



INGENIERIA EN COMPUTACIÓN

1. DATOS DE IDENTIFICACIÓN DE LA UNIDAD DE APRENDIZAJE

Nombre de la Academia: Lenguajes Informáticos		
Nombre de la Unidad de Aprendizaje: Fundamentos de Programación	Tipo: <input checked="" type="checkbox"/> Curso <input type="checkbox"/> Taller <input type="checkbox"/> Curso – Taller	Nivel: Licenciatura
Área de formación: <input checked="" type="checkbox"/> Básica Común <input type="checkbox"/> Básica Particular <input type="checkbox"/> Especializante Obligatoria <input type="checkbox"/> Especializante Selectiva <input type="checkbox"/> Optativa Abierta	Modalidad: <input type="checkbox"/> Presencial <input type="checkbox"/> Mixta <input checked="" type="checkbox"/> Distancia (en línea)	Claves de los Prerrequisitos:
Horas: 51_ Teoría 17_ Práctica 68_ Total	Créditos: 8	CNR: 43779 Clave: 15882
Elaboró: Hugo Adrián Delgado Rodríguez		Fecha de elaboración: Julio 2018
Actualizó: Hugo Adrián Delgado Rodríguez		Fecha de actualización: Diciembre 2023

2. RELACIÓN CON EL PERFIL EGRESO

Esta unidad de aprendizaje se relaciona con el perfil de egreso con respecto a lo siguiente “Tendrá dominio de los principios teóricos y de los aspectos prácticos y metodológicos que sustentan el diseño y desarrollo de sistemas computacionales complejos”.

3. RELACIÓN CON EL PLAN DE ESTUDIOS

Esta unidad de aprendizaje se relaciona con el plan de estudios y el módulo de: “Algoritmia”, “Matemáticas discretas”.

4. PROPÓSITO

El propósito de esta unidad de aprendizaje es adquirir conocimientos y habilidades sobre el lenguaje de programación C, los fundamentos de la programación, sus elementos básicos, operadores y expresiones, estructuras de control y selección, funciones, recursividad, arrays, aplicables para la manipulación de la información y necesarias para la creación y reconocimiento de lenguajes de programación.

