



Universidad Michoacana
de San Nicolás de Hidalgo

UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO
COORDINACIÓN GENERAL DE BACHILLERATO

PROGRAMA DE UNIDAD DE APRENDIZAJE



1. Datos Generales de la Unidad de Aprendizaje

Nombre de la Unidad de Aprendizaje	Clave	Semestre
Principios de Programación	3-UATP-20	3°

Carácter	Optativa-trayectoria	Tipo	Teórico-Práctico

Unidades de Aprendizaje antecedentes	Unidades de Aprendizaje consecuentes
Álgebra	Estructura de Datos
	Programación Orientada a Objetos

Horas teóricas	Horas prácticas	Total, de horas por semana	Semanas por semestre	Total, de horas por semestre	Valor en créditos
1	2	3	16	48	6

Autores del programa	Fecha de elaboración		Fecha de aprobación de los Consejos Técnicos de Bachillerato
Dr. Martín Herrejón Escutia Dr. Manuel Guzmán Tapia M.C. José Alejandro Chávez Cortés M.C. Juan Pablo Aguado Ayala M.I. Jesús Perea Núñez ING. Wendolín Jacinto Díaz			
Revisores del programa	Fecha de revisión	Porcentaje de ajuste	Fecha de visto bueno del Consejo Académico del Bachillerato



2. Presentación de la Unidad de Aprendizaje

Propósito y vinculación con el perfil del egresado

El Propósito de la Asignatura es comprender, conocer, analizar y aplicar los elementos básicos de un lenguaje de programación que le permitan diseñar y elaborar (implementar) un algoritmo en sus distintas interpretaciones a partir del planteamiento de un problema.

Por tanto, se vincula de manera directa con las siguientes competencias genéricas del perfil del egresado:

- Utiliza las tecnologías de la información y comunicación para procesar e interpretar información
- Maneja las tecnologías de la información y la comunicación para obtener información y expresar ideas

Propuesta didáctico-metodológica

Presencial:

Expositiva

Aprendizaje por descubrimiento,

Predicción, observación, explicación (POE),

Aprendizaje orientado a proyectos,

Demostración y experimentación

Virtual:

Análisis de videos

Simuladores virtuales

Juegos didácticos

Foros de opinión

Descripción de actividades específicas en las que incorporarán al menos uno de los tópicos de formación transversal: identidad nicolaita, responsabilidad social, ética, género, educación inclusiva

Utilización de la programación que nos permita recopilar información general de los estudiantes, agregando campos para el manejo de la responsabilidad social dentro del contexto de cada estudiante



3. Competencias a desarrollar

Eje formativo
Propedéutico de Trayectoria
Competencias disciplinares
<ul style="list-style-type: none">Desarrolla habilidades en el uso de las tecnologías de la información y comunicación, de manera eficiente y responsable, para la resolución de problemas.

4. Perfil académico del docente

Grado académico:	Licenciatura en Informática, Ingeniería en sistemas computacionales o área afín.
Experiencia:	Al menos dos años de experiencia académica en el área (docencia o investigación)

5. Temas y subtemas

Temas	Subtemas
1. Diseño Algorítmico	1.1. Conceptos básicos. 1.2. Representación de algoritmos: gráfica y pseudocódigo. 1.3. Diseño de algoritmos. 1.4. Diseño de funciones.



2. Introducción a la Programación	2.1. Conceptos básicos 2.2. Características del lenguaje de programación. 2.3. Estructura básica de un programa. 2.4. Elementos del lenguaje: tipos de datos, literales, constantes, variables, identificadores, parámetros, operadores y salida de datos. 2.5. Traducción de un programa: compilación, enlace, ejecución y errores.
3. Control de Flujo	3.1. Estructura secuenciales 3.2. Estructuras selectivas: simple, doble y múltiple 3.3. Estructuras iterativas: repetir mientras, hasta, desde.
4. Organización de Datos	4.1. Arreglos 4.1.1. Unidimensionales: conceptos básicos, operaciones y aplicaciones. 4.1.2. Multidimensionales: conceptos básicos, operaciones y aplicaciones. 4.2. Apuntadores 4.2.1. Apuntadores y direcciones de memoria 4.2.2. Operadores de dirección e indirección 4.2.3. Aritmética de apuntadores 4.2.4. Manejo de arreglos con apuntadores 4.3. Datos definidos por el usuario 4.3.1. Declaración de una estructura 4.3.2. Instancia de estructura y apuntador a estructura 4.3.3. Operadores de estructura 4.3.4. Arreglos de estructuras 4.3.5. Estructuras anidadas 4.4. Funciones 4.4.1. Funciones 4.4.2. Argumentos y parámetros 4.4.3. Paso por valor y paso por referencia 4.4.4. Retorno de la función 4.4.5. Bibliotecas creadas por el usuario 4.5. Funciones recursivas 4.5.1. Recursión 4.5.2. Pasos base y pasos recursivos 4.5.3. Tipos de recursión



	4.5.4. Recursión versus iteración
5. Manejo de memoria dinámica y archivos	<ul style="list-style-type: none">5.1. Arquitectura de memoria<ul style="list-style-type: none">5.1.1. Harvard5.1.2. Von Neumann5.1.3. Memoria de un proceso en el modelo de memoria única5.2. Memoria dinámica<ul style="list-style-type: none">5.2.1. Reservación dinámica de memoria5.2.2. Acceso y direccionamiento a arreglos dinámicos5.2.3. Liberación de memoria5.2.4. Memoria estática vs. Memoria dinámica5.3. Archivos<ul style="list-style-type: none">5.3.1. Archivos de acceso secuencial y de acceso directo5.3.2. Creación y/o apertura de archivos5.3.3. Lectura/escritura de archivos en modo texto5.3.4. Lectura/escritura de archivos en modo binario5.3.5. Cierre de archivos5.3.6. Procesamiento y redireccionamiento de flujos de entrada y salida
6. Programando Gráficos	<ul style="list-style-type: none">6.1. Introducción a Swing6.2. Creando una ventana Frame6.3. Posicionando una ventana Frame6.4. Desplegando información en un componente6.5. Trabajando con figuras en dos dimensiones6.6. Utilizando color6.7. Utilizando tipos de letras especiales para los textos6.8. Desplegando imágenes
7. Manejando Eventos	<ul style="list-style-type: none">7.1. Las bases del manejo de eventos7.2. Acciones7.3. Eventos del Mouse7.4. La jerarquía de eventos del AWT (Abstract Window Toolkit)



6. Criterios de evaluación

CRITERIOS A EVALUAR	PORCENTAJE
Exposiciones por equipos de Trabajo	10%
Realización de Prácticas del Laboratorio de informática (rúbrica)	20%
Examen Practico	30%
Proyecto final Programación (rúbrica)	40%
Porcentaje final	100%

7. Fuentes de información

Básica:
Joyanes, L. (2012) Fundamentos generales de programación. España; McGraw Hill Joyanes, L. (2010) Programación en C, C++, Java y UML. México: McGraw-Hill. ISBN 978-970-10-6949-3. Joyanes, L., Fernández, M. y Rodríguez L. (2003) Fundamentos de Programación Libro de Problemas Algoritmos Estructura de Datos y Objetos. México: Mc. Graw Hill. Márquez, G. (2012) Introducción a la programación estructurada en C. España: Pearson
Complementaria:
Deitel, H. M. y P. J. Deitel.(1998). Cómo programar en C/C++. Prentice-Hall Hispanoamericana. Farrell, J. (1999). Introducción a la Programación. Lógica y Diseño. Ed. Thomson. Kernighan, B. W. y D. M. Ritchie (1991). El lenguaje de programación C. Prentice-Hall.. Quero , E (2001). Programación en Lenguajes Estructurados. Paraninfo.