

Programa de asignatura por competencias de educación superior

Sección I. Identificación del Curso

Tabla 1. Identificación de la Planificación del Curso.

Actualización:	Marzo 22, 2022				
Carrera:	Ingeniería en Tecnologías de Software	Asignatura:	Programación Estructurada		
Academia:	Informática y Computación Virtual /	Clave:	19SICTS0303		
Módulo formativo:	Informática y Computación	Seriación:	19SICTS0406 - Programación Orientada a Objetos		
Tipo de curso:	Modalidad mixta	Prerrequisito:	19SICTS0202 - Introducción a la Programación		
Semestre:	Tercero	Créditos:	5.63	Horas semestre:	90 horas
Teoría:	0 horas	Práctica:	0 horas	Trabajo indpt.:	5 horas
				Total x semana:	5 horas

Sección II. Objetivos educacionales

Tabla 2. Objetivos educacionales

Objetivos educacionales		Criterios de desempeño	Indicadores
1	Solucionará problemas con sólidas bases científicas y fundamentos tecnológicos que le permitirán comprender, analizar, diseñar, organizar, producir, operar y dar soluciones prácticas a problemas relacionados con las áreas de Organización de Sistemas Computacionales e Ingeniería en Software para el sector productivo y social, promoviendo los principios de ética, responsabilidad y trabajo colaborativo.	El egresado implementará las diferentes etapas del ciclo de vida del software contemplando la protección de datos y prevención de desastres, salvaguardando con ética la seguridad de la información.	50 % Egresados trabajarán en cualquier proceso del desarrollo de software o áreas afines a los sistemas computacionales, promoviendo los principios de ética, responsabilidad y trabajo colaborativo.
2	Aportará soluciones innovadoras y sustentables en el área de la electrónica en el que establezca el análisis, diseño, implementación, selección de componentes de hardware de uso específico, el software asociado y su conectividad a través de redes de comunicación para el sector productivo y social.	El egresado implementará las diferentes técnicas de análisis y diseño de circuitos electrónicos que den una solución innovadora sustentable a problemas con el hardware.	20% Egresados trabajarán en cualquier proceso de creación y aplicación de hardware o áreas afines en el sector productivo y social.
3	Implementará soluciones innovadoras y sustentables con tecnologías de información que sean acordes a las necesidades, a las tecnologías disponibles y emergentes, para lograr un aprovechamiento óptimo de los recursos humanos y financieros en el sector productivo y social.	El egresado implementará las diferentes tecnologías emergentes en equipos multidisciplinarios que den una solución innovadora y sustentable a las necesidades que se presenten en el ámbito productivo y social.	20 % Egresados trabajarán en la aplicación de Tecnologías de la información o áreas afines en el sector productivo o social.



Atributos de egreso de plan de estudios		Criterios de desempeño	Componentes
AE3	Aplicar una experimentación adecuada con apoyo de metodologías y juicio ingenieril que permitan interpretar datos para obtener conclusiones que den solución a problemáticas en un contexto determinado.	<ul style="list-style-type: none"> - Comprenderá las sintaxis básicas y su aplicación en la programación estructurada en C++ en el diseño de programas para una solución efectiva en la resolución de problemas específicos. - Comprenderá las estructuras de control y su aplicación en la programación estructurada en C++ en el diseño de programas con mayor complejidad para una solución efectiva en la resolución de problemas específicos. - Comprenderá la sintaxis de los arreglos y su aplicación en la programación estructurada en C++ en el diseño de programas específicos para la solución de problemas con requerimientos de almacenamiento de datos. 	<ul style="list-style-type: none"> 1.1 Identificadores. 1.2 Tipos de datos. 1.3 Variables y constantes. 1.4 Operadores y jerarquía. 1.5 Expresiones. 1.6 Palabras reservadas. 1.7 Comentarios. 1.8 Funciones de entrada (scanf y gets). 1.9 Funciones de salida (printf). 2.1 Contadores y acumuladores. 2.2 Estructuras de control. <ul style="list-style-type: none"> 2.2.1 Secuenciales. 2.2.2 Selectivas. <ul style="list-style-type: none"> 2.2.2.1 Simples. 2.2.2.2 Dobles. 2.2.2.3 Múltiples. 2.2.3 Repetitivas. <ul style="list-style-type: none"> 2.2.3.1 Desde. 2.2.3.2 Mientras. 2.2.3.3 Hacer mientras. 3.1. Arreglos. <ul style="list-style-type: none"> 3.1.1. Arreglos unidimensionales. 3.1.2. Arreglos bidimensionales. 3.2. Funciones. <ul style="list-style-type: none"> 3.2.1. Prototipo. 3.2.2. Sin parámetro.



Continuación: Tabla 2. Objetivos educacionales (continuación)

No.	Atributos de egreso de plan de estudios	Criterios de desempeño	Componentes
			3.2.3. Con parámetro. 3.2.4. Que regresa valor. 3.2.5. Que no regresa valor.

