



**1. DATOS DE IDENTIFICACIÓN DE LA UNIDAD DE APRENDIZAJE**

NOMBRE DE LA UNIDAD DE APRENDIZAJE						
<b>Estructura de Datos</b>						
CLAVE	CRÉDITOS	CARGA HORARIA			PRERREQUISITOS	SERIACIÓN
		TEORÍA	PRÁCTICA	TOTALES		
ID933	8	40	40	80	Programación Estructurada	Programación Avanzada
ÁREA DE FORMACIÓN:		TIPO		MODALIDAD	NIVEL	
<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		<input type="checkbox"/> Curso <input checked="" type="checkbox"/> Curso-taller <input type="checkbox"/> Taller <input type="checkbox"/> Laboratorio <input type="checkbox"/> Curso-laboratorio		<input checked="" type="checkbox"/> Presencial <input type="checkbox"/> Mixta <input type="checkbox"/> Distancia (En Línea)	<input type="checkbox"/> Técnico Superior <input checked="" type="checkbox"/> Licenciatura <input type="checkbox"/> Posgrado	
CARRERA		ACADEMIA		DEPARTAMENTO		
Ingeniería en Electrónica y Computación		Ciencias Computacionales		Fundamentos del Conocimiento		
ELABORACIÓN						
NOMBRE DEL PROFESOR				FECHA		
ACTUALIZACIÓN						
NOMBRE DEL PROFESOR				FECHA		
Mtro. Rodolfo Camacho Velazquez				14 de febrero de 2020		
Mtro. Diego Alberto Rodríguez Cuadros				15 de Octubre de 2020		

**2. RELACIÓN CON EL PERFIL EGRESO**

En el mundo de la programación la estructura de datos es una parte importante en el ámbito del proceso computacional, esta unidad de aprendizaje contribuye de manera importante a poder resolver problemas de la programación los cuales son esenciales en la carrera. Se busca que el alumno analice, interprete y realice de forma individual lo aprendido en la teoría y desarrolle los productos solicitados de acuerdo con sus capacidades, sobre todo enfocado en el ámbito de la programación. Las horas dedicadas en el transcurso del curso, complementadas con el tiempo usado en horas de laboratorio, darán al egresado herramientas con la cuales podrá analizar e identificar dónde se utiliza este tipo de estructuras al momento de realizar programas con lo que podrá realizar la mejora de procesos en el campo de acción al egresar.

**3. RELACIÓN CON EL PLAN DE ESTUDIOS**

La unidad de aprendizaje busca en los estudiantes desarrollar habilidades sobre el conocimiento del lenguaje de programación, pieza importante en el desarrollo de la Ingeniería en Electrónica y Computación. Con las bases obtenidas con la materia anterior se busca desarrollar la capacidad de analizar problemas que se presenten y conocer los tipos de estructuras que existen, así como su funcionamiento necesarias para el desarrollo del perfil del alumno.



#### 4. PROPÓSITO

Al finalizar el curso el alumno relacionará los conceptos de arreglos, estructuras lineales y no lineales, recursividad, ordenamiento y búsqueda de manera propia para aplicarlos en problemáticas mediante la utilización del lenguaje de programación C++, a fin de que fortalezcan el dominio y manejo para apoyar la construcción de su conocimiento.

#### 5. COMPETENCIAS A LAS QUE CONTRIBUYE

##### a. COMPETENCIAS GENERICAS

- |                                     |   |
|-------------------------------------|---|
| <input type="checkbox"/>            | Capacidad para la comunicación oral y escrita;                          |
| <input checked="" type="checkbox"/> | Capacidad para la resolución de problemas;                              |
| <input type="checkbox"/>            | Capacidad para comunicarse en un segundo idioma;                        |
| <input type="checkbox"/>            | Capacidad de trabajo colaborativo;                                      |
| <input checked="" type="checkbox"/> | Capacidad para trabajar con responsabilidad social y ética profesional; |
| <input type="checkbox"/>            | Capacidad de autogestión;   |
| <input checked="" type="checkbox"/> | Capacidad de crear, innovar y emprender;                                |
| <input checked="" type="checkbox"/> | Capacidad por la investigación y desarrollo tecnológico.                |

