

### ASIGNATURA DE PRINCIPIOS DE PROGRAMACIÓN

1. <b>Competencias</b>	Administrar el programa de perforación de pozos considerando la normatividad y los procedimientos establecidos para optimizar los recursos, humanos y materiales durante la perforación, terminación y reparación de pozos.
2. <b>Cuatrimestre</b>	Cuarto
3. <b>Horas Teóricas</b>	15
4. <b>Horas Prácticas</b>	30
5. <b>Horas Totales</b>	45
6. <b>Horas Totales por Semana Cuatrimestre</b>	3
7. <b>Objetivo de aprendizaje</b>	El alumno programará equipos de PLC, Robots y microcontroladores, para atender las necesidades de mantenimiento que se presentan en la industria, mediante un lenguaje de programación de alto nivel.

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
I. <b>Lógica de programación</b>	4	8	12
II. <b>Fundamentos de programación</b>	6	12	18
III. <b>Tópicos específicos</b>	5	10	15
	<b>15</b>	<b>30</b>	<b>45</b>

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de TSU en Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T. y P.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## UNIDADES DE APRENDIZAJE

1. <b>Unidad de aprendizaje</b>	<b>I. Lógica de programación</b>
2. <b>Horas Teóricas</b>	4
3. <b>Horas Prácticas</b>	8
4. <b>Horas Totales</b>	12
5. <b>Objetivo de la Unidad de Aprendizaje</b>	El alumno formulará la solución a problemas relacionados con el mantenimiento, mediante el uso de algoritmos y diagramas de flujo, para su implementación con un lenguaje de programación.

Temas	Saber	Saber hacer	Ser
Algoritmos	Describir las características de los algoritmos.	Desarrollar algoritmos para la solución de problemas:  - Casos prácticos de la vida real - Mantenimiento o reparación de un equipo	Analítico Ordenado Proactivo
Diagramas de flujo	Identificar la simbología y estructura de los diagramas de flujo.  Identificar software de simulación de diagramas de flujos.	Construir diagramas de flujo para la solución de problemas tales como:  - Casos prácticos de la vida real - Mantenimiento o reparación de un equipo  -Emplear software de simulación de diagramas de flujos	Analítico Ordenado Proactivo

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Elabora diagramas de flujo basados en algoritmos, para la solución de problemas de mantenimiento, mediante el empleo de la simbología estándar.</p> <p>Elaborar simulaciones mediante el empleo de software de diagramas de flujos.</p>	<ol style="list-style-type: none"><li>1. Identificar algoritmos de programación que dan solución de problemas específicos</li><li>2. Comprender los conceptos de entradas y salidas de un problema</li><li>3. Identificar los elementos empleados en diagramas de flujo</li><li>4. Formular y verificar con la ayuda de software un diagrama de flujo como solución correcta a un problema</li></ol>	<p>Ejercicios prácticos</p> <p>Lista de verificación</p>

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Solución de problemas Tareas de investigación Mapa mental	Cañón Computadora Internet Software

### ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## UNIDADES DE APRENDIZAJE

1. <b>Unidad de aprendizaje</b>	<b>II. Fundamentos de programación</b>
2. <b>Horas Teóricas</b>	6
3. <b>Horas Prácticas</b>	12
4. <b>Horas Totales</b>	18
5. <b>Objetivo de la Unidad de Aprendizaje</b>	El alumno diseñará programas de computadora mediante el empleo de las reglas sintácticas y semánticas del lenguaje de programación de alto nivel seleccionado para dar solución a problemas de mantenimiento.

Temas	Saber	Saber hacer	Ser
Tipos de datos, operadores y expresiones	Describir las características de los tipos de datos, operadores y expresiones disponibles en un lenguaje de alto nivel.	Seleccionar los tipos de datos adecuados para almacenamiento y procesamiento de datos.  Construir expresiones con operadores, constantes y variables para la solución de problemas.	Analítico Ético Ordenado Observador Proactivo
Estructura de un programa	Describir los elementos que conforman la estructura de un programa para computadora.	Establecer los elementos que conforman la estructura de un programa para computadora.	Analítico Ético Ordenado Observador Proactivo
Decisión y repetición	Conocer las alternativas para decisión y repetición que incorpora el lenguaje en uso.	Construir programas que empleen decisiones y repeticiones.	Analítico Ético Ordenado Observador Proactivo

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

Temas	Saber	Saber hacer	Ser
Funciones	Describir la sintaxis empleada en la definición y uso de funciones.	Diseñar programas que empleen funciones.  Verificar la sintaxis de los programas elaborados para su depuración, corrigiendo la sintaxis y mejorando el estilo de programación.	Analítico Ético Ordenado Observador Proactivo

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>A partir de un caso dado, elabora y demuestra el funcionamiento de un programa usando lenguaje de alto nivel que incluya:</p> <ul style="list-style-type: none"><li>- Operadores, constantes y variables</li><li>- Decisiones y repeticiones</li><li>- Funciones</li></ul>	<ol style="list-style-type: none"><li>1. Identificar los tipos de datos requeridos en un problema</li><li>2. Interpretar las reglas sintácticas y semánticas del lenguaje de programación</li><li>3. Comprender el proceso para la generación de un programa</li><li>4. Identificar los errores en un programa y la forma de corregirlos</li></ol>	<p>Ejercicios prácticos Lista de verificación</p>

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Solución de problemas Tareas de investigación Mapa mental	Cañón Computadora con lenguaje de programación de alto nivel Internet

### ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## UNIDADES DE APRENDIZAJE

1. <b>Unidad de aprendizaje</b>	<b>III. Tópicos específicos</b>
2. <b>Horas Teóricas</b>	5
3. <b>Horas Prácticas</b>	10
4. <b>Horas Totales</b>	15
5. <b>Objetivo de la Unidad de Aprendizaje</b>	El alumno diseñará programas integrando arreglos, cadenas y bits, mediante los comandos, tipos y operadores que proporcione el lenguaje de programación empleado, para dar solución a problemas de mantenimiento.

