar y configurar piezas o componentes de software.

I. Contextualización

Introducción:

El Ingeniero en Mecatrónica requiere una formación integral conformada por varias disciplinas, entre estas se encuentran las tecnologías de la información, en particular la Ingeniería de Software. La asignatura propone las competencias básicas y profesionalizantes que proporcionan los criterios generales para la solución de problemas por computadora, obteniendo como producto piezas de software. La asignatura busca adquirir competencias respecto a la lógica de programación y la solución de problemas mediante algoritmos que son de gran utilidad para el estudiante, éstas le sirven tanto para afrontar materias relacionadas con la programación, como las que no, puesto le brindan herramientas que facilitan el modelado y construcción de soluciones a problemas diversos.

Se propone el uso del lenguaje de programación C como herramienta única para esta asignatura y queda a consideración del docente la elección del ambiente de desarrollo (IDE y compilador).

Las unidades didácticas se describen a continuación.

En la unidad didáctica I. “Introducción a la programación”. Retomar conocimientos previos sobre sistemas de cómputo y adquirir los conceptos básicos.

En la unidad didáctica II. “Principios de programación”. Abordar los contenidos básicos para el diseño de soluciones por computadora.

En la unidad didáctica III. “Estructuras de control y representación Algorítmica”. Aplicar la metodología de solución de problemas por computadora diseñar soluciones algorítmicas a problemas propuestos con el enfoque de programación estructurado.

En la unidad didáctica IV. “Introducción al lenguaje de programación C”. Conocer historia del lenguaje y los componentes de los programas.

En la unidad didáctica V. “Desarrollo de programas estructurados”. Elaborar programas utilizando las diferentes estructuras de control y aplicando las buenas prácticas de programación

En la unidad didáctica VI. “Estrategia divide y vencerás”. Aplicar el uso de funciones para segmentar código haciendo más sencillo el desarrollo de problemas complejos.

En la unidad didáctica VII. “Estructuras de datos complejas”. Utilizar arreglos en la solución de problemas por computadora.

En la unidad didáctica VIII. “Punteros o apuntadores”. Utilizar apuntadores C.

En la unidad didáctica IX. “Estructuras de datos dinámicas”. Comprender y utilizar estructuras de datos dinámicas en programas en C.

En la unidad didáctica X. “Persistencia de datos”. Utilizar archivos para almacenar datos de forma permanente.

Perfil del(los) instructor(es):	Deberá de tener experiencia en área de ciencias computacionales, sistemas de información y en particular con el desarrollo de software desde varios paradigmas de programación. Se sugieren dominio y experiencia de desarrollo en lenguajes de programación como C, C++ o Java.

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 programas en el lenguaje C para computadoras, con propósitos generales de ingeniería. Formar una lógica de pensamiento estructurada, para la comprensión del funcionamiento de un lenguaje de programación de alto nivel, indispensable para el desarrollo de tecnología mecatrónica.

Objetivos Específicos:

1. Desarrollar algoritmos de solución a problemas diversos.
2. Conocer la arquitectura computadora y los componentes esenciales para escribir un programa en C.
3. Aplicar las buenas prácticas en el desarrollo de programas.
4. Aprender escribir expresiones matemáticas utilizando la sintaxis de C.
5. Aprender a compilar, depurar y ejecutar programas.
6. Utilizar las estructuras de control utilizando el lenguaje C.
7. Utilizar funciones en la construcción de programas en C.
8. Aplicar el uso de punteros y estructuras de datos estáticas y dinámicas.
9. Aprende a crear, escribir y leer archivos de datos usando el lenguaje C.
10. Realizar programas que almacenan datos de forma permanente en archivos.

Unidades Didácticas:

Unidad didáctica I - INTRODUCCIÓN A LA PROGRAMACIÓN.
Unidad didáctica II - PRINCIPIOS DE PROGRAMACIÓN.
Unidad didáctica III - ESTRUCTURAS DE CONTROL Y REPRESENTACIÓN ALGORÍTMICA.
Unidad didáctica IV - INTRODUCCIÓN AL LENGUAJE DE PROGRAMACIÓN C.
Unidad didáctica V - DESARROLLO DE PROGRAMAS ESTRUCTURADOS.
Unidad didáctica VI - ESTRATEGIA DIVIDE Y VENCERÁS.
Unidad didáctica VII - ESTRUCTURAS DE DATOS COMPLEJAS.
Unidad didáctica VIII - PUNTEROS O APUNTADORES.
Unidad didáctica IX - ESTRUCTURAS DE DATOS DINÁMICAS.
Unidad didáctica X - PERSISTENCIA DE DATOS.

III. Didáctica del programa

Unidades Didácticas:

Unidad didáctica I. Introducción a la programación
Retomar conocimientos previos sobre sistemas de cómputo e identificar y aplicar los conceptos de:

- Problemas.
- Algoritmos.
- Paradigmas.
- Lenguajes de programación.

Unidad didáctica II. Principios de programación
Abordar los contenidos básicos para el diseño de soluciones por computadora.

