

### Programa descriptivo por unidad de competencia

<b>Programa educativo</b>	<b>Licenciatura en Ingeniería en Desarrollo y Tecnologías de Software</b>	<b>Modalidad</b>		Presencial	
<b>Clave</b>	IS05	<b>H S M</b>		<b>Horas semestrales</b>	<b>Créditos</b>
<b>Unidad de competencia</b>	<b>Programación avanzada</b>	<b>Teoría</b>	<b>Práctica</b>	64	6
		2	2		
<b>Ubicación</b>	Tercer semestre.	<b>Unidades CONAIC</b>		42.67	
<b>Prerrequisito</b>	Estructura de datos y programación orientada a objetos.	<b>H S M de cómputo</b>		2	
<b>Perfil docente</b>	Contar con título profesional, grado de maestría y preferentemente con grado de doctorado en áreas afines a informática y computación. Demostrar experiencia en docencia en el nivel medio superior o superior mínima de dos años. Dominar los lenguajes de programación actuales, deseable con una certificación en lenguajes de programación.				
<b>Presentación</b>	Proporciona al estudiante las competencias necesarias para el uso de software de diseño y simulación, pero debe existir de manera explícita y separada, para la creación de software que apoye a las áreas de interfaces, automatización y control.				
<b>Propósito</b>	Aprender técnicas para el desarrollo de sistemas mecatrónicos automáticos, controlados por computadora, a través de sus interfaces electrónicas, incluyendo el desarrollo de interfaces gráficas para interacción hombre-máquina.				
<b>Competencias genéricas</b>					
Maneja Tecnologías de la información y comunicación para la gestión y construcción de conocimientos. Se desempeña en su práctica profesional y vida personal bajo principios éticos y morales.					
<b>Competencias disciplinares</b>					
Aplica habilidades de abstracción y expresión matemática para la solución de problemas. Formula modelos matemáticos para la solución de problemas mediante el desarrollo de aplicaciones de software para diversos entornos.					
<b>Competencias profesionales</b>					
Selecciona de técnicas o algoritmos en computación teórica para la solución de problemas.					

## Mapa de la unidad de competencia

Unidad de competencia	Subcompetencia	Resultado de aprendizaje
<b>Programación avanzada</b>	1. Realiza programación visual.	1.1. Diseño de interfaces gráficas de usuario. 1.2. Maneja eventos para la interacción hombre-máquina.
	2. Manejo de puertos.	2.1. Diseña software para el envío y recepción de datos por el puerto serial. 2.2. Interconectar dispositivos al puerto serial e interactuar con ellos mediante el software diseñado. 2.3. Diseña software para el envío y recepción de datos por el puerto USB. 2.4. Diseña software para controlar la entrada y salida de datos por el puerto paralelo.
	3. Aplica algoritmos de programación en tiempo real.	3.1. Diseña un árbol binario y realizar el recorrido de un árbol utilizando un lenguaje de programación. 3.2. Utilizar un lenguaje de programación de alto nivel para representar expresiones regulares. 3.3. Realiza prácticas de laboratorio para la programación de PLCs, como casos de aplicación de autómatas. 3.4. Simula a través de un lenguaje de alto nivel, la representación de una máquina de Turing. y de una Red de Petri.
	4. Realiza el desarrollo de aplicaciones.	4.1. Elabora por equipo, un proyecto mecatrónico que involucre el diseño del software de control y la construcción de las interfaces electrónicas para su interacción con sensores y actuadores.

## Cuadro descriptivo por subcompetencia

<b>Subcompetencia</b>	<b>Realiza programación visual.</b>			<b>Número</b>	<b>1</b>
<b>Propósito de la subcompetencia</b>	Dominar y aplicar las tecnologías actuales y emergentes de programación visual para la solución de problemas en el ámbito de la automatización y el control.			<b>Total de horas</b>	12
<b>Resultado de aprendizaje</b>	1.1. Diseño de interfaces gráficas de usuario.			<b>Horas asignadas</b>	6
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Preguntas del tema. 2. Análisis del tema.	1. Examen. 2. Mapas Conceptuales. 3. Cuadro Sinóptico.	10%	1. Paradigma de la programación visual. 2. Lenguajes de programación visual y orientada a eventos.		
<b>Resultado de aprendizaje</b>	1.2. Maneja eventos para la interacción hombre-máquina.			<b>Horas asignadas</b>	6
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Preguntas del Tema. 2. Análisis del Tema.	1. Examen. 2. Mapas Conceptuales. 3. Cuadro Sinóptico.	15%	1. Diseño de interfaces visuales. 2. Aplicaciones.		

