


ASIGNATURA DE PROGRAMACIÓN AVANZADA

1. Competencias	Desarrollar proyectos de automatización y control, a través del diseño, la administración y la aplicación de nuevas tecnologías para satisfacer las necesidades del sector productivo.
2. Cuatrimestre	Noveno
3. Horas Teóricas	30
4. Horas Prácticas	45
5. Horas Totales	75
6. Horas Totales por Semana Cuatrimestre	5
7. Objetivo de aprendizaje	El alumno conocerá un modelo de programación más avanzado y metodológico, utilizando la orientación a objetos y será capaz de construir aplicaciones que se ejecuten en modo texto, en modo gráfico e internet.

Unidades de Aprendizaje		Horas		
		Teóricas	Prácticas	Totales
I.	Paradigmas de la Programación Estructurada	7	9	16
II.	Elementos de un Lenguaje Imperativo de Programación (Control y Arreglos)	7	13	20
III.	Manipulación de Funciones	5	8	13
IV.	Estructuras	5	7	12
V.	Aplicación de Punteros	6	8	14
Totales		30	45	75


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA


UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Paradigmas de la Programación Estructurada
2. Horas Teóricas	7
3. Horas Prácticas	9
4. Horas Totales	16
5. Objetivo de la Unidad de Aprendizaje	El alumno construirá la estructura y formación de un programa con base en el paradigma orientado a objetos, aplicando los conceptos de la programación y estructuración avanzada

Temas	Saber	Saber hacer	Ser
1. Programación modular y programación estructurada.	Explicar la arquitectura de composición de los programas modulares, así como las características que lo componen.	Determinar los principales elementos que componen un programa modular.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.
2. Operadores de asignación y expresión.	Definir e identificar los operadores y los tipos de operadores que los componen.	Utilizar correctamente los operadores necesarios para el desarrollo de un programa.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.
3. Instrucciones de asignación, entrada/salida.	Explicar el método punto de entrada y salida del flujo de datos.	Ejecutar un método de diseño de lógica combinacional y secuencial para el desarrollo de un diseño.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


Temas	Saber	Saber hacer	Ser
4. Estructuras selectivas.	Listar y explicar las diferentes operaciones que permiten realizar una estructura selectiva y sus múltiples ventajas.	Simular y programar una secuencia en un programa mediante software dedicado, utilizando en el procedimiento estructuras para el almacenamiento y procesamiento de datos internos y externos.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
El alumno entrega los reportes de las prácticas desarrolladas con base en los temas manejados, los cuales cubren las bases principales de la programación orientada a objetos.	<ol style="list-style-type: none">1. Identificar las características del paradigma de la programación orientada a objetos, módulos, entradas, salidas, programación.2. Comprender la configuración de las funciones de la programación avanzada, determinando los parámetros adecuados para diversos procesos.3. Comprender la conexión y utilización de entradas y salidas de los procesos que se manejan en la computadora.	Lista de verificación y Guía de observación Proyecto

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


PROGRAMACIÓN AVANZADA

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
1. Demostración práctica 2. Prácticas de laboratorio 3. Aprendizaje basado en proyectos	Pizarrón, Cañón, Equipo de cómputo, Software de programación.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA


UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Elementos de un Lenguaje Imperativo de Programación (Control y Arreglos)
2. Horas Teóricas	7
3. Horas Prácticas	13
4. Horas Totales	20
5. Objetivo de la Unidad de Aprendizaje	El alumno empleará las técnicas de programación que integra el uso y desarrollo de métodos de control para aplicaciones y programas logrando la optimización, mejora e innovación de procesos.