- Identificar los elementos que componen un programa desde el enfoque estructurado.
- Utilizar el método de solución de problemas por computadora.
- Identificar los elementos típicos o primitivos de los lenguajes: Estructuras de datos, Operaciones de entrada y salida y estructuras de control.
- Definir correctamente las estructuras de datos: variables y constantes, tipos e identificadores aplicando las buenas prácticas de programación.
- Aplicar los operadores y encontrar la solución de expresiones matemáticas utilizando operadores aritméticos, lógicos y relacionales.

Unidad didáctica III. Estructuras de control y representación algorítmica
Bajo la metodología de solución de problemas por computadora diseñar soluciones algorítmicas a problemas propuestos.

- Conocer los lenguajes algorítmicos: pseudocódigo y diagramas de flujo.
- Utilizar los lenguajes algorítmicos para diseñar soluciones a problemas secuenciales.
- Elaborar soluciones algorítmicas para problemas con decisión o selección.
- Construir soluciones algorítmicas para problemas utilizando ciclos o repetición.

Unidad didáctica IV. Introducción al lenguaje de programación C
Conocer el lenguaje de programación C para posteriormente utilizarlo en el desarrollo de programas.

- Conocer la historia del lenguaje C, sintaxis y palabras reservadas.
- Conocer y utilizar los entornos integrados de desarrollo.
- Escribir el primer programa en C y definir sus componentes principales.
- Utilizar el lenguaje para definir estructuras de datos utilizando las convenciones para notación de constantes, variables y funciones.
- Documentar el código utilizando comentarios.

Unidad didáctica V. Desarrollo de programas estructurados
Elaborar programas utilizando las diferentes estructuras de control y aplicando las buenas prácticas de programación.

- Construir programas utilizando las estructuras de control secuencial.
- Construir programas utilizando la estructura de control de condicional.
- Construir programas utilizando ciclos.

Unidad didáctica VI. Estrategia "divide y vencerás"
Conocer y aplicar el uso de funciones para segmentar código haciendo más sencillo el desarrollo de problemas complejos.

- Conocer los componentes de una función y aplicarla correctamente en los programas propios.
- Utilizar funciones que regresan valor.
- Aplicar el uso de funciones que retornan valor.
- Utilizar argumentos y parámetros en las funciones.

Unidad didáctica VII. **Estructuras de datos complejas**
Aplicar el uso de arreglos en la solución de problemas por computadora.

- Conocer la definición de estructuras de datos estáticas en C.
- Utilizar arreglos de una dimensión en la elaboración de programas en C.
- Utilizar arreglos bidimensionales en la elaboración de programas en C.

Unidad didáctica VIII. Punteros o apuntadores
Conocer y aplicar o apuntadores C.

- Identificar los conceptos de apuntadores y direcciones de memoria.
- Conocer la sintaxis en C correspondiente tipo de datos punteros.
- Utilizar punteros como parámetros: Paso de parámetros por referencia.

Unidad didáctica IX. Estructuras de datos dinámicas
Utilizar estructuras de datos dinámicas en la construcción de programas en C.

- Diferenciar entre estructuras estáticas y dinámicas.
- Utilizar las librerías estándar para el uso de memoria dinámica.
- Aplicar el uso de memoria dinámica en la creación de vectores.
- Utilizar arreglos de caracteres dinámicas como cadenas.

Unidad didáctica X. Persistencia de datos
Realizar programas que almacenan datos de forma permanente en archivos.

- Identificar los tipos de archivos.
- Utilizar apuntadores a archivos y aplicar los tipos de acceso.
- Realizar programas que escriben datos en archivos.
- Realizar programas que leen datos en archivos.
- Realizar programas que actualizan datos en archivos.

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.
    b. Realización de programas aplicando buenas prácticas de programación.
    c. Realización de programas utilizando estructuras estáticas y dinámicas.
    d. Desarrollo de programas por parte de los alumnos que se realicen en extra clase.
4. Participación activa en clase y 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.

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 Raptor Flow Chart para elaborar diagramas de flujo.
    b. Compilador de C.
    c. Ambiente de desarrollo Integrado (IDE).

Bibliografía	*Básica/ Complementaria*
Schildt Herbet. (2004). C Manual de Referencia. Edit. McGraw-Hill.	***Básica***
Kernighan & Ritchie. (1991). El Lenguaje de programación C. Edit. Pearson Education	***Básica***
Byron Gottfri. (2008). Programación en C. México: Edit. McGraw-Hill.	***Básica***
Romero Jorge. (2016). Desarrolla e instala Software de aplicación utilizando programación estructurada. Edit. Fondo de Cultura Económica	***Complementaria***
López Leobardo. (2010). Programación Estructurada en lenguaje C. Edit. AlfaOmega	***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 en 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, análisis, exposición y organización de ideas	Diseño y presentación de casos de estudio	20 %
3	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación a las unidades III y IV	Examen escrito	20 %
4	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación a las unidades V y VI	Examen escrito	20 %
5	C	Examen parcial	Se evaluará el nivel de conocimientos adquiridos en relación a las unidades VII, VIII. IX Y X	Examen escrito	20 %
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