


ASIGNATURA DE METODOLOGÍA DE LA PROGRAMACIÓN

1. Competencias	Desarrollar soluciones tecnológicas para entornos Web mediante fundamentos de programación orientada a objetos, base de datos y redes de área local que atiendan las necesidades de las organizaciones.
2. Cuatrimestre	Primero
3. Horas Teóricas	23
4. Horas Prácticas	52
5. Horas Totales	75
6. Horas Totales por Semana Cuatrimestre	5
7. Objetivo de aprendizaje	El alumno diseñará algoritmos, diagramas de flujo y pseudocódigos para generar programas computacionales.

Unidades de Aprendizaje	Horas		
	Teóricas	Prácticas	Totales
I. Conceptos básicos	3	2	5
II. Expresiones	5	10	15
III. Algoritmos, diagramas de flujo y pseudocódigos	15	40	55
Totales	23	52	75

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	I. Conceptos básicos
2. Horas Teóricas	3
3. Horas Prácticas	2
4. Horas Totales	5
5. Objetivo de la Unidad de Aprendizaje	El alumno determinará las entradas, procesos y salidas de un problema utilizando los elementos básicos de programación para el planteamiento de una solución.


Temas	Saber	Saber hacer	Ser
Adquisición, procesamiento y almacenamiento de datos.	Identificar los elementos básicos del proceso de información.		Analítico. Sistemático. Ordenado.
Conceptos de algoritmos, diagramas de flujo y pseudocódigos.	Definir las características de los algoritmos, diagramas de flujo y pseudocódigos.	Determinar las técnicas adecuadas para la solución de problemas mediante algoritmos.	Analítico. Sistemático. Ordenado. Asertivo.
Tipos de datos (numéricos, caracteres, cadenas y booleanos).	Identificar los diferentes tipos de datos utilizados en la programación, aplicándolo a la resolución de un problema.	Determinar los tipos de datos a utilizar de acuerdo al problema presentado.	Analítico. Sistemático. Ordenado.
Identificadores, variables y constantes.	Describir los conceptos de identificador, variable y constante.	Determinar las variables y constantes a utilizar, dependiendo el problema presentado.	Analítico. Sistemático. Ordenado.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Realiza una serie de ejercicios que incluya:</p> <ul style="list-style-type: none">• Valores de entrada, proceso y salida.• Tipos de datos a utilizar en variables y constantes.• Operaciones a utilizar.	<ol style="list-style-type: none">1. Analizar el proceso de información: Entrada, proceso y salida.2. Identificar los tipos de datos que procesa la computadora.3. Comprender los elementos de entrada, proceso y salida de problemas determinados.	<ul style="list-style-type: none">- Ejercicios prácticos.- Listas de cotejo.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


METODOLOGÍA DE LA PROGRAMACIÓN

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none">- Práctica demostrativa.- Mapas conceptuales.- Discusión en grupo.	<ul style="list-style-type: none">Pizarrón.Plumones.Computadora.Internet.Equipo multimedia.Ejercicios prácticos.Plataformas virtuales.

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X		

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	II. Expresiones
2. Horas Teóricas	5
3. Horas Prácticas	10
4. Horas Totales	15
5. Objetivo de la Unidad de Aprendizaje	El alumno convertirá expresiones algebraicas, aritméticas y lógicas a expresiones algorítmicas considerando la jerarquía de operadores para resolver problemas determinados.


Temas	Saber	Saber hacer	Ser
Operadores aritméticos, relacionales y lógicos.	Identificar los tipos de operadores existentes.	Seleccionar los diferentes tipos de operadores, de acuerdo a la problemática presentada.	Analítico. Lógico. Ordenado. Sistemático.
Jerarquía de los operadores.	Identificar la jerarquía de los operadores.	Realizar expresiones aritméticas y lógicas, utilizando la jerarquía de los operadores.	Analítico. Lógico. Ordenado. Sistemático.
Expresiones.	Identificar las expresiones aritméticas y lógicas.	Proponer ecuaciones algorítmicas a partir de expresiones algebraicas, aritméticas y lógicas a expresiones para resolver un problema dado.	Analítico. Lógico. Ordenado. Sistemático.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Realiza una serie de ejercicios que incluya:</p> <ul style="list-style-type: none">Conversiones de expresiones algebraicas, aritméticas y lógicas a expresiones algorítmicas, contemplando la jerarquía de los operadores.	<ol style="list-style-type: none">Identificar la función y jerarquía de los operadores.Comprender la función y jerarquía de los operadores.Analizar las expresiones algebraicas, aritméticas y lógicas.Comprender las expresiones algebraicas, aritméticas, lógicas y convertirlas a expresiones algorítmicas.	<ul style="list-style-type: none">Ejercicios prácticos.Listas de cotejo.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


METODOLOGÍA DE LA PROGRAMACIÓN

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Práctica demostrativa. - Mapas conceptuales. - Solución de problemas. 	<p>Pizarrón. Plumones. Computadora. Internet. Equipo multimedia. Ejercicios prácticos. Plataformas virtuales. Software para el desarrollo de pseudocódigo.</p>

ESPACIO FORMATIVO

Aula	Laboratorio / Taller	Empresa
X		


ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


METODOLOGÍA DE LA PROGRAMACIÓN

UNIDADES DE APRENDIZAJE

1. Unidad de aprendizaje	III. Algoritmos, diagramas de flujo y pseudocódigos.
2. Horas Teóricas	15
3. Horas Prácticas	40
4. Horas Totales	55
5. Objetivo de la Unidad de Aprendizaje	El alumno elaborará algoritmos, diagramas de flujo y pseudocódigos para la resolución de problemas.

