

Programa descriptivo por unidad de competencia

Programa educativo	Licenciatura en Ingeniería en Desarrollo y Tecnologías de Software	Modalidad		Presencial	
Clave	IS02	H S M		Horas semestrales	Créditos
Unidad de competencia	Programación estructurada	Teoría	Práctica	64	6
		2	2		
Ubicación	Primer semestre.	Unidades CONAIC		42.67	
Prerrequisito	Ninguno.	H S M de cómputo		2	
Perfil docente	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.				
Presentación	En esta Unidad de Competencia (UC) el estudiante desarrolla la capacidad para identificar, analizar y resolver problemas con técnicas de programación estructurada, coadyuvando a la automatización y programación de procesos que le permitan fortalecer su pensamiento lógico y analítico. Se ubica en el área de conocimiento de programación e ingeniería de software, y se relaciona con la UC de Estructura de datos. Atiende al perfil de egreso en el sentido que domine los conocimientos teóricos y prácticos para la construcción conceptual de soluciones de software, como los fundamentos de las habilidades de programación y del desarrollo de la unidad de competencia.				
Propósito	Analiza, diseña y desarrolla soluciones de problemas reales utilizando técnicas de programación estructurada y herramientas computacionales para implementarlos en un lenguaje de programación.				
Competencias genéricas					
Maneja Tecnologías de la información y comunicación para la gestión y construcción de conocimientos.					
Competencias disciplinares					
Posee los conocimientos teóricos y prácticos para la construcción conceptual de soluciones de software.					
Competencias profesionales					
Identifica los elementos necesarios para la implementación de software.					

Mapa de la unidad de competencia

Unidad de competencia	Subcompetencia	Resultado de aprendizaje
<p>Programación estructurada</p>	<p>1. Identifica las estructuras de control de flujo y los elementos básicos de la programación estructurada.</p>	<p>1.1. Identifica el uso de las estructuras de control, declaración de variables y constantes. 1.2. Utiliza estructuras de control, variables y constantes en programas de cómputo.</p>
	<p>2. Distingue estructuras estáticas y dinámicas.</p>	<p>2.1. Crea funciones en los programas que desarrolla e identifica los tipos de arreglos que utiliza en un problema específico.</p>
	<p>3. Usa apuntadores y cadenas.</p>	<p>3.1. Utiliza apuntadores y funciones de biblioteca para el manejo de cadenas en programas de cómputo.</p>
	<p>4. Aplica estructuras de datos y modifica archivos.</p>	<p>4.1. Crea estructura de datos para la manipulación de información. 4.2. Utiliza funciones de biblioteca de manipulación de archivos para la recuperación y escritura de datos.</p>

Cuadro descriptivo por subcompetencia

Subcompetencia	Identifica las estructuras de control de flujo y los elementos básicos de la programación estructurada.	Número	1
Propósito de la subcompetencia	Utiliza las estructuras de control mediante la creación de diagramas que representan las estructuras simples y complejas, y aprende a declarar variables y constantes.	Total de horas	14
Resultado de aprendizaje	1.1. Identifica el uso de las estructuras de control, declaraciones de variables y constantes.	Horas asignadas	8
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> 1. Genera diagramas de flujo de estructuras simples. 2. Crea programas en pseudocódigo. 	<ol style="list-style-type: none"> 1. Documento con un diagrama de flujo de estructura simple realizado con un software especializado. 2. Documento digital y/o impreso con el pseudocódigo del programa a realizar. 	10%	<ol style="list-style-type: none"> 1. Diagramas de Flujo. 2. Pseudocódigo. 3. Control de flujo. 4. Estructuras secuenciales. 5. Estructuras condicionales. 6. Estructuras repetitivas.
Resultado de aprendizaje	1.2. Utiliza estructuras de control, variables y constantes en programas de cómputo.	Horas asignadas	8
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> 1. Investiga el paradigma de programación estructurada. 2. Identifica el alcance y declaración de variables y constantes. 3. Crea un programa con base en un pseudocódigo generado con flujos de datos, variables y constantes. 4. Evaluación aprendizajes. 	<ol style="list-style-type: none"> 1. Documento con la investigación del paradigma de la programación estructurada y sus fuentes de información. 2. Documento que explique la declaración y ámbito de variables y constantes. 3. Documento digital con el código fuente del programa. 4. Examen escrito. 	15%	<ol style="list-style-type: none"> 1. Conceptos básicos de programación. 2. Declaración y alcance de variables. 3. Contantes. 4. Palabras reservadas. 5. Identificadores. 6. Operadores. 7. Estructura de un programa.