Temas	Saber	Saber hacer	Ser
1. Estructura de ciclos y desarrollo de la programación mediante un software dedicado.	Describir y explicar la composición de las estructuras y composición de ciclos y sus diferentes tipos de módulos que lo integran.	Construir algoritmos en un lenguaje de programación, utilizando estructuras de programación : a) Funciones de comparación y control (if, while, do-while, for, switch case) b) Máquina de estados (polling, secuencia de anillo).	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


Temas	Saber	Saber hacer	Ser
2. Manipulación de arreglos unidimensionales, bidimensionales y multidimensionales .	Comprender el concepto de arreglo, como se concibe en memoria y forma de acceder a sus elementos, así como la entrada y salida de elementos propios del índice de un arreglo de diferentes tamaños.	Construir algoritmos con expresiones utilizando funciones y procedimientos para la creación de arreglos mediante herramientas para el agrupamiento, organización y clasificación de datos. a) Creación de funciones para el manejo de arreglos de una sola dimensión. b) Creación de funciones para el manejo de arreglos de dos dimensiones. c) Creación de funciones para el manejo de arreglos de múltiples dimensiones.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje Razonamiento deductivo Proactivo, Iniciativa, Dinámico.
3. Estructuras de control.	Identificar el manejo y utilización de los sistemas de control, así como la sintaxis y parámetros que permite su manejo.	Diseñar o implementar programas que permitan el manejo y uso de las estructuras de control para monitorear la entrada y salida de procesos en tiempo real.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>El alumno elaborará programas en un lenguaje orientado a objetos enfocando el desarrollo final con base en diagramas de flujo y estado que utilicen:</p> <ul style="list-style-type: none">- Distintos tipos de variables.- Operaciones aritméticas, lógicas y relacionales.- Control de flujo.- Ingreso y exhibición de datos.- Estructuras de comparación y Funciones.	<p>1.- identificar las principales arquitecturas de los programas vistos.</p> <p>2- Comprender los aspectos del entorno de programación orientado a objetos.</p> <p>3.- Analizar los Fundamentos de las instrucciones.</p> <p>4.-Reconocer las estructuras de programación y funciones en un entorno de programación.</p> <p>5.- Comprender el diseño de programas estructurados en tiempo real, así como el proceso de prueba y depuración de los programas.</p>	<p>Ejercicios prácticos y lista de verificación.</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


PROGRAMACIÓN AVANZADA

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Aprendizaje Basado en Proyectos. Prácticas de laboratorio.	Pizarrón, Cañón, Equipo de cómputo, Software de programación.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
------	----------------------	---------

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


X	X	
---	---	--

PROGRAMACIÓN AVANZADA


UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	III. Manipulación de Funciones
2. Horas Teóricas	5
3. Horas Prácticas	8
4. Horas Totales	13
5. Objetivo de la Unidad de Aprendizaje	El alumno conocerá la sintaxis y uso principal de la creación de funciones, para el uso recursivo y estructuración mediante la programación orientada a objetos.

Temas	Saber	Saber hacer	Ser
1. Forma general de una función.	Describir los tipos y características, así como los parámetros que debe contener dicha estructura, así como las ventajas de aplicarla.	1.- identificar las principales arquitecturas de las funciones. 2- Comprender los aspectos del entorno de programación para su creación. 3.- Analizar los Fundamentos de las instrucciones.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


Temas	Saber	Saber hacer	Ser
2. Prototipos de funciones	<p>Desarrollar un mapa conceptual de la arquitectura y funciones.</p> <p>Elaborar programas en lenguaje de programación enfocado al uso de funciones a partir de diagramas de flujo y estado que utilicen:</p> <ul style="list-style-type: none"> - Distintos tipos de variables. - Operaciones aritméticas, lógicas y relacionales. - Control de flujo. - Ingreso y exhibición de datos. - Estructuras de comparación y Funciones. 	<p>1.-Reconocer las estructuras de programación y funciones en un entorno de programación para la aplicación de nuevas tecnologías de información.</p> <p>2.- Comprender el diseño de programas estructurados en Maquina de estado, así como el proceso de prueba y depuración de los programas mediante software dedicado.</p>	<p>Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, Proactivo, Iniciativa, Dinámico.</p>
3. Argumentos y parámetros de las funciones	<p>Describir los módulos internos para el uso óptimo de las funciones en un lenguaje de programación, así como el manejo del concepto de recursividad.</p>	<p>Construir expresiones en el lenguaje de programación, utilizando los módulos internos:</p> <ul style="list-style-type: none"> a) Para el manejo de recursividad. b) Tipos de datos en las funciones. c) Módulos de tipo entero. d) Módulos de cadena de caracteres y void. 	<p>Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, Proactivo, Iniciativa, Dinámico.</p>