## Cuadro descriptivo por subcompetencia

<b>Subcompetencia</b>	<b>Manejo de puertos.</b>			<b>Número</b>	<b>2</b>
<b>Propósito de la subcompetencia</b>	Identificar los diferentes puertos, su funcionamiento para el envío y recepción de datos, así como su programación.			<b>Total de horas</b>	16
<b>Resultado de aprendizaje</b>	2.1. Diseña software para el envío y recepción de datos por el puerto serial.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Examen teórico. 3. Desarrollo de programas.	5%	1. Teoría del muestreo y adquisición de datos.		
<b>Resultado de aprendizaje</b>	2.2. Interconectar dispositivos al puerto serial e interactuar con ellos mediante el software diseñado.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Examen teórico. 3. Desarrollo de programas.	5%	1. Conceptos básicos de programación de puertos mediante objetos.		
<b>Resultado de aprendizaje</b>	2.3. Diseña software para el envío y recepción de datos por el puerto USB.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Examen teórico. 3. Desarrollo de programas.	5%	1. Configuración de puertos paralelo. 2. Configuración de puertos serial.		
<b>Resultado de aprendizaje</b>	2.4. Diseña software para controlar la entrada y salida de datos por el puerto paralelo.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Examen teórico. 3. Desarrollo de programas.	10%	1. Configuración de puertos USB.		

## Cuadro descriptivo por subcompetencia

<b>Subcompetencia</b>	<b>Aplica algoritmos de programación en tiempo real.</b>			<b>Número</b>	<b>3</b>
<b>Propósito de la subcompetencia</b>	Aplicar algoritmos de programación en tiempo real para diseñar árboles binarios y representar expresiones regulares.			<b>Total de horas</b>	16
<b>Resultado de aprendizaje</b>	3.1. Diseña un árbol binario y realizar el recorrido de un árbol utilizando un lenguaje de programación.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Cuadro sinóptico. 3. Examen teórico. 4. Desarrollo de programas.	5%	1. Autómatas.		
<b>Resultado de aprendizaje</b>	3.2. Utilizar un lenguaje de programación de alto nivel para representar expresiones regulares.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Cuadro sinóptico. 3. Examen teórico. 4. Desarrollo de programas.	5%	1. Autómatas de estado finito determinístico. 2. Autómatas de estado finito no determinístico.		
<b>Resultado de aprendizaje</b>	3.3. Realiza prácticas de laboratorio para la programación de PLCs, como casos de aplicación de autómatas.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Cuadro sinóptico. 3. Examen teórico. 4. Desarrollo de programas.	10%	1. Máquinas de Turing		

<b>Resultado de aprendizaje</b>	3.4. Simula a través de un lenguaje de alto nivel, la representación de una máquina de Turing. y de una Red de Petri.			<b>Horas asignadas</b>	4
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>		
1. Análisis de los temas. 2. Preguntas de los temas. 3. Ejercicios de los temas.	1. Mapas conceptuales. 2. Cuadro sinóptico. 3. Examen teórico. 4. Desarrollo de programas.	10%	1. Redes de Petri.		

## Cuadro descriptivo por subcompetencia

<b>Subcompetencia</b>	<b>Realiza el desarrollo de aplicaciones.</b>	<b>Número</b>	<b>4</b>
<b>Propósito de la subcompetencia</b>	Utiliza la técnicas aprendidas en la unidad de competencia para implementarlas en su proyecto integrador.	<b>Total de horas</b>	20
<b>Resultado de aprendizaje</b>	4.1. Elabora por equipo, un proyecto mecatrónico que involucre el diseño del software de control y la construcción de las interfaces electrónicas para su interacción con sensores y actuadores.	<b>Horas asignadas</b>	20
<b>Actividades de evaluación</b>	<b>Evidencias a recopilar</b>	<b>%</b>	<b>Contenido</b>
1. Desarrollar proyecto con base en temas de la unidad de competencia.	1. Proyecto final.	20%	1. Proyecto integrador.

<b>Actitudes y valores</b>	Ordenado. Coherente. Proactivo. Asertivo.	
<b>Recursos, materiales y equipo didáctico</b>		
<b>Recursos didácticos</b>	<b>Equipo de apoyo didáctico</b>	
Antologías. Diapositivas. Videos.	Proyector de video. Software especializado.	
<b>Fuentes de información</b>		
<b>Bibliografía básica:</b> Sznajdleder, P. (2013). <i>Java a fondo: Estudio del lenguaje y desarrollo de aplicaciones</i> (2a. ed.). México: Ra-Ma. Álvarez, B. (2009). <i>Estructura de computadores, programación del proceso</i> . México: Alfaomega. Joyanes, L. (2014). <i>Programación en C/C++, Java y UML</i> (2a. ed.). México: Mc Graw Hill. Wu (2008). <i>Programación en java</i> . México: Mc Graw Hill.		
<b>Bibliografía complementaria:</b> Aho, Lam, Sethi, y Ullman (2008). <i>Compiladores principios técnicas y herramientas</i> (2a. ed.). México: Pearson Adison Wesley. Bell, D. (2010). <i>C# para estudiantes</i> . México: Pearson. Chapra, S. C. (2011). <i>Métodos numéricos para ingenieros</i> (6a. ed.). México: Mc Graw Hill.		
<b>Recursos digitales:</b> Ninguno.		