Cuadro descriptivo por subcompetencia

Subcompetencia	Distingue estructuras estáticas y dinámicas.	Número	2
Propósito de la subcompetencia	Crea, comprende, y aplica los diferentes tipos de datos, declaración y manipulación de funciones, los diferentes tipos de arreglos.	Total de horas	16
Resultado de aprendizaje	2.1. Crea, comprende, y aplica los diferentes tipos de datos, declaración y manipulación de funciones, los diferentes tipos de arreglos.	Horas asignadas	16
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Investiga los tipos de datos y el tamaño en bytes de los tipos de datos usados en ANSI C. Identifica los tipos de arreglos soportados en ANSI C. Experimenta con diferentes tipos de arreglos en ANSI C. Investiga la declaración e importancia de las funciones. Programa una aplicación con funciones y arreglos. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento con la investigación realizada de los tipos de datos soportados en ANSI C. Documento digital del código fuente del programa para el manejo de arreglos en ANSI C. Documento con la investigación realizada de la declaración y uso de funciones en ANSI C. Documento digital del código fuente del programa para el manejo de funciones y arreglos ANSI C. Examen escrito. 	25%	<ol style="list-style-type: none"> Tipos de datos. Arreglos unidimensionales. Arreglos bidimensionales. Arreglos multidimensionales. Funciones. Declaración de funciones. Funciones con paso de parámetros. Funciones recursivas.

Cuadro descriptivo por subcompetencia

Subcompetencia	Usa apuntadores y cadenas.	Número	3
Propósito de la subcompetencia	Crea, comprende, y aplica los diferentes tipos de datos, declaración y manipulación de funciones, los diferentes tipos de arreglos.	Total de horas	16
Resultado de aprendizaje	3.1. Utiliza apuntadores y funciones de biblioteca para el manejo de cadenas en programas de cómputo.	Horas asignadas	16
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> 1. Investiga el uso de apuntadores, paso por referencia, asignación dinámica de memoria. 2. Demuestra el empleo de apuntadores en un programa de computo. 3. Investiga funciones de ANSI C para el manejo de cadenas. 4. Demuestra mediante un programa el empleo de apuntadores para la manipulación de cadenas de texto. 5. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> 1. Documento con la investigación realizada de apuntadores. 2. Documento digital del código fuente del programa para el manejo de apuntadores en ANSI C. 3. Documento con la investigación realizada de las funciones de biblioteca para la manipulación de cadenas. 4. Documento digital del código fuente del programa para el manejo de apuntadores y cadenas en ANSI C. 	25%	<ol style="list-style-type: none"> 1. Apuntadores. 2. Paso por referencia. 3. Apuntadores a funciones. 4. Funciones de biblioteca para el manejo de cadenas. 5. Arreglos de cadenas. 6. Declaración de bibliotecas definidas por el usuario.

Cuadro descriptivo por subcompetencia

Subcompetencia	Aplica estructuras de datos y modifica archivos.	Número	4
Propósito de la subcompetencia	Conoce, crea y usa estructuras de datos para la administración de información y el manejo de bibliotecas para la manipulación de archivos.	Total de horas	16
Resultado de aprendizaje	4.1. Crea estructura de datos para la manipulación de información.	Horas asignadas	8
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Investiga la declaración y el uso de estructuras de datos. Crea un programa usando estructuras de datos. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento digital con la investigación de Estructuras en ANSI C. Código fuente utilizando estructura de datos en ANSI C. Examen escrito. 	10%	<ol style="list-style-type: none"> Introducción a las estructuras de datos. Declaración de estructuras de datos. Estructuras anidadas.
Resultado de aprendizaje	4.2. Utiliza funciones de biblioteca de manipulación de archivos para la recuperación y escritura de datos.	Horas asignadas	8
Actividades de evaluación	Evidencias a recopilar	%	Contenido
<ol style="list-style-type: none"> Investiga funciones en ANSI C para el manejo de archivos. Programa aplicaciones en ANSI C con estructura de datos para la manipulación de contenido en un archivo de texto. Evaluación del resultado de aprendizaje. 	<ol style="list-style-type: none"> Documento digital con la investigación de las funciones de biblioteca para el manejo de archivos en ANSI C. Código fuente en ANSI C de un programa para la creación de archivos, escritura de texto empleando estructura de datos. Examen escrito. 	15%	<ol style="list-style-type: none"> Funciones de biblioteca para la manipulación de archivos. Escritura hacia un archivo. Recuperación de datos. Bloques de archivos.

Actitudes y valores	Responsabilidad. Liderazgo. Cultura de trabajo.	
Recursos, materiales y equipo didáctico		
Recursos didácticos	Equipo de apoyo didáctico	
Diapositivas. Ejercicios. Guías de práctica. Prácticas de laboratorio.	Equipo de computo. Laboratorio de cómputo. Software especializado.	
Fuentes de información		
Bibliografía básica: Joyanes A. L. (2014). <i>Programación en C/C++ Java y UML</i> (2a. ed.). México: Mc Graw Hill. Márquez, O. (2011). <i>Introducción a la programación estructurada en C</i> . México: Pearson. Guerra C. H. (2003). <i>C, el comienzo...</i> México: Tecnología didáctica		
Bibliografía complementaria: Gottfried. B. (2014). <i>Programación en C</i> (2a. ed.). España: McGraw-Hill Interamericana. Ceballos S., F. (2014). <i>Microsoft C#. Curso de programación</i> (2a. ed.). México: Alfaomega Ra-Ma. Wirth, N. (1987). <i>Algoritmos y estructuras de datos</i> (2a. ed.). México: Prentice Hall. Aho, A.V. (1988). <i>Estructuras de datos y algoritmos</i> . USA: Addison Wesley Iberoamérica.		
Recursos digitales: Ninguno.		