

Programa de asignatura por competencias de educación superior

Sección I. Identificación del Curso

Tabla 1. Identificación de la Planificación del Curso.

Actualización:	Marzo 28, 2022				
Carrera:	Ingeniería en Diseño Electrónico y Sistemas Inteligentes	Asignatura:	Programación estructurada		
Academia:	Electrónica /	Clave:	19SDE01		
Módulo formativo:	Electrónica Digital	Seriación:	-		
Tipo de curso:	Presencial	Prerrequisito:	-		
Semestre:	Primero	Créditos:	6.75	Horas semestre:	108 horas
Teoría:	2 horas	Práctica:	4 horas	Trabajo indpt.:	0 horas
				Total x semana:	6 horas

Sección II. Objetivos educacionales

Tabla 2. Objetivos educacionales

Objetivos educacionales		Criterios de desempeño	Indicadores
OE2	Los egresados implementarán proyectos especializados en sistemas complejos de control y electrónicos en organizaciones públicas o privadas.	Conocerán e implementarán las teorías de gestión y dirección aplicadas a proyectos.	50% de los egresados conocerán diferentes teorías de gestión y dirección de proyectos
OE3	Los egresados resolverán problemas en el ámbito industrial con el desarrollo de proyectos de sistemas electrónicos.	Conocerán e implementarán las metodologías de análisis y diseño de sistemas electrónicos.	30% de los egresados analizarán un sistema electrónico.
OE5	Los egresados aplicarán y administrarán sistemas electrónicos y de control de manera ética, con responsabilidad social para contribuir al desarrollo sustentable.	Conocerán e implementarán modelos de sistemas electrónicos y de control.	30% de los egresados aplicarán modelos de sistemas electrónicos o de control.
OE6	Los egresados se integrarán a redes de colaboración públicas o privadas para el desarrollo de proyectos tecnológicos nacionales e internacionales.	Se integrarán al trabajo colaborativo en instancias públicas (Conacyt) o privadas mediante las estadías, las materias de proyecto y el intercambio con otras instituciones.	30% de los egresados trabajarán de forma colaborativa en instancias públicas como Conacyt desarrollando proyectos.
OE1	Los egresados diseñarán y desarrollarán proyectos especializados en sistemas complejos de control y electrónicos en organizaciones públicas o privadas.	Conocerán y aplicarán la metodología de la formulación, diseño, implementación y evaluación de Proyectos de tipo Industrial y de tecnologías Electrónicas Emergentes.	40% de los Egresados serán capaces de formular proyectos Electrónicos.
Atributos de egreso de plan de estudios		Criterios de desempeño	Componentes
AE1	Aplicar los conocimientos de ciencias básicas, como la química, física y matemáticas, y las ciencias de la ingeniería para resolver problemas dentro del campo de la electrónica.	Conocerá los fundamentos y los elementos de la programación en Lenguaje C y a partir del análisis del enunciado, desarrollará la estrategia de solución, la prueba y validación de resultados, mediante programas de lectura, procesamiento y salida de datos.	1.1 Razonamiento Lógico. 1.2 Razonamiento matemático. 1.4 Diseño de Programa. 2.1 Identificadores. 2.2 Tipos de datos. 2.3 Variables y Constantes.



Continuación: Tabla 2. Objetivos educacionales (continuación)

No.	Atributos de egreso de plan de estudios	Criterios de desempeño	Componentes
			2.6 Palabras Reservadas. 2.8 Funciones de entrada. 2.9 Funciones de salida. 3.1 Contadores y Acumuladores. 3.2 Secuencias de Control. 4.1 Arreglos. 4.2 Funciones.
AE3	Implementar estrategias a partir del juicio ingenieril para sacar conclusiones y tomar decisiones a partir de análisis estadísticos y mejorar así la calidad de los procesos industriales.	Resolverá problemas de forma clara y sistematizada.	1.1 Razonamiento Lógico. 1.2 Razonamiento matemático. 1.4 Diseño de Programa. 4.2 Funciones.

