

DATOS DE LA ASIGNATURA

Nombre de la asignatura:	Programación
Carrera:	Ingeniería en Energías Renovables
Clave de la asignatura:	ERF-1023
Créditos SATCA:	2-3-5

2.- PRESENTACIÓN

Caracterización de la asignatura.

Esta asignatura aporta al perfil del Ingeniero en Energías Renovables la capacidad para formular, gestionar, evaluar y resolver problemas relacionados con las fuentes renovables de energía, en el marco del desarrollo sustentable.

Para integrarla se ha hecho un análisis en el campo de esta Ingeniería, identificando los temas de programación que tienen una mayor aplicación en el quehacer profesional de este ingeniero.

Aportación al perfil

- a) Formular, gestionar y evaluar proyectos de desarrollo de ingeniería relacionados con las fuentes renovables de energía, en el marco del desarrollo sustentable.
- b) Diseñar, gestionar, implementar y controlar actividades de instalación, operación y mantenimiento de sistemas ingenieriles utilizados para la transformación y almacenamiento de la energía proveniente de fuentes renovables
- c) Colaborar en proyectos de investigación, desarrollo e innovación tecnológicos, relacionados con la energía proveniente de fuentes renovables.

Intención didáctica

Se organiza el temario, en siete unidades, agrupando los contenidos conceptuales de la asignatura en las dos primeras unidades; se incluyen cinco unidades más que se destina a la aplicación y desarrollo de soluciones a problemas específicos de los conceptos abordados en las dos primeras.

Se abordan los conceptos básicos de programación al comienzo del curso buscando una visión de conjunto de este campo de estudio. Al estudiar la clasificación de los lenguajes de programación se incluyen los conceptos involucrados con ella para hacer un tratamiento más significativo, oportuno e

integrado de dichos conceptos, que permitirán al estudiante resolver problemas.

Se sugiere una actividad integradora, a partir de la tercera unidad, que permita aplicar los conceptos estudiados de las unidades uno y dos. Esto permite dar un cierre a la materia mostrándola como útil por sí misma en el desempeño profesional.

El enfoque sugerido para la materia requiere que las actividades prácticas promuevan el desarrollo de habilidades para la formulación, gestión y la evaluación de proyectos, así como el uso de un lenguaje de programación; a través de: selección, iteración, identificación, y control de variables, datos relevantes, trabajo en equipo; asimismo, propicien procesos intelectuales como inducción-deducción y análisis-síntesis con la intención de generar una actividad intelectual compleja; por esta razón varias de las actividades prácticas se han descrito como actividades previas al tratamiento teórico de los temas, de manera que no sean una mera corroboración de lo visto previamente en clase, sino una oportunidad para conceptualizar a partir de lo observado. En las actividades prácticas sugeridas, es conveniente que el profesor busque sólo guiar a sus alumnos para que ellos hagan la elección de las técnicas a utilizar.

La lista de actividades de aprendizaje no es exhaustiva, se sugieren sobre todo las necesarias para hacer más significativo y efectivo el aprendizaje. Algunas de las actividades sugeridas pueden hacerse como actividad extra clase y comenzar el tratamiento en clase a partir de la discusión de los resultados de las observaciones e investigaciones documentales realizadas por el estudiante. Se busca partir de experiencias cotidianas, para que el estudiante se acostumbre a reconocer los procesos a su alrededor y no sólo se hable de ellos en el aula. Es importante ofrecer escenarios distintos, ya sean contruídos, artificiales, virtuales o naturales.

En las actividades de aprendizaje sugeridas, generalmente se propone la formalización de los conceptos a partir de experiencias concretas; se busca que el alumno tenga el primer contacto con el concepto en forma concreta y sea a través de la observación, la reflexión y la discusión que se dé la formalización; la resolución de problemas se hará después de este proceso. Pero se sugiere que se diseñen problemas de manera que el alumno se ejercite en la identificación de datos relevantes y elaboración de algoritmos y/o programas de solución.

En el transcurso de las actividades programadas es muy importante que el estudiante aprenda a valorar las actividades que lleva a cabo y entienda que está contruyendo su hacer futuro y en consecuencia actúe de una manera profesional; de igual manera, aprecie la importancia del conocimiento y los hábitos de trabajo; desarrolle la precisión y la curiosidad, la puntualidad, el entusiasmo y el interés, la tenacidad, la flexibilidad y la autonomía.

Es necesario que el profesor ponga atención y cuidado en estos aspectos en el desarrollo de las actividades de aprendizaje de esta asignatura.