Temas	Saber	Saber hacer	Ser
Arreglos	Describir la forma de declaración de arreglos de una o más dimensiones.	Integrar a un programa arreglos de una o más dimensiones.	Analítico Ético Ordenado Observador Proactivo
Cadenas	Describir la forma de implementar cadenas en el lenguaje de programación, así como las funciones y operaciones que se aplican a ellas.	Integrar cadenas a un programa de computadora.  Verificar la sintaxis de los programas elaborados para su depuración, corrigiendo la sintaxis y mejorando el estilo de programación.	Analítico Ético Ordenado Observador Proactivo
Manipulación de bits	Describir las operaciones y aplicaciones para manipulación de bits en el lenguaje de programación.	Diseñar programas que incorporen la manipulación de bits.  Verificar la sintaxis de los programas elaborados para su depuración, corrigiendo la sintaxis y mejorando el estilo de programación.	Analítico Ético Ordenado Observador Proactivo

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
A partir de un caso dado, crea y demuestra el funcionamiento de un programa de computadora en lenguaje de alto nivel que incluya arreglos, cadenas y manipulación de bits.	<ol style="list-style-type: none"><li>1. Comprender las reglas para uso y manipulación de arreglos, cadenas y bits</li><li>2. Comprender la sintaxis para la incorporación de cadenas, arreglos y manipulación en programas de computadora</li><li>3. Identificar los errores en programas que manipulen arreglos, cadenas o bits, y la forma de corregirlos</li><li>4. Generar programas incorporando arreglos, cadenas y manipulación de bits</li></ol>	Estudio de casos Lista de verificación

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
Solución de problemas Tareas de investigación Mapa mental	Cañón computadora con lenguaje de programación de alto nivel Internet

### ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
	X	

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

## PRINCIPIOS DE PROGRAMACIÓN

### CAPACIDADES DERIVADAS DE LAS COMPETENCIAS PROFESIONALES A LAS QUE CONTRIBUYE LA ASIGNATURA

Capacidad	Criterios de Desempeño
<p>Verificar el trabajo ejecutado y el funcionamiento de las partes y componentes de sistemas electromecánicos corregidos de acuerdo con las condiciones de operación, especificaciones técnicas del fabricante y a las políticas establecidas para asegurar la prestación óptima del servicio.</p>	<p>Elabora y aplica lista de verificación que incluye:</p> <p>Para el trabajo realizado:</p> <ul style="list-style-type: none"> <li>- Que las actividades se han realizado de acuerdo al procedimiento establecido</li> <li>- Que se utilizaron las herramientas y materiales adecuados</li> <li>- Que las actividades se realizaron de acuerdo a la normatividad aplicable</li> </ul> <p>Para el funcionamiento:</p> <ul style="list-style-type: none"> <li>- Medición de los parámetros de funcionamiento (según sea el caso, presión, temperatura, alimentación, potencia, rpm, entre otros)</li> <li>- Los compara los parámetros del fabricante</li> <li>- Realiza los ajustes necesarios</li> <li>- Valida el trabajo realizado</li> </ul>
<p>Determinar el funcionamiento de partes y componentes de acuerdo a especificaciones del fabricante, políticas de la organización y al programa de mantenimiento, para valorar la funcionalidad del sistema.</p>	<p>Elabora un reporte técnico de funcionamiento que incluye:</p> <ul style="list-style-type: none"> <li>- Tipo de parte o componente</li> <li>- Descripción del componente y su interrelación con otros componentes</li> <li>- Resultados de pruebas funcionales a la maquinaria, comparación los resultados con las especificaciones del fabricante</li> <li>- Determina si se encuentran dentro de los parámetros de funcionamiento</li> </ul>

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	

# PRINCIPIOS DE PROGRAMACIÓN

## FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Joyanes, Aguilar Luis	(2006)	<i>Programación en C++</i>	Madrid	España	McGraw-Hill
Joyanes, Aguilar Luis , Lucas Sánchez García	(2006)	<i>Programación en C++, serie SCHAUM</i>	Madrid	España	McGraw-Hill
LEE, R.C.T.	(2007)	<i>Introducción al Diseño y Análisis de Algoritmos</i>	California	USA	McGraw-Hill
VV. AA.	(2005)	<i>C: Algoritmos, Programación y Estructura de Datos</i>	Madrid	España	Schaum-McGraw-Hill
Sierra Urrecho, Alejandro Y Alfonseca Moreno, Manuel	(2005)	<i>Programación en C/C++ (Guías Prácticas)</i>	Madrid	España	Anaya Multimedia
VV. AA.	(2005)	<i>Programación en C</i>	Madrid	España	McGraw-Hill
J. Baltasar García / Rosalía Laza	(2008)	<i>Metodología y tecnología de la programación</i>	Madrid	España	Pearson, Prentice Hall
Charte, Ojeda Francisco	(2009)	<i>Programación Ensamblador para DOS, Linux y Windows</i>	Madrid	España	Anaya Multimedia
Charte, Ojeda Francisco	(2009)	<i>Programación Ensamblador</i>	Madrid	España	Anaya Multimedia

<b>ELABORÓ:</b>	Comité de Directores de la Carrera de Mantenimiento	<b>REVISÓ:</b>	Dirección Académica	
<b>APROBÓ:</b>	C. G. U. T.	<b>FECHA DE ENTRADA EN VIGOR:</b>	Septiembre de 2018	