Sección III. Atributos de la asignatura

Tabla 3. Atributos de la asignatura

Problema a resolver		
Adquirir las bases para la resolución de problemas, mediante la formulación y el desarrollo de pasos específicos, orientados a mejorar la claridad, calidad y tiempo de desarrollo de un programa de computadora.		
Atributos (competencia específica) de la asignatura		
Obtener los conocimientos fundamentales para las diversas aplicaciones de programación en las empresas, y que le permitan contar con las herramientas necesarias para estar en condiciones de elaborar, analizar e interpretar problemas a resolver utilizando lenguaje C para desarrollar programas.		
Aportación a la competencia específica		Aportación a las competencias transversales
Saber	Saber hacer	Saber Ser
<p>Conocer las herramientas necesarias para poder interpretar y analizar problemas.</p> <p>Conocer el alcance de la programación lógica estructurada.</p>	<p>Desarrollar programas de temáticas diversas conociendo el alcance de la programación lógica estructurada.</p> <p>Elaborar resolución de problemas mediante el desarrollo de programas en lenguaje C.</p>	<p>El alumno lleva a cabo las actividades tanto de manera individual como colaborativa, para un mejor desarrollo de las habilidades deseadas.</p>
Producto integrador de la asignatura, considerando los avances por unidad		
<p>Portafolio de evidencias:</p> <p>Ejercicios por escrito con el procedimiento que corresponda para resolución de problemas a partir de los enunciados.</p> <p>Realización de los programas de prueba que validen la solución de los problemas planteados.</p> <p>Evaluaciones acreditadas con un mínimo de eficiencia del 70%</p>		

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.1. Desglose específico de la unidad "Habilidades del pensamiento y conceptos básicos."

Número y nombre de la unidad: 1. Habilidades del pensamiento y conceptos básicos.				
Tiempo y porcentaje para esta unidad:		Teoría: 10 horas	Práctica: 17 horas	Porcentaje del programa: 25%
Aprendizajes esperados: Identificar los diferentes tipos de operaciones de conjuntos para su aplicación en el razonamiento lógico y matemático.				
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)
1.1 Razonamiento lógico 1.2 Razonamiento matemático 1.3 Diseño de programa 1.3.1 Análisis del problema 1.3.2 Algoritmos 1.3.3 Diagramas de flujo 1.3.4 Codificación	<p>Saber:</p> <ul style="list-style-type: none"> - Conocer la simbología utilizada en el desarrollo de diagramas de flujo. <p>Saber hacer:</p> <ul style="list-style-type: none"> - Resolver problemas determinados aplicando el razonamiento lógico y matemático y operaciones de conjuntos. - Abstractar, analizar y sintetizar información. Aplicar los conocimientos en la práctica. <p>Ser:</p> <ul style="list-style-type: none"> - Trabajo en forma autónoma. 	<ul style="list-style-type: none"> -Exposición. -Ejecución de prácticas de laboratorio. -Resolución de ejercicios. 	<p>Evaluación formativa:</p> <ul style="list-style-type: none"> - Reporte de investigación de algoritmos y diagramas. - Ejercicios programados y resolución de problemas. - Solución de problemas propuestos por el docente en el programa especificado por el profesor. <p>Evaluación sumativa:</p> <ul style="list-style-type: none"> - Evaluar los conocimientos adquiridos por el alumno mediante exámenes escritos. 	Solución de una necesidad real con la implementación de un programa a partir del lenguaje C++
Bibliografía				
<ul style="list-style-type: none"> - Ancona, M.A. (2011). Diseño de Algoritmos y su codificación en lenguaje C. México: Mc Graw Hill. - Aguilar, L.J. (1992). Metodología de la Programación y Programación Estructurada. México: Mc Graw Hill. - Byron, G. (2005). Programación en C. México: Mc Graw Hill. - Cairo, O. (2008). Metodología de la Programación. 3ra edición. México: Computec. - Alcalde, E.; García, M. (1990). Metodología de la Programación. México: Mc Graw Hill. - Aguilar, L.J. (1988). Problemas de la Metodología de la Programación. México: Mc Graw Hill. 				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.2. Desglose específico de la unidad "Introducción al lenguaje C."