Sección III. Atributos de la asignatura

Tabla 3. Atributos de la asignatura

Problema a resolver		
Iniciar a los estudiantes en la programación estructurada, partiendo del diseño, análisis y resolución de problemas mediante algoritmos y diagramas de flujo para la representación de la solución en programas diseñados en C++.		
Atributos (competencia específica) de la asignatura		
Desarrollar programas que permitan almacenar, disponer y procesar información, utilizando el paradigma de programación estructurada que permitirán la construcción de programas con el uso de las diferentes estructuras en C++.		
Aportación a la competencia específica		Aportación a las competencias transversales
Saber	Saber hacer	Saber Ser
<ul style="list-style-type: none"> - Identificar las diferentes sintaxis del lenguaje C++. - Analizar y sintetizar soluciones a problemas planteados 	<ul style="list-style-type: none"> - Resolver problemas con las diferentes estructuras del lenguaje. - Aplicar los conocimientos en la práctica en el desarrollo de programa en ++. - Identificar, plantear y resolver problemas específicos acordes a las estructuras de control y arreglos de C++. 	<ul style="list-style-type: none"> - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa. - Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo. - Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación. - Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning.
Producto integrador de la asignatura, considerando los avances por unidad		
Proyecto integrador, a partir de la creación de un programa, dando solución a una necesidad real en una organización, incorporando las competencias desarrolladas en cada una de las unidades de aprendizaje.		

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.1. Desglose específico de la unidad "Introducción al lenguaje C++"

Número y nombre de la unidad: 1. Introducción al lenguaje C++							
Tiempo y porcentaje para esta unidad:		Teoría:	12 horas	Práctica:	12 horas	Porcentaje del programa:	26.67%
Aprendizajes esperados: Conocer las sintaxis básicas para el diseño de un programa en C++							
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 Identificadores. 1.2 Tipos de datos. 1.3 Variables y constantes. 1.4 Operadores y jerarquía. 1.5 Expresiones. 1.6 Palabras reservadas. 1.7 Comentarios. 1.8 Funciones de entrada (scanf y gets). 1.9 Funciones de salida (printf).	Saber: - Definir qué son los identificadores y los tipos de datos. - Identificar los operadores y jerarquía, expresiones, palabras reservadas, comentarios, así como funciones de entrada y salida. Saber hacer: - Realizar ejercicios programados para las actividades de cada estructura. - Realizar tareas de ejercicios de programas.	- Exposición por parte del profesor de material teórico. - Complementar información con material audiovisual. - Revisar la Introducción a la programación estructurada mediante ejercicios prácticos de cada estructura	Evaluación Diagnóstica: -Rescatar conocimientos previos. Instrumento: cuestionario. Evaluación Formativa: -Realización de ejercicios prácticos. Instrumento: Rúbrica. Evaluación sumativa: -Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar los tipos de datos, variables, expresiones, funciones de entrada y salida, operadores que represente la solución en un programa acorde a la estructura básica en C++.			



Continuación: Tabla 4.1. 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
	<p>Ser:</p> <ul style="list-style-type: none"> - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa. - Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo. - Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación. - Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning. 			
Bibliografía				
<p>-Sznajdleder, P. (2017). Programa Estructurada a fondo. Implementación de algoritmos en C. Argentina: Alfaomega.</p> <p>-Domínguez, E.D. (2013). Programación Estructurada Raptor y Lenguaje c. México: Alfaomega.</p> <p>-Juganaru, M. (2014). Introducción a la programación. México: Patria.</p> <p>-Ayala, J. (2018). Lenguaje de Programación Estructurada. México: Centro Universitario UAEM.</p> <p>-Olvera, E.N.; Márquez, G. y Osorio,S.(2011). Introducción a la Programación Estructurada en C. México: Pearson.</p>				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.2. Desglose específico de la unidad "Estructuras de Control."

Número y nombre de la unidad: 2. Estructuras de Control.							
Tiempo y porcentaje para esta unidad:		Teoría:	12 horas	Práctica:	12 horas	Porcentaje del programa:	26.67%
Aprendizajes esperados:		Reconocer el funcionamiento y la sintaxis de contadores, acumuladores y estructuras de control secuenciales, selectivas y repetitivas para el diseño de programas más complejos en C++.					
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 Contadores y acumuladores. 2.2 Estructuras de control. 2.2.1 Secuenciales. 2.2.2 Selectivas. 2.2.2.1 Simples. 2.2.2.2 Dobles. 2.2.2.3 Múltiples. 2.2.3 Repetitivas. 2.2.3.1 Desde. 2.2.3.2 Mientras. 2.2.3.3 Hacer mientras.	Saber: - Analizar los conceptos de Contadores, acumuladores y estructuras de control. Saber hacer: - Realizar ejercicios programados para las actividades de cada estructura. - Realizar tareas de ejercicios de programas. Ser: - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y respetuosa.	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos. - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para programar.	Evaluación Diagnóstica: - Rescatar conocimientos previos y enlazarlos con la información de la unidad. Instrumento: cuestionario. Evaluación Formativa: - Realización de ejercicios prácticos. Instrumento: Rúbrica. Evaluación sumativa: -Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar las diferentes estructuras de control que represente la solución en un programa con mayor complejidad en C++.			