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
Funciones creadas por el usuario. Metodología de diseño. Archivo electrónico con el diagrama y la simulación. Resultado de la prueba en el sistema de desarrollo.	Reconocer una metodología de diseño para la creación de módulos y funciones con base en la arquitectura del lenguaje.	Ejecución de tareas. Lista de verificación y guía de observación.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


PROGRAMACIÓN AVANZADA

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Aprendizaje Basado en Proyectos. Prácticas de laboratorio	Pizarrón, Cañón, Equipo de cómputo, Software de programación.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
------	----------------------	---------

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


X	X	
---	---	--

PROGRAMACIÓN AVANZADA

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	IV. Estructuras
2. Horas Teóricas	5
3. Horas Prácticas	7
4. Horas Totales	12
5. Objetivo de la Unidad de Aprendizaje	El estudiante tendrá las bases para el desarrollo y operación de estructuras de datos para su implementación y aplicación de código recursivo.

Temas	Saber	Saber hacer	Ser
1. Acceso a los miembros de una estructura.	Explicar el entorno y la sintaxis de programación para la creación de estructuras en una clase objeto.	1. Identificar la sintaxis de una estructura en el lenguaje de programación. 3.-Diferenciar y utilizar los parámetros requeridos para el uso de clases.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.
2. Manejo de Estructuras con funciones.	Conocer el manejo de funciones estructuradas y los parámetros que deben de contener dichas clases creadas.	Establecer clases con estructuras sólidas y establecidas para la optimización y validación de procesos	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Temas	Saber	Saber hacer	Ser
3. Control del flujo de ejecución de un programa.	Describir los tipos y características del control de flujo en el proceso de compilación de un programa.	Calcular las estimaciones de la salida de los datos, los valores que llega a almacenar dichas variables y el manejo factible del debug.	Capacidad para el desarrollo de elementos de un programa óptimo. Aplicación de auto aprendizaje. Razonamiento deductivo, proactivo, iniciativa, dinámico.

PROGRAMACIÓN AVANZADA

PROCESO DE EVALUACIÓN


Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
---------------------------------	---------------------------------	--

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

<p>El alumno entrega proyecto, que describa el sistema de desarrollo aplicando programación estructurada que contenga:</p> <ul style="list-style-type: none"> -Planteamiento del problema. -Metodología de diseño. - Archivo electrónico con el diagrama y la simulación. - Resultado de las pruebas en el sistema de desarrollo. 	<p>1.- Comprender las principales ventajas y desventajas de la programación orientada a objetos.</p> <p>2.-Reconocer una metodología de diseño para la aplicación de módulos y optimización de procesos en la industria.</p> <p>3.- Comprender la construcción de un sistema de control digital para un proceso con base en la arquitectura de sistemas de información.</p>	<p>Ejecución de tareas.</p> <p>Lista de verificación y guía de observación.</p>
---	---	---

PROGRAMACIÓN AVANZADA

PROCESO ENSEÑANZA APRENDIZAJE


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Entregar un reporte final del proyecto que incluya: programas, diagramas de flujo, características, forma de desarrollo y conclusiones.	Pizarrón, Cañón, Equipo de cómputo, Software de programación.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X	X	


PROGRAMACIÓN AVANZADA

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	V. Aplicación de Punteros
2. Horas Teóricas	6
3. Horas Prácticas	8
4. Horas Totales	14
5. Objetivo de la Unidad de Aprendizaje	El alumno será capaz de resolver problemas de base científica mediante el diseño, codificación e implantación de programas de computadora mediante el uso de la memoria y sus diferentes formas de modificarla.