Número y nombre de la unidad: 2. Introducción al lenguaje C.				
Tiempo y porcentaje para esta unidad:	Teoría: 10 horas Práctica: 17 horas Porcentaje del programa: 25%			
Aprendizajes esperados: Identificar los diferentes tipos de datos utilizados en la programación para la resolución de problemas.				
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)
2.1 Identificadores 2.2 Tipos de datos 2.3 Variables y constantes 2.4 Operadores y Jerarquías 2.5 Expresiones 2.6 Palabras reservadas 2.7 Comentarios 2.8 Funciones de entrada (scanf, gets) 2.9 Funciones de salida (printf, puts)	Saber: - Identificar los tipos de operadores existentes. - Identificar los conceptos de identificador, variables y constantes, así como describir las reglas de creación de un identificador. Saber hacer: - Aplicar la jerarquía de los operadores. Aplicar los conceptos de identificador, variables y constantes. - Abstraer, analizar y sintetizar información. - Aplicar los conocimientos en la práctica.	- Exposición. - Ejecución de prácticas de laboratorio. - Resolución de ejercicios.	Evaluación formativa: - Ejercicios programados y resolución de problemas. - Solución de problemas propuestos por el docente en el programa especificado por el profesor. Evaluación sumativa: - Evaluar los conocimientos adquiridos por el alumno mediante exámenes escritos.	Presentación de la primera parte del proyecto integrador, proponiendo una necesidad real para dar solución con la implementación de un programa a partir del lenguaje C++.



Continuación: Tabla 4.2. Desglose específico de la unidad "Introducción al lenguaje C."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	Identificar, plantear y resolver problemas. Ser: - Trabajo en forma autónoma.			
Bibliografía				
<ul style="list-style-type: none">- Ancona, M.A. (2011). Diseño de Algoritmos y su codificación en lenguaje C. México: Mc Graw Hill.- Aguilar, L.J. (1992). Metodología de la Programación y Programación Estructurada. México: Mc Graw Hill.- Byron, G. (2005). Programación en C. México: Mc Graw Hill.- Cairo, O. (2008). Metodología de la Programación. 3ra edición. México: Computec.- Alcalde, E.; García, M. (1990). Metodología de la Programación. México: Mc Graw Hill.- Aguilar, L.J. (1988). Problemas de la Metodología de la Programación. México: Mc Graw Hill.				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.3. Desglose específico de la unidad "Programación estructurada."

Número y nombre de la unidad: 3. Programación estructurada.							
Tiempo y porcentaje para esta unidad:		Teoría:	10 horas	Práctica:	17 horas	Porcentaje del programa:	25%
Aprendizajes esperados: Conocer el funcionamiento de las estructuras de control repetitivas para facilitar el procesamiento y flujo de datos en los programas.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
3.2 Estructuras de control. 3.2.1 Contadores y acumuladores. 3.2.2 Secuencial. 3.2.3 Selectivas. 3.2.4 Repetitivas (while, do-while, for).	Saber: - Conocer la sintaxis y forma de uso de las secuencias de control de decisión: if, if-else y switch. - Conocer la sintaxis y forma de uso de las secuencias de control de repetición: while, do-while y for. Saber hacer: - Resolver problemas determinados aplicando las estructuras de control selectivas y verificar la solución a través de una prueba de escritorio, así como problemas determinados aplicando las	- Exposición. - Ejecución de prácticas de laboratorio. - Resolución de ejercicios.	Evaluación formativa: - Ejercicios programados y resolución de problemas. - Solución de problemas propuestos por el docente en el programa especificado por el profesor. - Evaluación sumativa: - Evaluar los conocimientos adquiridos por el alumno mediante exámenes escritos.	Desarrollo de la segunda parte del proyecto integrador a partir de una solución con un programa utilizando la programación estructurada.			



