

1. Datos Generales de la asignatura

Nombre de la asignatura:	Programación
Clave de la asignatura:	IHD-1023
SATCA¹:	2-3-5
Carrera:	Ingeniería Hidrológica

2. Presentación

Caracterización de la asignatura

Esta asignatura aporta al perfil del Ingeniero Hidrólogo la competencia para formular algoritmos y construir programas que resuelvan problemas relacionados con recursos hídricos.

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

1. Utiliza las tecnologías de la información y la comunicación para eficientar el aprovechamiento de los recursos hídricos, así como predecir y prevenir desastres naturales en beneficio de la sociedad.

Programación tiene relación directa con métodos numéricos ya que le permite utilizar algoritmos numéricos para obtener soluciones aproximadas de modelos matemáticos de interés en física e ingeniería, aplicando como herramienta un lenguaje de programación y tiene relación indirecta con Estadística y Técnicas estadísticas en hidrología, ya que puede elaborar programas que le permitan agilizar el análisis y la interpretación de los datos.

Intención didáctica

La asignatura se organiza en seis temas, en el primero se construyen competencias que le permiten distinguir los diferentes entornos y la importancia de las tic's para su carrera, así como la construcción de algoritmos, su diagramación y la aplicación de pruebas de escritorio para evaluar los algoritmos construidos.

En el segundo tema se revisan los entornos de desarrollo y se aplica la sintaxis del lenguaje en la codificación de los algoritmos construidos en el primer tema.

En el tercer tema se trabaja con las estructuras de control condicional e iterativo, se revisan problemas que involucren el uso de estas estructuras, se construyen algoritmos y se desarrollan programas en el lenguaje, es importante que se realicen los programas y se corrijan los errores antes de llegar a la compilación del programa para generar buenas prácticas de programación.

En el cuarto tema se trabaja con funciones y procedimientos, buscando que el alumno desarrolle las competencias que le permitan construir programas modulares que mejoren la estructuración del código para facilitar el desarrollo y la depuración de estos.

En el tema cinco se integran a los programas la estructura de arreglos para poder manipular datos de

¹ Sistema de Asignación y Transferencia de Créditos Académicos

vectores y matrices, así como tablas que puedan ser utilizadas posteriormente con el manejo de archivos.

En el tema seis se utilizan las bibliotecas de funciones que proporciona el lenguaje para desarrollar programas robustos que permitan resolver problemas complejos, se trabaja con archivos para almacenar datos necesarios para la toma de decisiones y se aplican las Interfaces Gráficas de Usuario (GUI's) que le permitan desarrollar ambientes amigables a los programas construidos.

3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Evento
Instituto Tecnológico de Roque, del 6 al 8 de octubre de 2010.	Representantes de los Institutos Tecnológicos de: Apizaco, Boca del Río, Chilpancingo, Ciudad Madero, Orizaba, Pachuca, Roque, Superior de Irapuato, Superior de Poza Rica, Altiplano de Tlaxcala, Toluca, Veracruz y Villahermosa.	Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de la Carrera de Ingeniería Hidrológica.
Instituto Tecnológico de Roque, del 6 al 9 de diciembre de 2010.	Representantes de los Institutos Tecnológicos de: Apizaco, Boca del Río, Orizaba, Pachuca, Roque, Superior de Irapuato, Superior de Poza Rica, Veracruz y Villahermosa.	Reunión Nacional de Consolidación de los Programas en Competencias Profesionales de la Carrera de Ingeniería Hidrológica.
Instituto Tecnológico de Roque, el 3 y 4 de noviembre de 2011.	Representantes de los Institutos Tecnológicos de: Apizaco, Boca del Río, Orizaba y Roque.	Reunión de Trabajo para la Consolidación del Programa en Competencia de la Carrera de Ingeniería Hidrológica.
Instituto Tecnológico de Toluca, del 10 al 13 de febrero de 2014.	Representantes de los Institutos Tecnológicos de: Chilpancingo y Roque.	Reunión de Seguimiento Curricular del Programa en Competencia de la Carrera de Ingeniería Hidrológica.

4. Competencia(s) a desarrollar

Competencias específicas de la asignatura

Desarrolla y evalúa programas mediante el uso de estructuras de control, bibliotecas, funciones, arreglos y archivos para dar solución a problemas donde intervienen recursos hídricos.