Temas	Saber	Saber hacer	Ser
1. Declaración e inicialización de variables de punteros.	Explicar los principales elementos y la arquitectura que conforman la inicialización de punteros y el almacenamiento de memoria dinámica en un programa.	Determinar los principales elementos que componen la creación de punteros y los almacenamientos de memoria dinámica.	Responsabilidad Analítico Proactivo, iniciativa, dinámico Orden y limpieza Creativo Trabajo en equipo Innovación Toma de decisiones
2. Llamar funciones por referencia.	Listar y explicar las diferentes formas de mandar a recursividad una función y el espacio en memoria que ocupa la arquitectura. Listar y explicar las diferentes operaciones aritméticas que soporta la utilización de funciones por referencia.	Programar una función por referencia y mandarla llamar varias veces en el mismo programa para hacer factible el paradigma de la programación orientada a objetos en un módulo de programación, utilizando memorias y registros internos del lenguaje de programación para el almacenamiento y procesamiento de datos internos y externos.	Responsabilidad Analítico Proactivo, iniciativa, dinámico Orden y limpieza Creativo Trabajo en equipo Innovación Toma de decisiones

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	


Temas	Saber	Saber hacer	Ser
3. Expresiones de apuntadores y aritmética de apuntadores.	Listar y explicar la arquitectura de apuntadores en la programación orientada a objetos ejemplos de control de dichos procedimientos.	Diseñar e implementar un sistema aplicando los llamados de memoria en la computadora en el manejo de alguna aplicación en la industria.	Responsabilidad Analítico Proactivo, iniciativa, dinámico Orden y limpieza Creativo Trabajo en equipo Innovación Toma de decisiones
4. Aplicación de llamada por referencia a funciones.	Listar y explicar los requerimientos y parámetros físicos que debe seguir la estructura final de un módulo programado, así como el manejo de los elementos de herencia, recursividad y polimorfismo para disminuir recursos físicos de hardware de un proceso a controlar.	Diseñar e implementar un sistema de control para procesos con arquitectura utilizando tecnologías de la información.	Responsabilidad Analítico Proactivo, iniciativa, dinámico Orden y limpieza Creativo Trabajo en equipo Innovación Toma de decisiones

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>El alumno entrega un proyecto, el cual contenga todos los elementos vistos en el curso y contemple las principales ventajas del uso de la programación orientada a objetos.</p> <p>El alumno entrega un manual técnico del sistema desarrollado, el cual contenga los módulos y sus características que lo aplica.</p>	Comprender la construcción de un sistema para el control de un proceso digital en tiempo real.	Ejecución de tareas. Lista de verificación y guía de observación.


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

PROGRAMACIÓN AVANZADA

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Expositiva y Discusión del proyecto realizado. Prácticas de laboratorio. Aprendizaje Basado en Proyectos.	Pizarrón, Cañón, Equipo de cómputo, Software de programación.

ESPACIO FORMATIVO


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Aula	Laboratorio / Taller	Empresa
X	X	


PROGRAMACIÓN AVANZADA

CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA

Capacidad	Criterios de Desempeño
Integrar, construir e innovar algoritmos de inteligencia computacional para el mejoramiento de procesos industriales capaces de cumplir tareas específicas, generando procesos con mayor calidad, eficiencia, precisión, versatilidad, seguridad para el mejoramiento de la competitividad de nuestras empresas.	<p>Elabora una propuesta del diseño que integre:</p> <ul style="list-style-type: none"> • Necesidades del cliente en el que se identifique: capacidades de producción, medidas de seguridad, intervalos de operación del sistema, flexibilidad de la producción, control de calidad • Descripción del proceso • Esquema general del proyecto, • Sistemas y elementos a integrar al proceso y sus especificaciones técnicas por áreas: Eléctricos, Electrónicos, Mecánicos, Elementos de control • Características de los requerimientos del área a automatizar. • Estimado de costos y tiempos de entrega.
Modelar y diseñar elementos propuestos apoyados en las tecnologías de información y simulación de los sistemas que intervienen en la automatización y control para definir sus características técnicas.	<p>Entregue el diagrama y el modelo del prototipo físico o virtual por implementar o probar, estableciendo las especificaciones técnicas de cada elemento y sistema que componen la propuesta, diagramas de flujo o programas mostrando los resultados que nos arrojan, en la que nos muestre el funcionamiento correcto.</p> <ul style="list-style-type: none"> - Descripción de entradas, salidas de datos. - Modelado del programa. - Configuración y/o programación.