Continuación: Tabla 4.3. Desglose específico de la unidad "Programación estructurada."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	estructuras de control Repetitivas. Resolver problemas determinados aplicando la estructura de control secuencia y verificar la solución a través de una prueba de escritorio. Abstractar, analizar y sintetizar información. Aplicar los conocimientos en la práctica. Ser: - Trabajo en forma autónoma.			
Bibliografía				
<ul style="list-style-type: none"> - Ancona, M.A. (2011). Diseño de Algoritmos y su codificación en lenguaje C. México: Mc Graw Hill. - Aguilar, L.J. (1992). Metodología de la Programación y Programación Estructurada. México: Mc Graw Hill. - Byron, G. (2005). Programación en C. México: Mc Graw Hill. - Cairo, O. (2008). Metodología de la Programación. 3ra edición. México: Computec. - Alcalde, E.; García, M. (1990). Metodología de la Programación. México: Mc Graw Hill. - Aguilar, L.J. (1988). Problemas de la Metodología de la Programación. México: Mc Graw Hill. 				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.4. Desglose específico de la unidad "Arreglos y funciones."

Número y nombre de la unidad: 4. Arreglos y funciones.							
Tiempo y porcentaje para esta unidad:		Teoría:	10 horas	Práctica:	17 horas	Porcentaje del programa:	25%
Aprendizajes esperados: Resolver problemas determinados aplicando arreglos unidimensionales verificando la solución a través de una prueba de escritorio.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
4.1 Arreglos 4.1.1 Arreglos unidimensionales 4.1.2 Arreglos bidimensionales 4.1.3 Arreglos o cadenas de caracteres 4.2 Funciones 4.2.1 Prototipo 4.2.2 Parámetros 4.2.3 Retorno de valor	Saber: - Conocer el prototipo de funciones (funciones sin paso de parámetro, funciones con paso de parámetro, funciones que no regresan valor, funciones que regresan valor). Saber hacer: - Resolver los problemas determinados aplicando arreglos bidimensionales, verificando la solución a través de los diferentes tipos y funciones. - Abstraer, analizar y sintetizar información. - Aplicar los conocimientos en la práctica.	- Exposición. - Ejecución de prácticas de laboratorio. - Resolución de ejercicios.	Evaluación formativa: - Ejercicios programados y resolución de problemas. - Solución de problemas propuestos por el docente en el programa especificado por el profesor. Evaluación sumativa: - Evaluar los conocimientos adquiridos por el alumno mediante exámenes escritos.	Elaboración de la etapa final del proyecto integrador, incorporando arreglos y funciones a su programa desarrollado y utilizando como base la programación orientada a objetos.			



Continuación: Tabla 4.4. Desglose específico de la unidad "Arreglos y funciones."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	Ser: - Trabajo en forma autónoma.			
Bibliografía				
<ul style="list-style-type: none">- Ancona, M.A. (2011). Diseño de Algoritmos y su codificación en lenguaje C. México: Mc Graw Hill.- Aguilar, L.J. (1992). Metodología de la Programación y Programación Estructurada. México: Mc Graw Hill.- Byron, G. (2005). Programación en C. México: Mc Graw Hill.- Cairo, O. (2008). Metodología de la Programación. 3ra edición. México: Computec.- Alcalde, E.; García, M. (1990). Metodología de la Programación. México: Mc Graw Hill.- Aguilar, L.J. (1988). Problemas de la Metodología de la Programación. México: Mc Graw Hill.				



V. Perfil docente

Tabla 5. Descripción del perfil docente

Perfil deseable docente para impartir la asignatura
<p>Carrera(s): Licenciatura o ingeniería en:</p> <ul style="list-style-type: none">-Ciencias de la ingeniería electrónica.-Comunicaciones y electrónica.-Electrónica.-Electrónica y comunicaciones.-Electrónica en computación.-Comunicaciones en eléctrica y electrónica.-Ingeniero en sistemas Computacionales o afín. o carrera afín<ul style="list-style-type: none">- Experiencia profesional relacionada con la materia a impartir.- Experiencia mínima de dos años- Ingeniera en sistemas computacionales, preferentemente Maestría relacionada con el área de conocimiento.