Temas	Saber	Saber hacer	Ser
Reglas para elaborar diagramas de flujo y pseudocódigos.	Identificar la sintaxis utilizada para el desarrollo de diagramas de flujo y pseudocódigos. Identificar el procedimiento para realizar una prueba de escritorio.	Proponer soluciones a problemas empleando algoritmos.	Analítico. Lógico. Ordenado. Sistemático.
Contadores y acumuladores.	Distinguir el funcionamiento de contadores y acumuladores.	Establecer contadores y acumuladores en la solución de problemas.	Analítico. Lógico. Ordenado. Sistemático.
Estructuras de control de selección.	Describir el funcionamiento de las estructuras de control selectivas: - Simples (if - else). - Compuestas (if anidado). - Múltiples (switch).	Proponer las estructuras de control selectivas para un problema determinado.	Analítico. Lógico. Ordenado. Sistemático.
Estructuras de control de repetición.	Describir el funcionamiento de las estructuras de control repetitivas: - Repetir (for). - Mientras (while). - Repetir - hasta (do - while).	Proponer las estructuras de control repetitivas para un problema determinado.	Analítico. Lógico. Ordenado. Sistemático.
Implementación de algoritmos en un lenguaje de programación.	Identificar la sintaxis para declaración de variables, constantes, expresiones, operadores, estructuras de control selectivas y de repetición aplicadas a un lenguaje de programación.	Realizar programas que incluyan declaración de variables, constantes, expresiones, operadores, estructuras de control selectivas y de repetición a partir de un algoritmo.	Analítico. Lógico. Ordenado. Sistemático.


ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

PROCESO DE EVALUACIÓN

Resultado de aprendizaje	Secuencia de aprendizaje	Instrumentos y tipos de reactivos
<p>Entrega un portafolio de evidencias que incluya:</p> <ul style="list-style-type: none"> Los diagramas de flujo o pseudocódigos donde se apliquen las estructuras de control de selección, estructuras de control repetitivas y pruebas de escritorio. La codificación documentada de los diagramas de flujo o pseudocódigos (programas) y resultado de la ejecución. 	<ol style="list-style-type: none"> Analizar el problema. Identificar elementos de entrada, proceso y salida. Identificar las variables, constantes, estructuras de control de selección y repetición. Comprender la solución de un problema mediante un algoritmo. Comprender el proceso de desarrollo de pruebas de escritorio. Comprender la sintaxis de un lenguaje de programación, para codificar y documentar de acuerdo al pseudocódigo creado. 	<ul style="list-style-type: none"> Ejercicios prácticos. Listas de cotejo.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	


METODOLOGÍA DE LA PROGRAMACIÓN

PROCESO ENSEÑANZA APRENDIZAJE

Métodos y técnicas de enseñanza	Medios y materiales didácticos
<ul style="list-style-type: none"> - Práctica demostrativa. - Mapas conceptuales. - Solución de problemas. 	<ul style="list-style-type: none"> Pizarrón. Plumones. Computadora. Internet. Equipo multimedia. Ejercicios prácticos. Plataformas virtuales. Diagramadores. IDE de desarrollo.

ESPACIO FORMATIVO


Aula	Laboratorio / Taller	Empresa
X	X	

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

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


Capacidad	Criterios de Desempeño
Codificar aplicaciones Web a través de los fundamentos de programación orientada a objetos y conexión a base de datos para desarrollarla.	Entrega el código fuente documentado de la aplicación web: <ul style="list-style-type: none">• Métodos.• Atributos.• Variables.• Conexión a la base de datos.• Componentes.

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

FUENTES BIBLIOGRÁFICAS


Autor	Año	Título del Documento	Ciudad	País	Editorial
José Luis López Goytia	2014 ISBN: 9786074387711	<i>Programación Orientada a Objetos Con C++ y Java: Un Acercamiento Interdisciplinario</i>	D.F.	México	Grupo Editorial Patria
Uno López Takeyas	2016 ISBN: 978-607-622-659-9	Curso de Programación Orientada a Objetos En C#.Net. Ejemplos con Aplicaciones Visuales y de Consola	D.F.	México	Alfaomega Grupo Editor
David J. Barnes	2017 ISBN: 9788490355312	<i>Programación Orientada a Objetos Con Java Tm Usando Bluej</i>	Madrid	España	Pearson Educacion
Báez López, David; Cervantes Villagómez, Ofelia; Arízaga Silva, Juan Antonio ; Castillo Juárez, Esteban	2017 ISBN: 978-607-622-673-5	<i>Python Con Aplicaciones a Las Matemáticas, Ingeniería y Finanzas</i>	D.F.	México	Alfaomega Grupo Editor
José Alfredo Jiménez	2104 ISBN: 9786076222027	<i>Fundamentos de Programación: Diagramas de Flujo, Diagramas N-S, Pseudocódigo y Java</i>	D.F.	México	Alfaomega Grupo Editor

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	

METODOLOGÍA DE LA PROGRAMACIÓN

FUENTES BIBLIOGRÁFICAS

Autor	Año	Título del Documento	Ciudad	País	Editorial
Gastón C. Hillar	2015 ISBN: 9781785289934	<i>Learning Object-Oriented Programming</i>	Birmingham	Reino Unido	Packt Publishing
Vaskaran Sarcar	2016 ISBN: 9781484225448	<i>Interactive Object Oriented Programming in Java</i>	New York	EEUU	Editorial Apress

ELABORÓ:	Comité técnico académico de diseño curricular del subsistema de CGUTyP de la familia de carreras de Tecnologías de la Información	REVISÓ:	Dirección Académica	
APROBÓ:	C. G. U. T. y P.	FECHA DE ENTRADA EN VIGOR:	Septiembre de 2018	