Continuación: Tabla 4.2. Desglose específico de la unidad "Estructuras de Control."

Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad
	<ul style="list-style-type: none"> - Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo. - Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación. - Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning. 			
Bibliografía				
<p>-Sznajdleder, P. (2017). Programa Estructurada a fondo. Implementación de algoritmos en C. Argentina: Alfaomega.</p> <p>-Domínguez, E.D. (2013). Programación Estructurada Raptor y Lenguaje c. México: Alfaomega.</p> <p>-Juganaru, M. (2014). Introducción a la programación. México: Patria.</p> <p>-Ayala, J. (2018). Lenguaje de Programación Estructurada. México: Centro Universitario UAEM.</p> <p>-Olvera, E.N.; Márquez, G. y Osorio,S.(2011). Introducción a la Programación Estructurada en C. México: Pearson.</p>				

Sección IV. Desglose específico por cada unidad formativa

Tabla 4.3. Desglose específico de la unidad "Arreglos y funciones."

Número y nombre de la unidad: 3. Arreglos y funciones.							
Tiempo y porcentaje para esta unidad:		Teoría:	12 horas	Práctica:	12 horas	Porcentaje del programa:	26.67%
Aprendizajes esperados: Reconocer el funcionamiento y sintaxis de los arreglos y funciones en la programación estructurada.							
Temas y subtemas (secuencia)	Criterios de desempeño	Estrategias didácticas	Estrategias de evaluación	Producto Integrador de la unidad (Evidencia de aprendizaje de la unidad)			
3.1. Arreglos. 3.1.1. Arreglos unidimensionales. 3.1.2. Arreglos bidimensionales. 3.2. Funciones. 3.2.1. Prototipo. 3.2.2. Sin parámetro. 3.2.3. Con parámetro. 3.2.4. Que regresa valor. 3.2.5. Que no regresa valor.	Saber: - Analizar los conceptos de Arreglos y funciones. Saber hacer: - Realizar ejercicios programados para las actividades de cada arreglo. - Realizar tareas de ejercicios de funciones. Ser: - Aporta puntos de vista con apertura a aprender de los otros y considera los de otras personas de manera reflexiva y	-Exposición por parte del docente con apoyo de las presentaciones con los temas descritos. - Complementar información con material audiovisual. -Resolver el problemario de ejercicios para programar.	Evaluación diagnóstica: - Rescatar conocimientos previos y enlazarlos con la información de la unidad. Instrumento: cuestionario. Evaluación Formativa: - Realización de ejercicios prácticos. Instrumento: Rúbrica. Evaluación sumativa: -Elaborar el planteamiento de un problema laboral o cotidiano. Instrumento: Rúbrica o lista de cotejo.	Planteamiento de un problema laboral o cotidiano en el que se pueda aplicar las funciones y arreglos que represente la solución en un programa con mayor complejidad en C++.			



Continuación: Tabla 4.3. 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
	<p>respetuosa.</p> <ul style="list-style-type: none"> - Participa activamente en la construcción de su aprendizaje y en la resolución de problemas, colaborando de manera productiva en espacios y equipos de trabajo. - Entrega en tiempo y forma sus actividades de aprendizaje siguiendo las indicaciones y considerando los criterios de evaluación. - Utiliza la tecnología para apoyar su aprendizaje y para el desarrollo de habilidades metacognitivas, el aprendizaje autónomo y el longlife learning. 			
Bibliografía				
<p>-Sznajdleder, P. (2017). Programa Estructurada a fondo. Implementación de algoritmos en C. Argentina: Alfaomega.</p> <p>-Domínguez, E.D. (2013). Programación Estructurada Raptor y Lenguaje c. México: Alfaomega.</p> <p>-Juganaru, M. (2014). Introducción a la programación. México: Patria.</p> <p>-Ayala, J. (2018). Lenguaje de Programación Estructurada. México: Centro Universitario UAEM.</p> <p>-Olvera, E.N.; Márquez, G. y Osorio,S.(2011). Introducción a la Programación Estructurada en C. México: Pearson.</p>				



V. Perfil docente

Tabla 5. Descripción del perfil docente

Perfil deseable docente para impartir la asignatura
<p>Carrera(s):</p> <ul style="list-style-type: none">- Ingeniería en Tecnologías de software.- Ingeniería en Informática, Computación.- Ingeniería en Tecnología de la Información y Comunicación.- Licenciatura en informática.- Ingeniería en Tecnologías de la Información y Comunicación.- Ingeniería en Tecnologías de la Información y Comunicaciones.- Licenciatura en Comunicación e Información.- Ingeniería en sistemas de computación.- Licenciatura en Ingeniería en sistemas computacionales.- Licenciatura en ingeniería en sistemas de información.- Ingeniería en sistemas estratégicos de información, especialización en redes.- Licenciatura en sistemas computacionales. o carrera afín. <p>Experiencia profesional:</p> <ul style="list-style-type: none">- Experiencia profesional comprobable correspondiente al campo disciplinar o carrera.- Experiencia mínima de dos años- Licenciatura en Informática o Sistemas Computacionales o Maestría relacionada con el área de conocimiento.