

Misión del Centro Universitario

Somos un centro que forma parte de la Red Universitaria de la Universidad de Guadalajara. Como institución de educación superior pública asumimos el compromiso social de satisfacer necesidades de formación y generación de conocimiento en el campo de las ciencias exactas y las ingenierías. La investigación científica y tecnológica, así como la vinculación y extensión, son parte fundamental de nuestras actividades para incidir en el desarrollo de la sociedad; por lo que se realizan con vocación internacional, humanismo, calidad y pertinencia.

1.- Identificación de la Unidad de Aprendizaje

Nombre de la Unidad de Aprendizaje

Seminario de problemas de Programación

Clave de la UA	Modalidad de la UA	Tipo de UA		Valor de créditos	Área de formación
19888	Presencial	Seminario		4	Básica particular
Hora semana		Horas teoría/semestre	Horas práctica/semestre	Total de horas:	Seriación
4		0	68	68	Ninguno
Departamento			Academia		
Departamento de Ciencias Computacionales			Programación		

Presentación

En este curso se estudiarán los conceptos básicos de programación necesarios para codificar programas que respondan a la solución de problemas utilizando el lenguaje de programación C.

Competencia de la Unidad de Aprendizaje (UA)

Codificar programas que resuelven problemas sencillos utilizando el lenguaje de programación C.

Tipos de saberes

Se refiere al desglose de aquellos conocimientos, habilidades, actitudes y valores que se encuentran ligados a la descripción de la competencia, y al desarrollarlos deben observar la parte de los nuevos aprendizajes y capacidades que logrará el estudiante

Saber (conocimientos)	Saber hacer (habilidades)	Saber ser (actitudes y valores)
<ul style="list-style-type: none"> Comprende los elementos básicos del lenguaje de programación C Diseña código eficiente para solucionar problemas sencillos Comprende el funcionamiento de un programa a partir de su código fuente 	<ul style="list-style-type: none"> Abstracción Razonamiento lógico Razonamiento matemático Análisis y síntesis de información Destreza en el uso de la computadora 	<ul style="list-style-type: none"> Es responsable en las fechas de entrega Colabora con sus compañeros. Autogestión a su conocimiento

Competencia genérica

- Pensamiento matemático
- Pensamiento crítico y reflexivo
- Aprendizaje autónomo

Competencia profesional

- Diseño y desarrollo de software

Competencias previas del alumno

<ul style="list-style-type: none"> ● Escucha, interpreta y emite mensajes pertinentes en distintos contextos mediante la utilización de medios, códigos y herramientas apropiados ● Desarrolla innovaciones y propone soluciones a problemas a partir de métodos establecidos. ● Sustenta una postura personal sobre temas de interés y relevancia general, considerando otros puntos de vista de manera crítica y reflexiva. ● Aprende por iniciativa e interés propio a lo largo de la vida. ● Participa y colabora de manera efectiva en equipos diversos.
Competencia del perfil de egreso
<ul style="list-style-type: none"> ● Diseña y desarrolla software
Perfil deseable del docente
<ul style="list-style-type: none"> ● Docente con licenciatura en ingeniería en computación o carrera afín, con conocimientos en programación, y conocimientos del lenguaje de programación C..

2.- Contenidos temáticos
Contenido
<p>Módulo 1. Conceptos básicos del lenguaje C [8 hrs.]</p> <p>1.1 Historia del lenguaje C (1 hr.)</p> <p>1.2 Entorno del compilador (2 hrs.)</p> <p>1.3 Elementos de un programa (5 hrs.)</p> <p>1.3.1 Identificador</p> <p>1.3.2 Tipos de datos primitivos</p> <p>1.3.3 Definición y declaración de datos</p> <p>1.3.3.1 Variables</p> <p>1.3.3.2 Constantes (const, #define)</p> <p>1.3.4 Operadores y expresiones</p> <p>1.3.5 Palabras reservadas</p> <p>1.3.6 Comentarios</p> <p>Módulo 2. Programación estructurada [24 hrs.]</p> <p>2.1 Características (2 hrs.)</p> <p>2.2 Estructuras de control (8 hrs.)</p> <p>2.2.1 Clasificación de estructuras</p> <p>2.2.1.1 Secuencial</p> <p>2.2.1.2 Selectiva</p> <p>2.2.1.3 Selectiva doble</p> <p>2.2.1.4 Selectiva múltiple</p> <p>2.3 Estructuras de control repetitivas o de iteración condicional (10 hrs.)</p> <p>2.3.1 Tipos de estructuras de control</p> <p>2.3.1.1 Contadores, acumuladores y banderas</p> <p>2.3.1.2 Mientras</p> <p>2.3.1.3 Hacer mientras</p> <p>2.3.1.4 Desde</p> <p>2.4 Implementación de estructuras anidadas (4 hrs.)</p> <p>Módulo 3. Arreglos [8 hrs.]</p> <p>3.1 Definición (2 hrs.)</p> <p>3.2 Tipos de arreglos (6 hrs.)</p> <p>3.2.1 Vectores</p> <p>3.2.2 Matrices</p> <p>Módulo 4. Manejo de funciones [16 hrs.]</p> <p>4.1 Definición (2 hrs.)</p> <p>4.2 Tipos de funciones (14 hrs.)</p> <p>4.2.1 Funciones sin paso de parámetros</p>