##### b. COMPETENCIAS ESPECÍFICAS

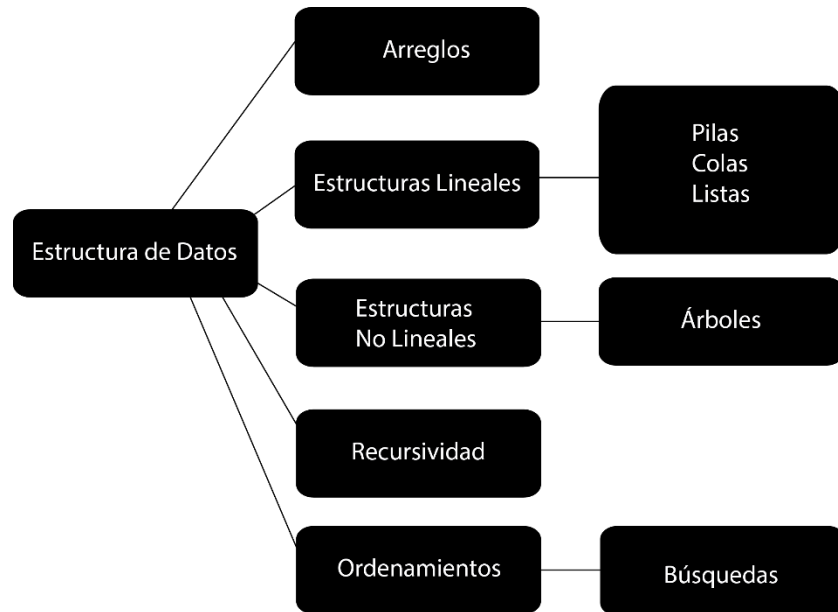
- |                                     |  |
|-------------------------------------|--|
| <input type="checkbox"/>            | Dominio de los principios básicos de la física vinculados con su profesión;  |
| <input type="checkbox"/>            | Aplicación de conocimientos matemáticos para la resolución de problemas vinculados con la ingeniería;                              |
| <input checked="" type="checkbox"/> | Dominio de lenguajes de programación.  |
| <input checked="" type="checkbox"/> | Uso y programación de las computadoras, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería; |
| <input type="checkbox"/>            | Diseño de sistemas electrónicos, analógicos y digitales;   |
| <input type="checkbox"/>            | Diseño y manejo de sistemas de control;  |
| <input checked="" type="checkbox"/> | Desarrollo y aplicación de algoritmos computacionales.   |

##### c. COMPETENCIAS ESPECIALIZANTES

- |                                     |  |
|-------------------------------------|--|
| <input type="checkbox"/>            | Diseño y administración de sistemas de telecomunicación;       |
| <input checked="" type="checkbox"/> | Diseño de sistemas embebidos mediante lenguajes de alto nivel; |
| <input type="checkbox"/>            | Diseño de sistemas optoelectrónicos.                           |
| <input checked="" type="checkbox"/> | Diseño de sistemas interactivos y videojuegos                  |



**6. REPRESENTACION GRÁFICA**



**7. ESTRUCTURACIÓN DE LA UNIDAD DE APRENDIZAJE**

**7.1. COMPETENCIA GENERAL:**

El alumno será capaz de construir programas de cómputo utilizando estructuras de datos para la solución de problemas en el campo laboral.

**7.2. PRODUCTO INTEGRADOR:**

El alumno elaborará un programa en el que demuestre la utilización de lo visto durante la unidad de aprendizaje, debe manipular datos ingresados por el usuario, almacenarlos, ordenarlos y buscar en específico algunos de ellos. El producto debe de cumplir estos puntos, dejando en libertad del estudiante las formas de ordenación y de búsqueda que más sea de su elección.

UNIDAD DE COMPETENCIA I	
<b>COMPETENCIA ESPECÍFICA:</b>	
Interpreta el manejo de arreglos en el lenguaje programación C++ con la finalidad de elaborar una correcta estructuración a través de la elaboración de programas.	
<b>PRODUCTO INTEGRADOR:</b>	
Debe elaborar un programa utilizando el uso de arreglos.	
<b>CONOCIMIENTOS: (Saberes teóricos)</b>	1. Punteros-apuntadores 1.1 Punteros vs Arreglos 1.2 Como se almacenan las variables en memoria según el tipo de dato. 1.3 Uso de punteros para acceder a la dirección de memoria



# UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DEL NORTE

DEPARTAMENTO DE FUNDAMENTOS DEL CONOCIMIENTO

	<p>de una variable.</p> <p>1.4.1 Propiedades de los punteros.</p> <p>1.4.2 Puntero a puntero</p> <p>1.4.3 Funciones con paso a puntero</p> <p>1.4.4 Funciones con regreso de punteros</p>
<b>HABILIDADES: (Saberes prácticos)</b>	<ol style="list-style-type: none"> <li>1. Argumenta el análisis y la síntesis.</li> <li>2. Resuelve problemas.</li> <li>3. Desarrolla los conocimientos teóricos a la práctica</li> <li>4. Genera nuevas ideas.</li> </ol>
<b>ACTITUDES Y VALORES: (Saberes formativos)</b>	<ol style="list-style-type: none"> <li>1. Compromiso ético.</li> <li>2. Integra lo solicitado.</li> <li>3. Cuida y respeta los tiempos de entrega.</li> </ol>

## UNIDAD DE COMPETENCIA 2

### COMPETENCIA ESPECÍFICA:

Es capaz de aplicar recursividad para la elaboración de programas a través del uso del lenguaje de programación C++.

### PRODUCTO INTEGRADOR:

Elaboración de un programa haciendo uso de los algoritmos de recursividad atendiendo la sintaxis y estructurados de forma correcta.

<b>CONOCIMIENTOS: (Saberes teóricos)</b>	<ol style="list-style-type: none"> <li>2. Recursividad             <ol style="list-style-type: none"> <li>2.1. Definición y características de la recursividad</li> <li>2.2. Llamadas recursivas directas e indirectas</li> <li>2.3. Comparación entre funciones iterativas y recursivas</li> </ol> </li> </ol>
<b>HABILIDADES: (Saberes prácticos)</b>	<ol style="list-style-type: none"> <li>1. Interpreta de manera correcta la sintaxis en el uso de recursividad.</li> <li>2. Diseña de forma correcta las estructuras de recursividad.</li> <li>3. Resuelve lo solicitado durante el objeto de estudio.</li> <li>4. Fundamenta lo realizado dentro del producto integrador.</li> </ol>
<b>ACTITUDES Y VALORES: (Saberes formativos)</b>	<ol style="list-style-type: none"> <li>1. Elabora lo solicitado de manera funcional y sin errores.</li> <li>2. Procura que lo solicitado este completo.</li> <li>3. Integra lo aprendido durante la unidad de aprendizaje.</li> <li>4. Respeta los tiempos de entrega.</li> </ol>

## UNIDAD DE COMPETENCIA 3

### COMPETENCIA ESPECÍFICA:

El alumno es capaz de utilizar las estructuras lineales de datos con la finalidad de hacer uso correcto del manejo de sintaxis a través de un programa en C++.

### PRODUCTO INTEGRADOR:

Elaboración de programa con el uso de las estructuras enlistadas en la unidad de aprendizaje.

<b>CONOCIMIENTOS: (Saberes teóricos)</b>	<ol style="list-style-type: none"> <li>3. Estructuras lineales de datos con Memoria Dinámica y Estática</li> </ol>
--	--



# UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DEL NORTE

DEPARTAMENTO DE FUNDAMENTOS DEL CONOCIMIENTO

	<ul style="list-style-type: none"> <li>3.1. Definición de Estructura de datos</li> <li>3.2. Listas</li> <li>3.3. Pilas</li> <li>3.4. Colas</li> </ul>
<b>HABILIDADES: (Saberes prácticos)</b>	<ul style="list-style-type: none"> <li>1. Resuelve problemas.</li> <li>2. Programa lo solicitado.</li> <li>3. Detecta complicaciones al momento de realizar los programas</li> </ul>
<b>ACTITUDES Y VALORES: (Saberes formativos)</b>	<ul style="list-style-type: none"> <li>1. Crea lo solicitado de manera funcional y sin errores.</li> <li>2. Cuida y respeta los tiempos de entrega.</li> </ul>

## UNIDAD DE COMPETENCIA 4

### COMPETENCIA ESPECÍFICA:

Analiza ordenamientos y búsquedas para identificar el funcionamiento de cada uno de ellos elaborando programas a través del uso del lenguaje de programación C++.

### PRODUCTO INTEGRADOR:

Elaboración un programa con el uso de los ordenamientos y búsqueda de datos.