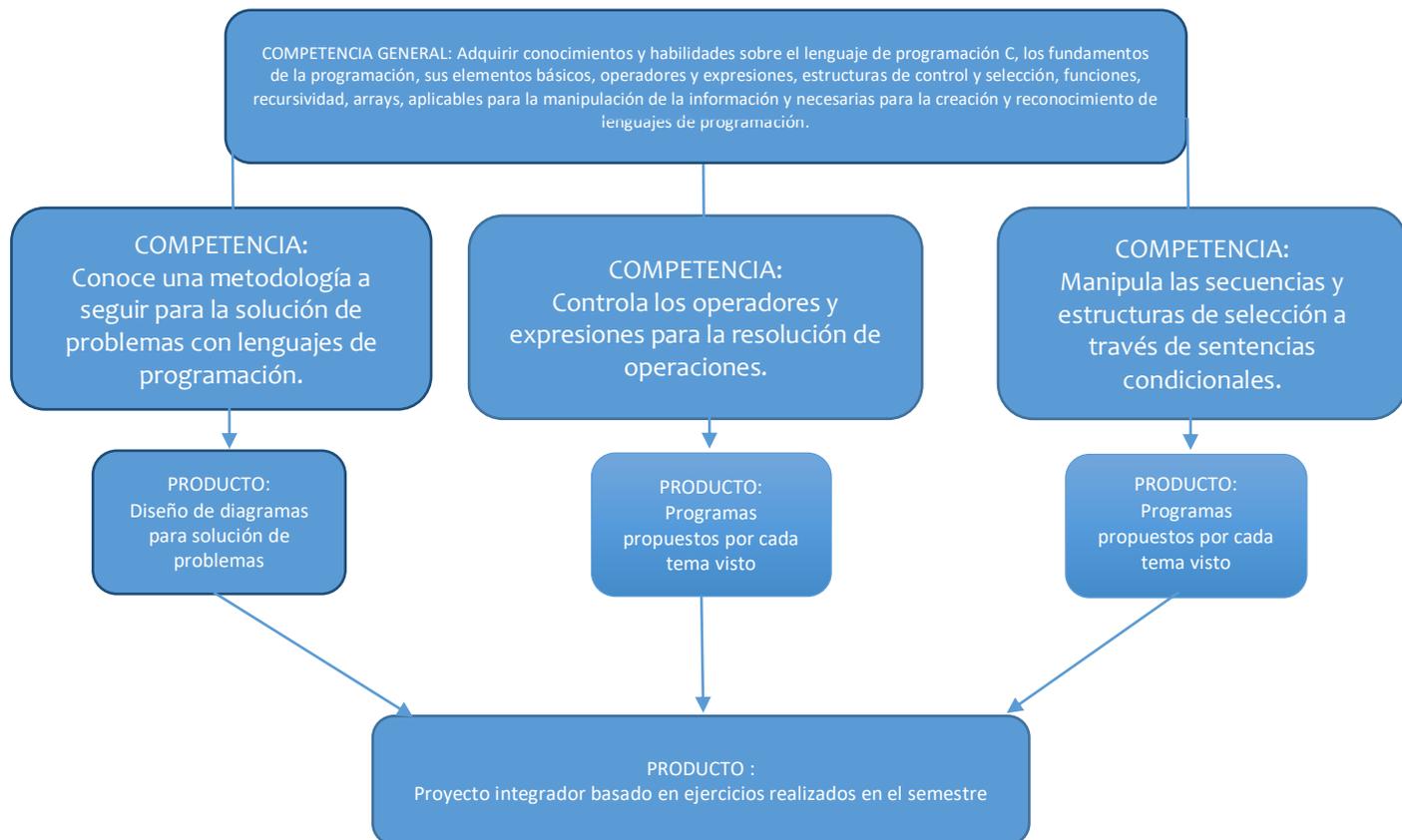


5. **COMPETENCIAS** a las que contribuye la unidad de aprendizaje.

COMPETENCIAS GENÉRICAS
<p>(x) Capacidad para la comunicación oral y escrita</p> <p>(x) Capacidad para la resolución de problemas</p> <p>() Capacidad para comunicarse en un segundo idioma</p> <p>(x) Capacidad de trabajo colaborativo</p> <p>(x) Capacidad para trabajar con responsabilidad social y ética profesional</p> <p>(x) Capacidad de autogestión</p> <p>() Capacidad de crear, innovar y emprender</p> <p>(x) Capacidad por la investigación y desarrollo tecnológico</p>
COMPETENCIAS ESPECÍFICAS
<p>(x) Aprende a seguir una metodología para la solución de problemas con computadoras y con el lenguaje de programación C.</p> <p>(x) Conoce los operadores y expresiones para la resolución de operaciones.</p> <p>(x) Controla las secuencias y estructuras de selección a través de sentencias condicionales</p> <p>(x) Manipula estructuras de control iterativas para controlar la secuencia de veces que una sentencia o lista se ejecutan</p>
COMPETENCIAS ESPECIALIZANTES
<p>ARQUITECTURA Y PROGRAMACIÓN DE SISTEMAS</p> <p>(x) Interpreta los datos para lograr la abstracción y síntesis de información. Conoce la estructura operacional y funcional de un sistema de computadoras.</p> <p>(x) Maneja volúmenes de datos organizados en estructuras para minimizar los costos de acceso a la información.</p> <p>() Maneja el almacenamiento secundario y realiza una clasifica datos que le permite generar consultas. Abstracción y síntesis de información.</p> <p>() Comprende el funcionamiento interno del procesador, y utilizar las directivas a bajo nivel.</p> <p>() Conoce las técnicas de organización, utilización y optimización de los sistemas y traductores.</p> <p>SISTEMAS INTELIGENTES</p> <p>() Emplea el razonamiento lógico-matemático para la resolución de problemas.</p> <p>(x) Emplea sus conocimientos matemáticos en el cálculo del tiempo de ejecución de un algoritmo y el análisis del orden de complejidad.</p> <p>() Aplica modelos matemáticos y de control para garantizar un comportamiento inteligente.</p> <p>() Resuelve problemas utilizando algoritmos de aprendizaje automático.</p> <p>SISTEMAS DISTRIBUIDOS</p> <p>() Identifica los protocolos de comunicación de redes de computadoras y verificar capacidad de respuesta de un sistema.</p> <p>() Relaciona los sistemas informáticos con su fiabilidad, seguridad y calidad.</p> <p>() Interpreta las funciones básicas de un sistema operativo distribuido en una red de computadoras.</p> <p>() Desarrollo de sistemas Web en un entorno distribuido.</p>