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Capacidad	Criterios de Desempeño
Implementación de modelos físicos o virtuales considerando el diagrama, para validar y depurar la funcionalidad del diseño.	<p>Depura y optimiza el modelo mediante:</p> <ul style="list-style-type: none"> • La configuración y programación de los elementos que así lo requieran de acuerdo a las especificaciones del fabricante. • Desarrollo de pruebas de desempeño de los elementos y sistemas, y registro de los resultados obtenidos. • Ajustes necesarios para la optimización del desempeño de los elementos y sistemas.
Ajustar el modelo propuesto con base en normas aplicables, su eficiencia y costos para determinar su factibilidad.	<p>Determina los costos presupuestados y tiempos de realización.</p> <p>Documentar el diseño en forma de diagramas de flujo, para su reproducción y control de cambios, elaborando un reporte que contenga:</p> <ul style="list-style-type: none"> • Propuesta de diseño • Planos, diagramas o programas realizados. • Especificaciones de ensamble, configuración y/o programación de los elementos que lo requieran. • Características de suministro de energía (eléctrica, neumática, etc.), • Protocolos de comunicación. • Resultados de la simulación de desempeño de los elementos y sistemas. • Ajustes realizados al diseño de los elementos y sistemas. • Costos y tiempos de realización. • Resultado de la evaluación del diseño.
Controlar el desarrollo del proyecto de automatización y control por medio del liderazgo de comunicación efectiva, utilizando el sistema de control estadístico (Project, cuadro mando integral, diagramas de Gantt) para alcanzar los objetivos y metas del proyecto.	Elabora y justifica en un reporte que incluya: el avance programático de metas alcanzadas vs programadas; las acciones correctivas y preventivas.


ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Capacidad	Criterios de Desempeño
Supervisar la instalación, puesta en marcha y operación de sistemas, con base en las características especificadas, recursos destinados, procedimientos, condiciones de seguridad, y la planeación establecida, para asegurar el proceso final.	Realiza una lista de verificación de tiempos y características donde registre: *Tiempos de ejecución, *Recursos ejercidos, *Cumplimiento de características, *Normativas y seguridad, y *Funcionalidad *Procedimiento de arranque y paro. Realiza un informe de acciones preventivas y correctivas que aseguren el cumplimiento del proyecto.
Evaluar los resultados del desempeño del sistema automatizado con base en pruebas ejecutadas en condiciones normales y máximas de operaciones para realizar ajustes y validar el cumplimiento de los requisitos especificados.	Aplica procedimientos de evaluación considerando: análisis estadísticos de resultados, pruebas físicas, repetibilidad y análisis comparativos respecto del diseño del proceso, registrando los resultados de operación en función a las características solicitadas en condiciones normales y máxima de operación.


PROGRAMACIÓN AVANZADA

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Deitel, Harvey	2007	Como Programar en C/C++ y JAVA 4ª Edición	Distrito Federal	México	Prentice Hall ISBN: 9702605318
Joyanes aguilar, Luis	2014	Programación en C/C++ Java y UML	Distrito Federal	México	MCGRAW HILL EDUCATION ISBN: 6071512123
Sharp, John	2010	Visual C# 2010	Distrito Federal	México	Anaya multimedia, ISBN: 8441528241
Francisco Javier Ceballos	2014	Enciclopedia de Microsoft Visual C#: Interfaces Gráficas y Aplicaciones para internet con Windows Forms y ASP.NET 4ª Edición	Distrito Federal	México	Alfaomega ISBN: 9786077079194

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	

Autor	Año	Título del Documento	Ciudad	País	Editorial
W. Larsen, Ronald	(2010)	<i>Labview for Engineers</i>	Montana	EUA	Alfaomega
Sharp, John	2013	Microsoft Visual C# 2013 Step by Step	EUA	EUA	Pearson Education ISBN: 9780735681835

ELABORÓ:	Comité de Directores de la Carrera de Ingeniería en Mecatrónica	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2020	