3.- COMPETENCIAS A DESARROLLAR

Competencia específica <ul style="list-style-type: none">• Implementar mediante el uso de estructuras de control, bibliotecas, funciones, arreglos y archivos, problemas donde intervienen los sistemas renovables de energía.	Competencias genéricas: Competencias instrumentales <ul style="list-style-type: none">• Capacidad de Abstracción, análisis y síntesis.• Capacidad de organizar y planificar el tiempo.• Conocimientos sobre el área de estudio y la profesión.• Comunicación oral y escrita• Habilidades básicas de manejo de la computadora• Habilidad para buscar, procesar y analizar información proveniente de fuentes diversas.• Capacidad creativa.• Capacidad para identificar, planear y resolver problemas.• Tomar de decisiones. Competencias interpersonales <ul style="list-style-type: none">• Capacidad crítica y autocrítica• Trabajo en equipo• Habilidades interpersonales• Habilidad de Observación• Compromiso ético. Competencias sistémicas <ul style="list-style-type: none">• Capacidad de aplicar los conocimientos en la práctica• Habilidades de investigación• Capacidad de aprender• Capacidad de generar nuevas ideas (creatividad)• Habilidad para trabajar en forma autónoma
---	---

4.- HISTORIA DEL PROGRAMA

LUGAR Y FECHA DE ELABORACIÓN O REVISIÓN	PARTICIPANTES	OBSERVACIONES (CAMBIOS Y JUSTIFICACIÓN)
Instituto Tecnológico de Puebla, del 8 al 12 de junio de 2009.	Representantes de los Institutos Tecnológicos de: Celaya, Minatitlán, Saltillo, Toluca, Veracruz, Villahermosa y Milpa Alta.	Contexto global y nacional en energía. Marco jurídico nacional e internacional. Justificación de la carrera
Instituto Tecnológico de Puerto Vallarta, del 10 al 14 de Agosto de 2009.	Representantes de los Institutos Tecnológicos de Chihuahua, Chihuahua II, Chilpancingo, Durango, La Piedad, León, Mexicali, Milpa Alta, Minatitlán, Saltillo, Toluca, Villahermosa, Orizaba y La Laguna.	Reunión de Diseño curricular de la carrera, definiendo la retícula y los programas sintéticos.
Instituto Tecnológico de Villahermosa del 24 al 28 de agosto de 2009	Representantes de los Institutos Tecnológicos de: Toluca, Saltillo, Minatitlán y Villahermosa	Formulación de programas desarrollados para las materias de primer semestre
Instituto Tecnológico de Villahermosa del 28 de agosto del 2009 al 21 de mayo de 2010.	Representantes de la Academia de Sistemas y Computación.	Formulación de propuesta de programa desarrollado por competencias.
Instituto Tecnológico de Villahermosa del 24 al 28 de mayo de 2010	Representantes de los Institutos Tecnológicos de: Chihuahua, León, Mexicali, Minatitlán, Saltillo, Toluca, Veracruz, Villahermosa y Milpa Alta.	Reunión Nacional de Consolidación de la carrera de Ingeniería en energías renovables.

5.- OBJETIVO GENERAL DEL CURSO

- Implementar mediante el uso de estructuras de control, bibliotecas, funciones, arreglos y archivos, problemas donde intervienen los sistemas renovables de energía.

6.- COMPETENCIAS PREVIAS

- Aplicar los conocimientos conceptuales en la práctica.
- Manejo en el uso básico de computadoras.
- Habilidad de aprender y actualizarse permanentemente.
- Capacidad para actuar en nuevas situaciones.
- Conocimientos de Aritmética, Probabilidad y Estadística.

7.- TEMARIO

Unidad	Temas	Subtemas
1	Antecedentes y algoritmos.	1.1 Introducción al uso de las tecnologías de información en las ingenierías. 1.2 Ambientes operativos (Windows, Mac, Unix, Linux y otros). 1.3 Fundamentos de algoritmos. 1.4 Métodos para solución de problemas con algoritmos. 1.5 Estructuras algorítmicas. 1.6 Diagramación. 1.7 Pruebas de escritorio.
2	Entorno del lenguaje de programación.	2.1 Fundamentos conceptuales de programación (datos, información, lenguajes de alto nivel, tipos de datos, lenguajes de aplicación, compiladores, ensambladores, utilería de archivos, bibliotecas). 2.2 Estructura de un programa el lenguaje a utilizar. 2.3 Declaración de variables. 2.4 Sentencias de entrada/Salida de datos. 2.5 Expresiones (Aritméticas, lógicas y relacionales). 2.6 Depuración de un programa.
3	Estructuras de control.	3.1 Asignación.