<b>CONOCIMIENTOS: (Saberes teóricos)</b>	<ul style="list-style-type: none"> <li>4. Ordenamiento y búsquedas               <ul style="list-style-type: none"> <li>4.1. Ordenamiento burbuja</li> <li>4.2. Ordenamiento quicksort</li> <li>4.3. Ordenamiento shell</li> <li>4.4. Búsqueda secuencial</li> <li>4.5. Búsqueda binaria</li> </ul> </li> </ul>
<b>HABILIDADES: (Saberes prácticos)</b>	<ul style="list-style-type: none"> <li>1. Diseña cada uno de los ordenamientos.</li> <li>2. Estructura la forma de cada uno de los ordenamientos.</li> <li>3. Fundamenta lo realizado en el producto integrador.</li> <li>4. Programa lo visto en el objeto de estudio.</li> </ul>
<b>ACTITUDES Y VALORES: (Saberes formativos)</b>	<ul style="list-style-type: none"> <li>1. Crea lo solicitado de manera funcional y sin errores.</li> <li>2. Procura que lo solicitado este completo.</li> <li>3. Integra los aprendido durante la unidad de aprendizaje.</li> <li>4. Respeta los tiempos de entrega.</li> </ul>

## UNIDAD DE COMPETENCIA 5

### COMPETENCIA ESPECÍFICA:

Analiza los diferentes tipos de estructuras de datos no lineales para conocer su funcionamiento a través del lenguaje de programación C++

### PRODUCTO INTEGRADOR:

Elaboración de un programa con la utilización de los tipos de árboles.

<b>CONOCIMIENTOS: (Saberes teóricos)</b>	<ul style="list-style-type: none"> <li>5. Estructura de Datos no Lineales               <ul style="list-style-type: none"> <li>5.1. Grafos</li> <li>5.2. Árboles                   <ul style="list-style-type: none"> <li>5.2.1.Árbol binario</li> <li>5.2.2.Árbol binario de Búsqueda</li> </ul> </li> </ul> </li> </ul>
--	---



# UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DEL NORTE

DEPARTAMENTO DE FUNDAMENTOS DEL CONOCIMIENTO

<b>HABILIDADES: (Saberes prácticos)</b>	<ol style="list-style-type: none"><li>1. Diseña árboles para ejemplificar su funcionamiento.</li><li>2. Argumenta los productos solicitados.</li><li>3. Resuelve los solicitado en la unidad de aprendizaje.</li><li>4. Interpreta de manera correcta conceptos utilizados en la unidad de aprendizaje.</li></ol>
<b>ACTITUDES Y VALORES: (Saberes formativos)</b>	<ol style="list-style-type: none"><li>1. Elabora lo solicitado de manera funcional y sin errores.</li><li>2. Respeta que lo entregado este completo.</li><li>3. Argumento los problemas que encuentra.</li><li>4. Integra lo aprendido durante la unidad de aprendizaje.</li><li>5. Respeta los tiempos de entrega.</li></ol>

## 8. EVALUACIÓN

Actividades Preliminares:	20 %
Actividades Aprendizaje:	20 %
Actividades Integradoras	30%
Producto Integrador	30%

## 9. FUENTES DE APOYO Y CONSULTA

### 9.1. BÁSICAS

BIBLIOGRAFÍA			
Autor (es)	Título	Editorial	Año
Joyanes Aguilar, Luis	Programación en C, C++, Java y UML	McGraw-Hill	2014
Trejos Buriticá, Omar Iván	Lógica de programación	Ediciones de la U	2017
Pantoja Libardo	Estructuras de datos dinámicos: una forma fácil de aprender	Ra-ma	2017
Augusto Sznajdleder, Pablo	Programación estructurada a fondo : implementación de algoritmos en C	Alfaomega	2017

### 9.2. COMPLEMENTARIA

BIBLIOGRAFÍA			
Autor (es)	Título	Editorial	Año
Deitel Paul, Deitel Harvey	Cómo Programar en C/C++ y Java	Person	2004
Méndez Girón, Alejandra Margarita	Diseño de algoritmos y su programación en C	Alfaomega	2013
Deitel Paul, Deitel Harvey	C++ Cómo programar	Person	2014
Joyanes Aguilar, Luis	Programación en C: metodología, algoritmos y estructura de datos	McGraw-Hill	2001



UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DEL NORTE

DEPARTAMENTO DE FUNDAMENTOS DEL CONOCIMIENTO

## 10. PERFIL DEL PROFESOR

El profesor con formación académica en Ingeniería en Computación o a fin, enfocado en el uso de lenguajes de programación.