REPRESENTACIÓN GRÁFICA:



6. ESTRUCTURACIÓN DE LA UNIDAD DE APRENDIZAJE por temas (unidades temáticas), mencionando las competencias.

COMPETENCIA GENERAL DE LA UNIDAD DE APRENDIZAJE: Al final del curso, el alumno adquiera conocimientos y habilidades sobre el lenguaje de programación C, los fundamentos de la programación, sus elementos básicos, operadores y expresiones complejas para la solución de problemas en posteriores lenguajes de programación.

Competencia específica: Conoce los fundamentos de la programación, introducción a las computadoras y las fases para el análisis de solución de problemas en lenguaje C.	
Conocimientos (contenidos)	<ol style="list-style-type: none"> 1.1. Introducción a las computadoras y lenguajes de Programación 1.2. Fases en la resolución de problemas <ol style="list-style-type: none"> 1.2.1. Análisis del problema 1.2.2. Diseño del algoritmo



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE LA COSTA

DIVISIÓN DE INGENIERÍAS

DEPARTAMENTO DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

	<ul style="list-style-type: none"> 1.2.3. Codificación de un programa 1.2.4. Compilación y ejecución de un programa 1.2.5. Verificación y depuración 1.2.6. Documentación y mantenimiento
Habilidades	Identifica las fases en la resolución de problemas, codificación, ejecución y depuración.
Actitudes	<ul style="list-style-type: none"> Capacidad de autogestión Capacidad de trabajo colaborativo Capacidad por la investigación y desarrollo tecnológico

Competencia específica: Identifica El lenguaje C y sus elementos básicos y avanzados	
Conocimientos (contenidos)	<ul style="list-style-type: none"> 2. Estructura general de un programa en C <ul style="list-style-type: none"> 2.1.1. Directivas del preprocesador 2.1.2. Declaraciones globales 2.1.3. Función main() 2.1.4. Funciones definidas por el usuario 2.2. Los elementos de un programa C 2.3. Tipos de datos en C <ul style="list-style-type: none"> 2.3.1. Enteros 2.3.2. Tipo de coma flotante 2.3.3. Caracteres 2.4. El tipo de dato lógico 2.5. Constantes 2.6. Variables 2.7. Entradas y salidas 3. Operadores y expresiones <ul style="list-style-type: none"> 3.1. El operador de asignación 3.2. Operadores aritméticos



	<ul style="list-style-type: none">3.3. Operadores de incremento y decremento3.4. Operadores relacionales3.5. Operadores lógicos3.6. Operadores de manipulación de bits<ul style="list-style-type: none">3.6.1. Operadores de asignación adicionales3.6.2. Operadores de desplazamiento de bits3.7. Operadores condicionales3.8. Operador sizeof3.9. Conversiones de tipo4. Estructuras de selección<ul style="list-style-type: none">4.1. Estructuras de control4.2. La sentencia if con una alternativa4.3. Operadores de comparación4.4. La sentencia if con dos alternativas: if-else4.5. Sentencia de control switch5. Estructuras de control<ul style="list-style-type: none">5.1. La sentencia while5.2. Repetición: el bucle for5.3. Repetición: el bucle do while5.4. Comparación de bucles while, for, y do while6. Funciones<ul style="list-style-type: none">6.1. Concepto de función6.2. Estructura de una función6.3. Prototipos de las funciones6.4. Parámetros de una función6.5. Clases de almacenamiento6.6. Concepto de uso de funciones de biblioteca7. Arrays<ul style="list-style-type: none">7.1. Arrays7.2. Inicialización de un array7.3. Arrays de caracteres y cadena de texto7.4. Arrays multidimensionales7.5. Utilización de arrays como parámetros8. Algoritmos de ordenación y búsqueda
--	--