4	Modularidad.	3.2 Selección (<i>if</i> , <i>switch</i> , <i>case</i>). 3.3 Iteración (<i>do</i> , <i>while</i> , <i>for</i>). 3.4 Combinadas y anidadas. 4.1 Optimización de código. 4.2 Descomposición funcional (<i>top-down</i>). 4.3 Rutinas, Funciones y Procedimientos. 4.4 Paso de argumentos.
5	Uso de bibliotecas del lenguaje.	5.1 Biblioteca matemática. 5.2 Biblioteca de números aleatorios. 5.3 Biblioteca gráfica. 5.4 Biblioteca definida por el programador.
6	Arreglos y datos	6.1 Arreglos unidimensionales: algoritmo, codificación y aplicación. 6.2 Arreglos multidimensionales: algoritmo, codificación y aplicación. 6.3 Representación de TDA: algoritmo, codificación y aplicación.
7	Tópicos de programación.	7.1 Procesamiento de archivos. 7.2 Diseño de GUI's.

8.- SUGERENCIAS DIDÁCTICAS (desarrollo de competencias genéricas)

El profesor debe:

- Ser conocedor de la disciplina que está bajo su responsabilidad, conocer su origen y desarrollo histórico para considerar este conocimiento al abordar los temas.
- Desarrollar la capacidad para coordinar y trabajar en equipo; orientar el trabajo del estudiante y potenciar en él la autonomía, el trabajo cooperativo y la toma de decisiones.
- Mostrar flexibilidad en el seguimiento del proceso formativo y propiciar la interacción entre los estudiantes.
- Tomar en cuenta el conocimiento de los estudiantes como punto de partida y como obstáculo para la construcción de nuevos conocimientos.
- Propiciar actividades de búsqueda, selección y análisis de información en distintas fuentes.

- Fomentar actividades grupales que propicien la comunicación, el intercambio argumentado de ideas, la reflexión, la integración y la colaboración de y entre los estudiantes.
- Observar y analizar fenómenos y problemáticas propias del campo ocupacional.
- Relacionar los contenidos de esta asignatura con las demás del plan de estudios a las que ésta da soporte para desarrollar una visión interdisciplinaria en el estudiante.
- Propiciar el desarrollo de capacidades intelectuales relacionadas con la lectura, la escritura y la expresión oral.
- Proponer al estudiante problemas de programación que le permitan la integración de los contenidos de la asignatura para su análisis y solución
- Desarrollar actividades de aprendizaje que propicien la aplicación de los conceptos, modelos y metodologías que se van aprendiendo en el desarrollo de la asignatura.
- Relacionar los contenidos de la asignatura con el cuidado del medio ambiente; así como el ahorro de energía.
- Propiciar el uso de las nuevas tecnologías en el desarrollo de la asignatura. (Uso de software para solución de problemas).

9.- SUGERENCIAS DE EVALUACIÓN

La evaluación debe ser continua y formativa por lo que se debe considerar el desempeño en cada una de las actividades de aprendizaje, haciendo especial énfasis en:

- Elaborar algoritmos y diagramas para la solución de un problema real.
- Información obtenida durante las investigaciones solicitadas plasmada en documentos escritos.
- Elaborar reportes de prácticas.
- Realizar un cuadro comparativo de las diferentes alternativas de solución de un problema dado.
- Describir en un documento escrito de otras experiencias de solución que podrían realizarse adicionalmente.
- Diseñar e implementar programas en un lenguaje de programación utilizando los elementos descritos en el contenido del curso.
- Exámenes escritos para comprobar el manejo de aspectos teóricos y declarativos en cada unidad.
- Solución de Problemas y ejercicios propuestos de programas.

10.- UNIDADES DE APRENDIZAJE

UNIDAD 1.- ANTECEDENTES Y ALGORITMOS.

Competencia específica a desarrollar	Actividades de Aprendizaje
Identificar el uso de las tecnologías, ambientes operativos, diagramación y pruebas de escritorio en la solución de problemas.	<ul style="list-style-type: none">▪ Investigar y exponer información relacionada con el uso de las TIC's en el campo de la Ingeniería en Energías Renovables.▪ Identificar y elaborar un resumen de las diferencias y similitudes existentes entre los ambientes operativos que ofrecen Windows, Mac, Unix, Linux y otros ambientes.• Resolver problemas mediante la aplicación de algoritmos.• Resolver problemas donde aplique diagramación para la solución del mismo.

UNIDAD 2.- ENTORNO DEL LENGUAJE DE PROGRAMACIÓN.

Competencia específica a desarrollar	Actividades de Aprendizaje
Identificar los fundamentos conceptuales de programación. Utilizar estructuras, sentencias y expresiones en la solución de problemas mediante la creación de programas.	<ul style="list-style-type: none">▪ Identificar y elaborar un resumen de los conceptos de programación: datos, información, lenguajes de alto nivel, tipos de datos, lenguajes de aplicación, compiladores, ensambladores, utilería de archivos, bibliotecas.▪ Explorar la web y documentar sobre la sintaxis y estructuras del lenguaje de programación.▪ Investigar y elaborar un resumen las palabras reservadas del lenguaje.