5. Competencias previas

- Ninguna

6. Temario

No.	Temas	Subtemas
1	Antecedentes y algoritmos.	1.1. Introducción al uso de las tecnologías de información en las ingenierías. 1.2. Ambientes operativos. 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	Fundamentos 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 del 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. 3.2. Selección (if, switch, case). 3.3. Iteración (do, while, for). 3.4. Combinadas y anidadas.
4	Modularidad.	4.1. Optimización de código. 4.2. Descomposición funcional (top-down). 4.3. Rutinas, Funciones y Procedimientos. 4.4. Paso de argumentos.
5	Arreglos y datos	5.1. Arreglos unidimensionales: algoritmo, codificación y aplicación. 5.2. Arreglos multidimensionales: algoritmo, codificación y aplicación. 5.3. Representación de TDA: algoritmo, codificación y aplicación.
6	Tópicos de programación	6.1. Bibliotecas del lenguaje 6.1.1. Biblioteca matemática. 6.1.2. Biblioteca de números aleatorios. 6.1.3. Biblioteca gráfica. 6.1.4. Biblioteca definida por el programador. 6.2. Procesamiento de archivos. 6.3. Diseño de GUI's.

7. Actividades de aprendizaje de los temas

Antecedentes y Algoritmos	
Competencias	Actividades de aprendizaje
<p>Específicas:</p> <p>Aplica y evalúa los fundamentos de programación, diagramación y prueba de escritorio para construir algoritmos que resuelvan problemas donde intervengan recursos hídricos.</p> <p>Genéricas:</p> <p>Habilidades para buscar, procesar y analizar información procedente de fuentes diversas.</p> <p>Capacidad de abstracción, análisis y síntesis.</p> <p>Capacidad de aplicar los conocimientos en la práctica.</p> <p>Habilidades en el uso de las tecnologías de la información y de la comunicación.</p> <p>Capacidad para identificar, plantear y resolver problemas.</p>	<p>Investigar y exponer información relacionada con el uso de las TIC's en el campo de la Ingeniería Hidrológica.</p> <p>Identificar y elaborar un resumen de las diferencias y similitudes existentes entre los ambientes operativos.</p> <p>Resolver problemas mediante la aplicación de algoritmos.</p> <p>Resolver problemas donde aplique diagramación para la solución del mismo.</p> <p>Realizar pruebas de escritorio que le permitan evaluar y corregir los algoritmos construidos.</p>
Fundamentos de Programación	
Competencias	Actividades de aprendizaje
<p>Específicas:</p> <p>Aplica la sintaxis del lenguaje de programación en la solución de ejemplos para desarrollar programas que cumplan con la estructura de dicho lenguaje.</p> <p>Genéricas:</p> <p>Habilidades en el uso de las tecnologías de la información y de la comunicación</p> <p>Capacidad de abstracción, análisis y síntesis.</p> <p>Capacidad para actuar en nuevas situaciones.</p> <p>Habilidades para buscar, procesar y analizar información procedente de fuentes diversas.</p> <p>Capacidad para identificar, plantear y resolver problemas.</p>	<p>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.</p> <p>Explorar la web y documentar sobre la sintaxis y estructuras del lenguaje de programación.</p> <p>Investigar y elaborar un resumen las palabras reservadas del lenguaje.</p> <p>redactar ejemplos de aplicación de las instrucciones aplicando la sintaxis de las instrucciones.</p> <p>Codificar programas en base a los algoritmos desarrollados.</p>

Estructuras de control	
Competencias	Actividades de aprendizaje
<p>Específicas: Aplica estructuras de control condicional e iterativo para construir programas que resuelvan problemas donde intervengan recursos hídricos.</p> <p>Genéricas: Habilidades en el uso de las tecnologías de la información y de la comunicación. Capacidad de abstracción, análisis y síntesis. Capacidad para actuar en nuevas situaciones. Habilidades para buscar, procesar y analizar información procedente de fuentes diversas. Capacidad para identificar, plantear y resolver problemas.</p>	<p>Identificar y construir algoritmos con estructuras de control. Elaborar ejemplos aplicando la sintaxis de las estructuras de decisión. Elaborar programas utilizando estructuras de decisión. Elaborar ejemplos aplicando las sintaxis de las estructuras de control iterativo. Elaborar programas utilizando estructuras iterativas.</p>
Modularidad	
Competencias	Actividades de aprendizaje
<p>Específicas: Aplica funciones y procedimientos para construir programas que resuelvan problemas donde intervengan recursos hídricos.</p> <p>Genéricas: Habilidades en el uso de las tecnologías de la información y de la comunicación. Capacidad de abstracción, análisis y síntesis. Capacidad para actuar en nuevas situaciones. Habilidades para buscar, procesar y analizar información procedente de fuentes diversas. Capacidad para identificar, plantear y resolver problemas.</p>	<p>Dividir un problema utilizando funciones y procedimientos para su solución. Desarrollar los algoritmos de los problemas planteados. Elaborar ejemplos de instrucciones aplicando la sintaxis de funciones y procedimientos. Elaborar programas en un lenguaje de programación para implementar soluciones de problemas que empleen rutinas, funciones y procedimientos.</p>
Arreglos y Datos	
Competencias	Actividades de aprendizaje
<p>Específicas: Aplica arreglos para construir programas que resuelvan problemas donde intervengan recursos hídricos.</p>	<p>Identificar requerimientos de uso de vectores y matrices que pueden ser implementados con arreglos de datos. Realizar programas en donde se declaren arreglos</p>

<p>Genéricas: Habilidades en el uso de las tecnologías de la información y de la comunicación. Capacidad de abstracción, análisis y síntesis. Capacidad para actuar en nuevas situaciones. Habilidades para buscar, procesar y analizar información procedente de fuentes diversas. Capacidad para identificar, plantear y resolver problemas.</p>	<p>unidimensionales. Realizar programas en donde se declaren arreglos multidimensionales. Realizar ejercicios en donde se implementen TDA.</p>
Tópicos de Programación	
Competencias	Actividades de aprendizaje
<p>Específicas: Aplica y diseña diferentes bibliotecas del lenguaje elegido, procesamiento de archivos y GUI's, para construir programas que resuelvan problemas donde intervengan recursos hídricos. Genéricas: Habilidades en el uso de las tecnologías de la información y de la comunicación. Capacidad de abstracción, análisis y síntesis. Capacidad para actuar en nuevas situaciones. Habilidades para buscar, procesar y analizar información procedente de fuentes diversas. Capacidad para identificar, plantear y resolver problemas.</p>	<p>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. Realizar una síntesis de la aplicación y utilización de archivos. Desarrollar programas en la solución de problemas empleando procesamiento de archivos. Diseñar interfaces amigables de sus programas. Elaborar programas con interfaces amigables.</p>

8. Práctica(s)

<ul style="list-style-type: none"> • Edición, compilación y depuración de programas basados en algoritmos que resuelvan problemas con flujo lineal. • Edición, compilación y depuración de programas basados en algoritmos que resuelvan problemas con flujo condicional simple y múltiple. • Edición, compilación y depuración de programas basados en algoritmos que resuelven problemas de flujo repetitivo. • Edición, compilación y depuración de programas basados en algoritmos que resuelven problemas que utilicen funciones y procedimientos. • Edición, compilación y depuración de programas basados en algoritmos que resuelven problemas que utilicen diferentes tipos de bibliotecas. • Edición, compilación y depuración de programas basados en algoritmos que resuelven problemas que utilicen archivos de datos.

9. Proyecto de asignatura

El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:

- **Fundamentación:** marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo.
- **Planeación:** con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo.
- **Ejecución:** consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar.
- **Evaluación:** es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la Meta cognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.

10. Evaluación por competencias

En el proceso de evaluación las evidencias estarán conformadas por:

- Evaluación de los conceptos: exámenes diseñados considerando alguna taxonomía para identificar el nivel de complejidad del aprendizaje, reporte de investigación documental, mapas mentales.
- Evaluación de habilidades: Documentos con los algoritmos y programas desarrollados, realización de prácticas de edición, compilación y depuración de programas.
- Evaluación de actitudes: Participación, exposiciones, asistencia.

Para evaluar las actividades de aprendizaje se solicitarán: Reportes de prácticas,

- La evaluación se debe basar en criterios establecidos en rubricas para revisar los programas elaborados que consideren, el diseño de los algoritmos, la codificación del programa, su ejecución y prueba.
- Se sugiere utilizar listas de observación para el trabajo de edición, codificación y depuración de programas en el laboratorio y para las exposiciones en clase.
- Se sugiere la utilización de un portafolio de evidencias que considere las evidencias del curso por tema.
- Se sugiere realizar evaluación formativa en el desarrollo de las prácticas a través de la retroalimentación en el proceso.

11. Fuentes de información

1. López, G, Jeder, I, Vega, A. (2009). “Análisis y diseño de Algoritmos: implementación en C y Pascal”. Ed. Alfaomega. Buenos Aires.
2. Ramírez, F. (2007). “Introducción a la programación – Algoritmos y su implementación en VB.Net, C#, Java y C++”. Ed. Alfaomega. Buenos Aires.
3. Cairó, O. (S/D) “Fundamentos de Programación”. Pearson Educación.
4. Deitel, Harvey M. (2008). “Como programar en Java”. Prentice Hall. México.
5. Deitel, Harvey M./ Deitel, Paul J. (S/D). “Cómo Programar C++”. Pearson Educación. Sexta Edición.
6. García, J. Baltasar / Laza, R. (S/D). “Metodología y tecnología de la programación”. Pearson Educación.
7. García, José R./ Giner, Bermejo. (S/D). “Programación estructurada en C”. Pearson Educación.
8. Joyanes A, L. (2006) "Programación En C++. Algoritmos, Estructura de datos y objetos". Mc Graw Hill. Segunda edición.
9. Joyanes A, L. (S/D). “Fundamentos de programación”. Mc Graw Hill.