	<ul style="list-style-type: none">8.1. Ordenación8.2. Ordenación por burbuja8.3. Ordenación por selección8.4. Ordenación por inserción8.5. Ordenación Shell8.6. Ordenación rápida9. Punteros<ul style="list-style-type: none">9.1. Concepto de puntero9.2. Punteros (NULL y VOID)9.3. Punteros y arrays9.4. Aritmética de punteros9.5. Punteros como argumentos de funciones10. Asignación dinámica de memoria<ul style="list-style-type: none">10.1. Gestión dinámica de la Memoria10.2. Función malloc()10.3. Liberación de Memoria10.4. Funciones calloc() y realloc()11. Cadenas<ul style="list-style-type: none">11.1. Concepto de cadena11.2. Inicialización de variables de cadena11.3. Lectura de cadenas11.4. Las funciones de STRING.H11.5. Conversiones de cadenas a números12. Entrada y salida de archivos<ul style="list-style-type: none">12.1. Flujos12.2. Aperturas de un archivo12.3. Funciones de lectura y escritura13. Organización de datos en un archivo<ul style="list-style-type: none">13.1. Registros13.2. Organización de archivos13.3. Archivos con direccionamiento hash13.4. Archivos secuenciales indexados13.5. Ordenación de archivos: ordenación externa14. Tipos de datos TAD/Objetos
--	---



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE LA COSTA

DIVISIÓN DE INGENIERÍAS

DEPARTAMENTO DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

	<ul style="list-style-type: none">14.1. Tipos de datos14.2. Tipos abstractos de datos14.3. Especificación de los TAD15. Listas enlazadas<ul style="list-style-type: none">15.1. Fundamentos teóricos15.2. Clasificación de las listas enlazadas15.3. Operaciones en listas enlazadas15.4. Listas circulares16. Pilas y colas
Habilidades	Conoce los fundamentos a la programación en lenguaje C
Actitudes	Capacidad para trabajar con responsabilidad social y ética profesional Capacidad de autogestión Capacidad de trabajo colaborativo Capacidad por la investigación y desarrollo tecnológico

8. MODALIDAD DE EVALUACIÓN

	Evidencias o productos	Instrumentos de evaluación	Factor de ponderación
1	Examen teórico	Rúbrica	30%
2	Actividades en clase	Rúbrica	40%
3	Ejercicios resueltos en clase	Rúbrica	30%
	Total		100%

9. FUENTES DE APOYO Y DE CONSULTA (BIBLIOGRAFÍA, HEMEROGRAFÍA, FUENTES ELECTRÓNICAS)

BIBLIOGRAFÍA BÁSICA

Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible (en su caso)
Luis Joyanes	Algoritmos, programación y estructura de datos	Schaum	2011	



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE LA COSTA

DIVISIÓN DE INGENIERÍAS

DEPARTAMENTO DE CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

Corona Nakamura, María Adriana	Diseño de Algoritmos y su codificación en lenguaje C.	McGraw-Hill Interamericana de España	2011	
Luis Joyanes Aguilar	C. Programación y Algoritmos	Mc Graw Hill	2010	
Vozmediano, A.M.	Aprender a programar en C.	Alfredo Vozmediano	2017	

BIBLIOGRAFÍA COMPLEMENTARIA

Autor(es)	Título	Editorial	Año	URL o biblioteca digital donde está disponible (en su caso)
VEGA, Augusto; JEDER, Ismael; LOPEZ, Gustavo	Análisis y diseño de algoritmos - implementaciones en c y pascal	Alfaomega	2010	https://wdg.biblio.udg.mx/

10. PERFIL DEL PROFESOR

El profesor deberá contar como mínimo con una licenciatura afín al área de la Computación. Es importante tenga experiencia en la docencia, como impartición de cursos, talleres o diplomados.

11. PLANEACIÓN POR SEMANAS

Semana	Tema	Contenidos	Actividades para su movilización	Recursos	Evaluación	Temas transversales