UNIDAD 3.- ESTRUCTURAS DE CONTROL.

Competencia específica a desarrollar	Actividades de Aprendizaje
	<ul style="list-style-type: none">• Diseñar la solución de problemas

Diseñar soluciones a problemas específicos utilizando las técnicas de estructuras de control.	<p>a resolver con computadora utilizando alguna técnica algorítmica.</p> <ul style="list-style-type: none"> • Utilizar un lenguaje de programación para implementar soluciones de problemas planteados en el ámbito de su competencia. • Elaborar soluciones a problemas específicos mediante el uso de estructuras de control.
---	---

UNIDAD 4.- MODULARIDAD.

Competencia específica a desarrollar	Actividades de Aprendizaje
Diseñar soluciones a problemas específicos utilizando funciones y procedimientos.	<ul style="list-style-type: none"> • Elaborar programas en un lenguaje de programación para implementar soluciones de problemas que empleen rutinas, funciones y procedimientos.

UNIDAD 5.- USO DE BIBLIOTECAS DEL LENGUAJE.

Competencia específica a desarrollar	Actividades de Aprendizaje
<p>Diseñar soluciones a problemas específicos utilizando las diferentes bibliotecas del lenguaje de programación elegido.</p> <p>Desarrollar bibliotecas por parte del usuario para la solución de problemas.</p>	<ul style="list-style-type: none"> • Realizar un mapa conceptual sobre las distintas bibliotecas del lenguaje de programación elegido. • Elaborar programas en un lenguaje de programación donde se implementen bibliotecas definidas por el usuario.

UNIDAD 6.- ARREGLOS Y DATOS.

Competencia específica a desarrollar	Actividades de Aprendizaje
Diseñar soluciones a problemas específicos donde se implementen arreglos.	<ul style="list-style-type: none"> • Realizar programas en donde se declaren arreglos unidimensionales. • Realizar programas en donde se declaren arreglos multidimensionales.

	<ul style="list-style-type: none"> Realizar ejercicios en donde se implementen TDA.
--	--

UNIDAD 7.- TOPICOS DE PROGRAMACION.

Competencia específica a desarrollar	Actividades de Aprendizaje
<p>Diseñar soluciones a problemas específicos empleando el procesamiento de archivos.</p> <p>Diseñar soluciones a problemas específicos utilizando GUI's.</p>	<ul style="list-style-type: none"> Desarrollar programas en la solución de problemas empleando procesamiento de archivos. Diseñar interfaces amigables de sus programas.

11. FUENTES DE INFORMACIÓN

- López, Gustavo. Jeder, Ismael. Vega, Augusto. "Análisis y diseño de algoritmos: implementación en C y Pascal". Alfaomega. Primera edición. Buenos Aires. 2009.
- Ramírez, Felipe.(2007). "Introducción a la programación – Algoritmos y su implementación en VB.Net, C#, Java y C++". Alfaomega.
- Cairó, Osvaldo. (S/D) "Fundamentos de Programación". Pearson Educación.
- Deitel, Harvey M. (2008). "Como programar en Java". Prentice Hall. México.
- Deitel, Harvey M./ Deitel, Paul J. (S/D). "Cómo Programar C++". Pearson Educación. Sexta Edición.
- García, J. Baltasar / Laza, Rosalia. (S/D). "Metodología y tecnología de la programación". Pearson Educación.
- García, José R./ Giner, Bermejo. (S/D). "Programación estructurada en C". Pearson Educación.
- Joyanes Aguilar, Luis. (2006) "Programación En C++. Algoritmos, Estructura de datos y objetos". Mc Graw Hill. Segunda edición.
- Joyanes Aguilar, Luis. (S/D). "Fundamentos de programación". Mc Graw Hill.
- Schildt, Herbert. (S/D). "Fundamentos de Java". Mc Graw Hill

12.- PRÁCTICAS PROPUESTAS

- Realizar algoritmos y diagramación de expresiones aritméticas y lógicas.
- Realizar programas de: selección e iteración.
- Realizar programas: conversión de unidades métricas, monedas, tablas de multiplicar, en función a una calificación imprimir un mensaje, etc.
- Realizar programas: ordenamiento de números en un arreglo unidimensional o bidimensional utilizando el método de burbuja, resolver la determinante de una matriz, etc.

5. Realizar programas donde se emplee el uso de bibliotecas (matemáticas, números aleatorios, gráficas y las definidas por el programador)
6. Realizar programas que la entrada y salida de los datos se almacenen en archivos.