<p>4.2.2 Funciones con paso de parámetros</p> <p>4.2.2.1 Funciones con parámetros (tipos de datos primitivos)</p> <p>4.2.2.2 Funciones con parámetros (tipo arreglo)</p> <p>4.2.3 Funciones predefinidas</p> <p>4.2.3.1 Matemáticas</p> <p>4.2.3.2 De manejo de caracteres</p> <p>Módulo 5. Registros [12 hrs.]</p> <p>5.1 Definición (2 hrs.)</p> <p>5.2 Operaciones con registros (6 hrs.)</p> <p>5.2.1 Entrada de datos de los elementos de un registro.</p> <p>5.2.2 Salida de datos de un registro</p> <p>5.3 Arreglos de registros (4 hrs.)</p> <p>5.3.1 Definición</p> <p>5.3.2 Operaciones con arreglos de registros</p> <p>5.3.2.1 Entrada de datos de los elementos de un arreglo de registros</p> <p>5.3.2.2 Salida de datos de un arreglo de registros.</p>						
Estrategias docentes para impartir la unidad de aprendizaje						
<ol style="list-style-type: none"> 1. Exponer 2. Resolver ejemplos 3. Proponer ejercicios y problemas 4. Propiciar el trabajo en equipo 						
Bibliografía básica						
<p>Joyanes Aguilar, L. (2014). Programación en C, C++, Java y UML. McGraw Hill. 2a. Edición</p> <p>Juganaru Mathieu, M. (2012). Introducción a la Programación. Grupo Editorial Patria</p> <p>Márquez G., Osorio S., Olvera N. (2011). Introducción a la Programación Estructurada en C. Pearson</p>						
Bibliografía complementaria						
3.-Evaluación						
Evidencias						
<ol style="list-style-type: none"> 1. Entrega archivo electrónico con código fuente de un programa que resuelve expresiones aritméticas 2. Entrega archivo electrónico con código fuente de un programa que aplica sentencias selectivas para la solución de un problema 3. Entrega archivo electrónico con código fuente de un programa que aplica sentencias repetitivas para la solución de un problema 4. Entrega archivo electrónico con código fuente de un programa que aplica arreglos para la solución de un problema 5. Entrega archivo electrónico con código fuente de un programa que aplica funciones para la solución de un problema 6. Entrega archivo electrónico con código fuente de un programa que aplica registros para la solución de un problema 						
Tipo de evaluación						
Heteroevaluación procedimental						
Criterios de Evaluación (% por criterio)						
<table> <tr> <td>Entregables</td> <td style="text-align: right;">50%</td> </tr> <tr> <td>Actividades complementarias</td> <td style="text-align: right;">30%</td> </tr> <tr> <td>Proyecto</td> <td style="text-align: right;">20%</td> </tr> </table>	Entregables	50%	Actividades complementarias	30%	Proyecto	20%
Entregables	50%					
Actividades complementarias	30%					
Proyecto	20%					
4.-Acreditación						
Tener por lo menos el 80% de asistencia a clases						

Obtener calificación aprobatoria en la unidad de aprendizaje

Tener por lo menos 65% de asistencia a clases
Obtener calificación aprobatoria en el examen extraordinario

5.- Participantes en la revisión y actualización

Fecha de revisión y actualización: